Received: April 6, 2024. Revised: June 22, 2024.

INASS

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

A Novel Pseudo-Random Sequence Generator Based Cellular Automata and 2D Hénon Map: Employing Image Encryption via One-Time Pad

Mohamed Amine Nemmich[1]* **Asmaa Boudali[2]** **Adda Ali Pacha[2]**

[1]*Department of Computer Science, Mathematics Laboratory,*
Djillali Liabes University of Sidi Bel Abbes, Sidi Bel Abbes, Algeria
[2]*Department of Electronics, Laboratory of coding and security of information,*
University of Sciences and Technology of Oran Mohamed Boudiaf, Oran, Algeria
* Corresponding author's Email: amine.nemmich@univ-sba.dz

Abstract: This paper proposes a new pseudo-random sequence generator to improve the security of data. The method innovatively integrates a 2D Henon map into the evolutionary process of cellular automata to introduce chaotic features into selected dynamic neighbourhoods during evolution, boosting the generator's efficacy. The proposed generator outperforms static neighborhood-based pseudo-random generators and meets the criteria defined by the National Institute of Standards. To overcome the limitation of the one-time pad algorithm, where the key length must match the plaintext length, the proposed generator ensuring that the key length matches the length plain text. The encryption scheme achieves an average entropy information value of 7.99%, which indicates robustness against entropy attacks. Additionally, the Number of Pixels Change Rate (NPCR) was above 99% and the Unified Average Changing Intensity (UACI) was close to the ideal value of 33%, demonstrating its suitability for image encryption with enhanced security and minimal complexity.

Keywords: Cellular automata, Data security, Randomness, 2D henon map, Pseudo random sequence generator.

1. Introduction

As computing power increases, the risk of cybercrimes also rises, making data transmission security crucial [1]. Encryption is a vital method for safeguarding data, with research focusing on encrypting image data gaining prominence [2-5].

In this study, a digital image was chosen as the focus of investigation. Cellular Automata (CA), a type of discrete system, have been utilized by several researchers for image encryption, harnessing CA's capabilities to offer effective methods for image security. There are primarily two CA-based image encryption approaches. The first involves generating encryption keys where CA acts as a random number generator to create a key matrix filled with pseudo-random numbers, subsequently used to encrypt the original image. The second method uses CA to directly manipulate image data for encryption purposes, transforming the image into a binary format and applying a specific CA, like a second-order CA, to diffuse the data and scramble the image content.

There are many methods proposed during the literature based CA, including [6], where the work proposes a novel two step image encryption technique leveraging chaotic maps and CA to enhance image security. Initially, the method addresses strong pixel correlation in the original image by separating it into red (R), green (G), and blue (B) channels. Each channel undergoes independent permutation using a specific key and a 3D chaotic transformation. Subsequently, diffusion involves additional scrambling of the image data. The permuted image is transformed into 24 separate "one-bit patches", likely representing bit plane images where each pixel's value is divided into eight individual bits. These bit planes are then subjected to bitwise XOR operations using appropriate 2D reversible CA rules, utilizing CA generated keys to further scramble the individual bit values in each channel and enhance resistance against statistical attacks. In addition, a new image cryptosystem uses boundary elementary CA (ECA) with permutation diffusion architecture is designed in [7]. The method exploits specific attractors from elementary periodic boundary CAs to efficiently transform pixel values. It uses symmetric private key encryption with five keys generated to rearrange pixels, swap hash values, and perform bitwise XOR operations. The encryption process involves splitting blocks, calculating the hash, permuting image and hash blocks, and evolving the CA to produce state attractors. Each layer of the color image undergoes individual encryption steps; including hashing, permutation, and CA based encryption. The encrypted image is constructed by merging encrypted layers. Moreover, in [8], the authors utilize the CA to the confusion phase, introducing complexity and non-linearity to the encryption process. By leveraging both the chaotic dynamics of the logistic sine map and the state transitions of CA. Furthermore, in the research work [9], the proposed image encryption algorithm features three distinct stages. The first stage utilizes a Pseudo Random Generator key along with a novel 7D hyper chaotic function for diffusion. In the second stage, an extended CA S box is employed to redistribute image data. Lastly, a key generated by a CA is used to implement another diffusion stage. Additionally, an algorithm utilizes ECA in which, during the confusion phase, it shuffles the image according to a key controlled permutation box governed by a Henon chaotic map. Rule 15 of ECA plays a crucial role in the second phase, where a shuffled image serves as input for further encryption processes, integrating ECA dynamics for efficient image transformation and encryption [10]. The work presented in [11] introduces a novel image encryption algorithm based on chaotic maps and CA. Initially, the secret keys of the chaotic maps are computed using the SHA-256 hash value of the original image. Subsequently, the plaintext image is diffused and scrambled using CA. The final encrypted image is derived by transforming the scrambled image with CA. The algorithm employs 1D CA for bit-level image encryption and sequence numbers based on CA are randomly generated by evaluating the interval of values of the pseudo random sequence, thereby enhancing the randomness in selecting CA transformation rules. In [12], the authors present an image encryption scheme integrating hyper chaotic system, N+2 ring Joseph algorithm, and reversible CA operations. The N+2 ring Joseph algorithm disrupts pixel information using N rings rolling forward, offering enhanced scrambling and flexibility, leveraging N chaotic sequences to shift image pixels and enhance randomness. Reversible CA technology in diffusion involves XOR operations with chaotic matrices, followed by iterations to achieve a consistent histogram for the encrypted image. On another side, chaotic methods are also used in encryption method but focus on blurring pixel positions in the image. They will give essentially the same result as the original image. As a result, many studies advocate combining multiple encryption techniques to improve overall security [13-15]. As in [16], a hybrid encryption algorithm proposed combines MD5 hash algorithm, DNA sequence, and Henon chaotic map to enhance robustness against attacks. Unlike other evolutionary based encryption methods, this method employing the Henon chaotic function enhanced by the Imperialist Competitive Algorithm. The objective is to find optimal α and β, values for image encryption. On the other hand, among the algorithms used for image encryption, there is the One-Time Pad (OTP) algorithm also referred to as the Vernam cipher, which is a substitution cryptography algorithm known for its perfect security when using a truly unpredictable key stream [17]. However, the strength of this algorithm heavily relies on the key used which must be entirely random, used only once, and shared exclusively between the sender and receiver. Therefore, this paper aims to address this limitation by utilizing CA and chaotic maps to generate high-quality pseudorandom number sequences suitable for use as secure secret keys within the Vernam algorithm. This approach ensures that the key length matches the length of the plaintext, thereby enhancing practicality. While this concept is not novel, the key innovation lies in leveraging an internal combination of chaotic maps within the evolutionary process of CA to generate chaotic data. This stands in contrast to conventional methods, where external combinations are typically employed, such as using chaotic maps to generate private keys for CA as their initial configurations, or employing genetic algorithms to evolve Boolean functions with good cryptographic properties for CA, specifically focusing on applications like pseudo-random sequence generation [18]. In a related study [19], the research proposes a method for generating pseudo-random sequences of numbers using CA with two active cells. It achieves this by combining the outputs of two CA cells with an XOR operation, resulting in a sequence with better statistical properties. Another study investigates a new method for designing one-dimensional CA with five-cell neighborhoods for use in cryptographic key sequence generation is proposed [20]. This method addresses the challenge of creating symmetric CAs that correspond to specific mathematical properties (primitive polynomials) by leveraging transition matrices and block matrices.

In this paper, the proposed generator pseudo random sequence is constructing by integrating the 2D Henon map into the parameters evolution of ECA, this approach introduces chaotic characteristics within selected neighborhoods during evolution, thereby bolstering the efficacy of the new pseudo-random sequence generator. This generator exhibits robust random properties and has undergone thorough theoretical and experimental validation, demonstrating exceptional chaotic attributes and resilience against differential attacks. It outperforms classical pseudo-random generators commonly utilized for data generation [21] which using Rule 30 and the that selects a batch of n random bits from the center column to form the required n-bit random number, then proceeds sequentially to build subsequent random numbers using the next n bits. Importantly, the proposed generator successfully meets all criteria set forth by the National Institute of Standards. The use of chaotic systems in CA offers numerous mathematical advantages, allowing for the expansion of neighborhood space, the introduction of rich and complex dynamics, the modeling of natural systems, the optimization of complex problems, and the generation of random numbers. These properties make chaotic CAs a powerful tool for the study of complex systems, scientific research, and technological innovation. In addition, images encrypted via OTP with the new generator achieved good performance in terms of entropy, correlation and resistance to differential attacks, which brings them online and can be used in the field of cryptography, compared recent research proposed in this area.

The remainder of this paper is structured as follows: Section 2 introduces the basic concepts and terminologies of CA and the 2D Henon map. Section 3 outlines the proposed approach for constructing the new pseudo-random sequence generator and its application in image encryption. The security analysis results are discussed in Section 4, and Section 5 presents the conclusion.

2. Preliminaries

2.1 Cellular automata (Background)

A CA is a collection of simple cells comprising a regular network of cells defined as a simple type of computing machine that is discrete in both space and time [22]. The CA was presented in the 1940s by John von Neumann as a formal model of self-replicating systems, and Stanislaw Ulam, who examined the growth of crystals [23]. Subsequently, Wolfram introduced the CA as a mathematical model for self-organizing statistical systems [22].

The structure of CA is a discrete network of cells in one or more dimensions, where each cell has a finite number of possible states and maintains a state S. In formal terms, CA can be represented by a quadruple <*C*, *S*, *V*, *f*>, where: *C* is the automata cell, *S* contain a collection of potential states for all cells within a CA, *V* define the size of the automata and the set of neighboring states, and f define the transition rule for the transfer of CA, where f: *S*→*S*.

The configuration of CA is the state of all cells at time t, and it is denoted CA^t. The subsequent state of CA is represented by CA^{t+1}. It is supposed that the state CA_i^t of a cell *i* at time *t*+1 (C_i^{t+1}) is determined by the transition function, taking into account the current state and relying only on its neighboring cells at time *t*. This phenomenon is represented by Eq. (1):

$$CA_i^{t+1} = f\left(C_{i-r}^t, C_{i-r+1}^t,, C_i^t, C_{i+1}^t, \dots, C_{i+r}^t\right) \quad (1)$$

In ECA, each cell *i* called a central cell, is connected to *r* local neighbors on either side, where *r* is the radius. Thus, each cell has 2*r* + 1 neighbors, including the cell *i*. During the evolution, the next state of the cell *i* at time *t* +1 (S_{t+1} (*i*)) is updated by discrete time steps, depending on the old state $S_t(i)$ and the neighboring cells statements ($S_t(i$-1), $S_t(i$+1)), according to a well-defined rule (f). This phenomenon is represented by Eq. (2):

$$S_{t+1}(i) = f\left(S_t(i-1), S_t(i), S_t(i+1)\right) \quad (2)$$

In this type of model, there are two possible states, *S* ∊ (0,1) and 3-bit neighborhoods.

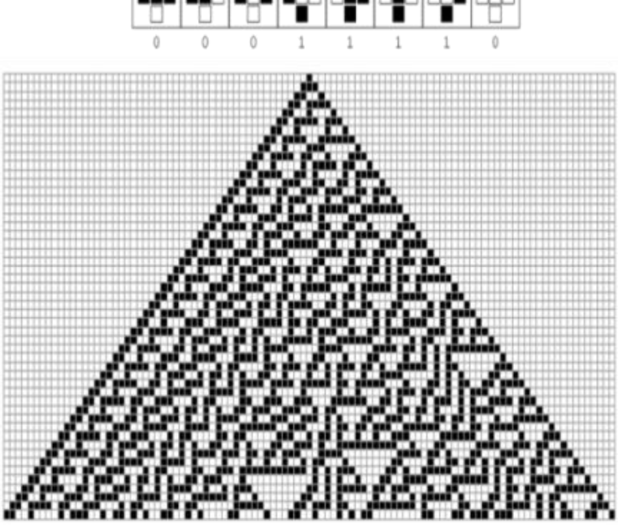


Figure. 1 Rule table and patterns of rule 30 CA

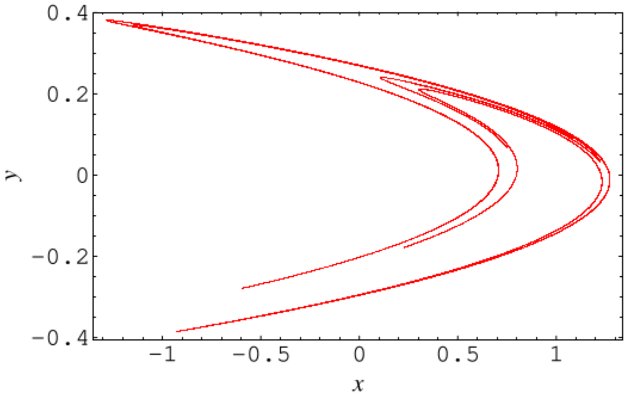


Figure. 2 A visualization of the Henon attractor

With 8 possible neighborhoods states (000,001,010,011, 100,101,110, 111) there are 2^{2^3}= 256 possible rules.

Wolfram has given a particular name for the 256 CA rules, where each rule is called by the decimal number given by the conversion of the binary string presented in its truth table [24]. Wolfram has encoded the set of local rules governing the time evolution of an ECA [22]. An illustration of CA rule 30 is provided in Fig. 1.

2.2 The two-dimensional hénon map

The Henon map is defined as one of the simplest 2D quadratic recurrence equations which Henon initially introduced them in [25] as a simplified model of the Poincaré section of the Lorenz model. Mathematically, the Henon map defined as follows:

$$x_{n+1} = 1 - \alpha x_n^2 + y_n \tag{3}$$

$$\mathrm{y_{n+1}} = \beta \mathrm{x_n} \tag{4}$$

The map is characterized by two control parameters, α and β, where n represents the number of map iterations. In the classical case of the Henon map with α = 1.4 and β = 0.3, the dynamical system can exhibit chaotic behaviors. Fig. 2 depicts the outline on a 2D plane for the Henon map.

3. The research methods

3.1 Process of constructing the generator pseudo random sequence

To evolve the ECA using the classical method [22], the new cell state (S_i^{t+1}) in the next configuration is generated according to the current state (S_i^t) and the two nearest neighbors of the state selected from the present configuration. The equation above describes this process.

$$S_i^{t+1} = (S_{i-1}^t, S_i^t, S_{i+1}^t) \tag{5}$$

However, in the evolutionary process proposed [15], two configurations are required to generate the next configuration in ECA: the initial configuration provides the neighbors of the precedent state, and the actual configuration provides the current state. This process is described by Eq. (6).

$$S_i^{t+1} = (S_{i-1}^{t-1}, S_i^t, S_{i+1}^{t-1}) \tag{6}$$

In This paper the proposed evolution incorporates the 2D Henon map as an internal parameter within the CA neighborhood, enhancing dynamism and chaos compared to static neighborhood approaches proposed in literature [26].

The new cell state (S_i^{t+1}) in the next configuration is generated based on the two chaotic chosen neighboring values (S_a, S_b) from the preceding configuration, and the previous state (S_i^t) from the current configuration (Eq. (7)).

$$S_i^{t+1} = (S_a^{t-1}, S_i^t, S_b^{t-1}) \tag{7}$$

The 2D Henon map utilized has lower computational complexity and excellent chaotic characteristics [27]. x_i, y_i are iterated values from the 2D Henon Map. These values can be identified as control parameters of Eqs. (8) and (9) to determine the position of the neighborhoods. This process is defined in the equations bellow, where a, and b define the position values in the input vector of configurations, and L is the length of the input vector.

$$a = ((x_i * 10^7) - i) mod(L) \tag{8}$$

$$b = (i + (y_i * 10^7)) \, mod(L) \tag{9}$$

In more detail, the steps for the proposed evolution process are described below:

3.1.1. Initialize parameters and configurations

a. Set pre-initial configurations CA_{-1}.
b. Obtain the dimensions of the initial configuration I and save it into a variable L.
c. Define vectors X and Y with dimensions (1×L), then fill them using a 2D Henon map to generate random numbers.
d. Define the transition rule R.

3.1.2. Evolution process

a. Generate the initial configuration CA_0 using Eqs. (10) and (11), where S_i is the current state from the pre initial configuration, and S_a and S_b are two chaotic neighboring values based X_i and Y_i selected from the pre initial configuration. Then, apply the chaotic conditions based on the values of *a* and *b* to modify S_a and S_b if necessary.

$$CA_0 = R(CA_{-1}) \tag{10}$$

$$CA_{-1} = (S_a^t, S_i^t, S_b^t) \tag{11}$$

b. Iterate through the following steps for each subsequent configuration CA_n:

- Generate the next configuration CA_1 based on the tow precedent configurations CA_{-1} and CA_0, using Eq. (12), where, S_i is the current state from the initial configuration CA_0, and S_a and S_b two chaotic neighboring values based X_i and Y_i selected from the pre-initial configuration CA_{-1}.

$$CA_1 = R\left((S_i^t)_{CA_0}, (S_a^{t-1}, S_b^{t-1})_{CA_{-1}}\right) \tag{12}$$

- Update CA_{-1} to CA_0 and CA_0 to CA_1 for the next iteration.
- Repeat the evolution process until the desired number of configurations is generated using the Eq. (13), where S_i is the current state from the actual configuration, and S_a and S_b are two chaotic neighboring values based X_i and Y_i selected from the precedent configuration.

$$CA_n = R\left((S_i^t)_{CA_{n-1}}, (S_a^{t-1}, S_b^{t-1})_{CA_{n-2}}\right) \tag{13}$$

Chaotic conditions:

Introduce conditions based on the values of *a* and *b* to add chaos to the evolution process.

- If both *a* and *b* are even, apply the modifications as follows:

$$S_a = XOR(S_a, S_b) \tag{14}$$

$$S_b = XOR(S_b, (NOT\ S_a)) \tag{15}$$

- If both *a* and *b* are odd, apply different modifications as follows:

$$S_a = XOR\ (S_a, (NOT\ S_b)); \tag{16}$$

$$S_b = S_b; \tag{17}$$

- Otherwise, leave S_a and S_b unchanged.

3.2 Application of the pseudo-random generator in image encryption based on one-time pad

In this subsection, the new pseudo-random generator is utilized in One-Time Pad (OTP) to generate a key of the same length as the plaintext image. Subsequently, the exclusive-OR operator is applied between them to produce the cipher image. The flow diagram of the encryption system is illustrated in Fig. 3.

The initial values for the CA are generated using the SHA-256 hash function applied to the original image. In cases where there’s a discrepancy of just 1 bit, the resulting hash values of the two images exhibit considerable divergence, thus augmenting the sensitivity of the encryption system to the secret key. Before proceeding with encryption, the 256-bit hash value of the plain image is computed and designated as the secret key, denoted as K. The plain image employed in this proposed cryptosystem is “Lena” with dimensions of 256×256 pixels is shown in Fig. 4. Fig. 5 illustrates the key image generated based on the hash value of Lena image and the encrypted image of Lena is depicted in Fig. 6.

As illustrated in Fig. 6, the encrypted image exhibits similar appearances and does not reveal any discernible visual details about the original plaintext image. Thus, the suggested schema offers effective visual security.

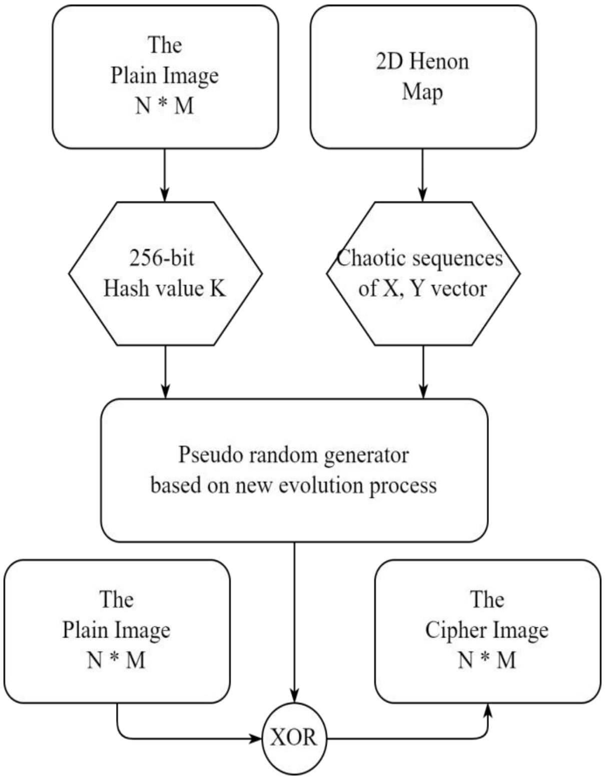


Figure. 3 Diagram of the stream cipher proposed

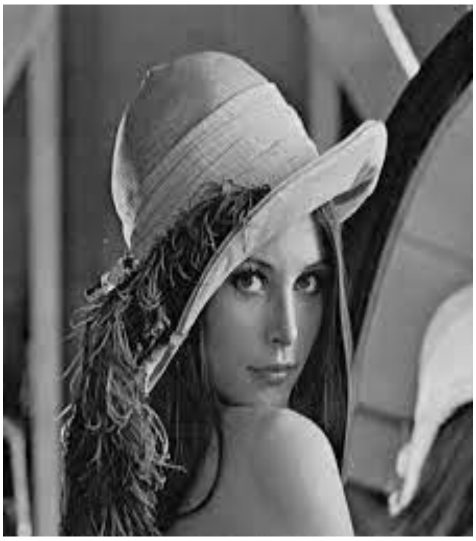


Figure. 4 Plain Lena image

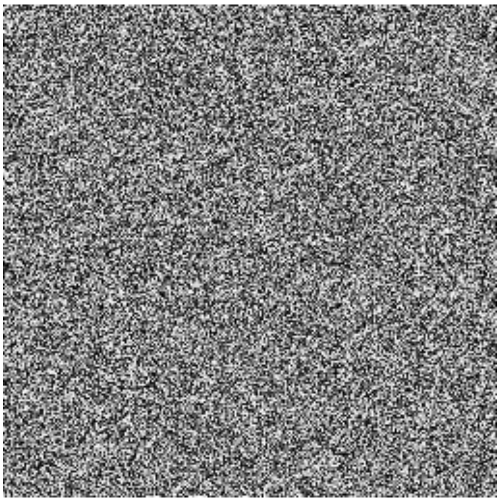


Figure. 5 Key image

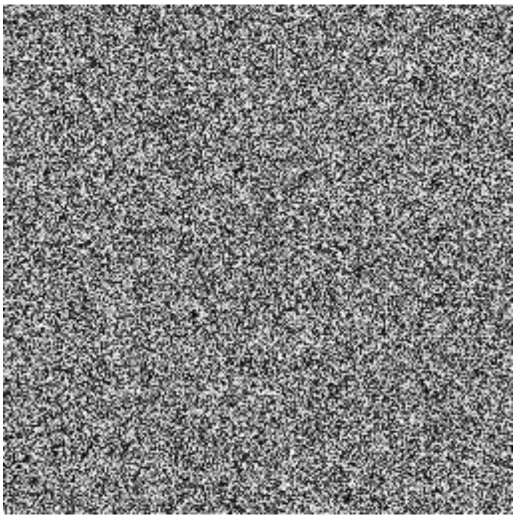


Figure. 6 Chiper image

4. Results and discussion

4.1 The pseudo random generator criteria

Evaluating the effectiveness of any new generator design requires thorough analysis and comparison of its security features with existing ones. In our case, the comparison of our proposed generator is carried out with the generator created by the classical evolution method used in most creations of AC-based generators with static neighborhood.

4.1.1. NIST SP 800-22 test

The statistical properties of the proposed pseudo-random number generator were rigorously evaluated using the NIST Test Suite [28].

Table 1. NIST test suite

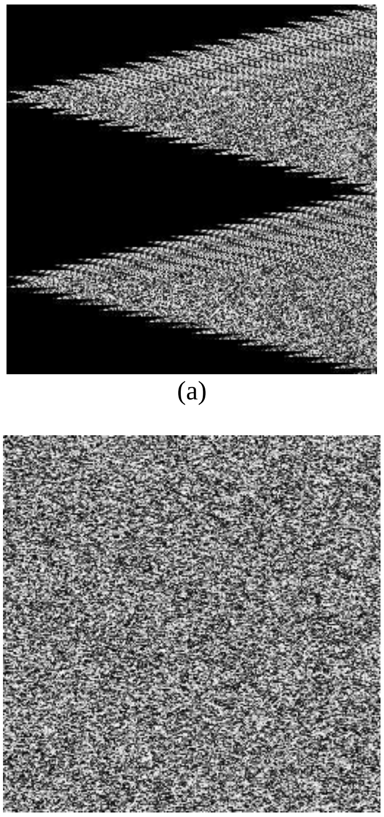
Test name	Our proposed generator		Generator based rule 30 [21]	
	P-values	Result	P-values	Result
monobit_test	0.7284483091099526	Pass	0.0	Fail
frequency_within_block_test	0.2854838330299568	Pass	0.0	Fail
runs_test	0.7610047231617658	Pass	0.0	Fail
longest_run_ones_in_a_block_test	0.21773182121075785	Pass	2.01797822502071e-23	Fail
binary_matrix_rank_test	0.04480654996864274	Pass	0.0	Fail
dft_test	0.39757181048441276	Pass	0.0	Fail
non_overlapping_template_matching_test	1.0000001050098712	Pass	0.9277500518708596	Pass
overlapping_template_matching_test	0.33093073070380513	Pass	3.2665446384295794	Fail
maurers_universal_test	0.33276185758963156	Pass	0.0	Fail
linear_complexity_test	0.10770300019099753	Pass	0.0	Fail
serial_test	0.016887866752506666	Pass	0.0	Fail
approximate_entropy_test	0.017130351217610633	Pass	0.0	Fail
cumulative_sums_test	0.6892676036074903	Pass	0.0	Fail
random_excursion_test	0.020208663444870618	Pass	2.772041441002831e-12	Fail
random_excursion_variant_test	0.03276509474012674	Pass	0.4795001221869535	Pass

A sequence of 1,034,240 bits was generated from the new generator initialized with a key having the central cell set to 1 and surrounded by 0.

Comprehensive testing revealed that this bit stream passed all evaluations in the NIST suite, exhibiting robust random properties resilient against pertinent attacks and attaining maximum confidence levels of 100% and 99% randomness. These results highlight the scientific advancement of the proposed approach, which demonstrates vastly improved statistical properties compared to the classic generator by [21] that failed all tests except the random excursion variant. This stark contrasts to the deficiencies of prior art underscores the scientific contribution of this research in developing a random number generator suitable for security and privacy solutions. Detailed results are provided in Table 1.

4.1.2. Histogram analysis

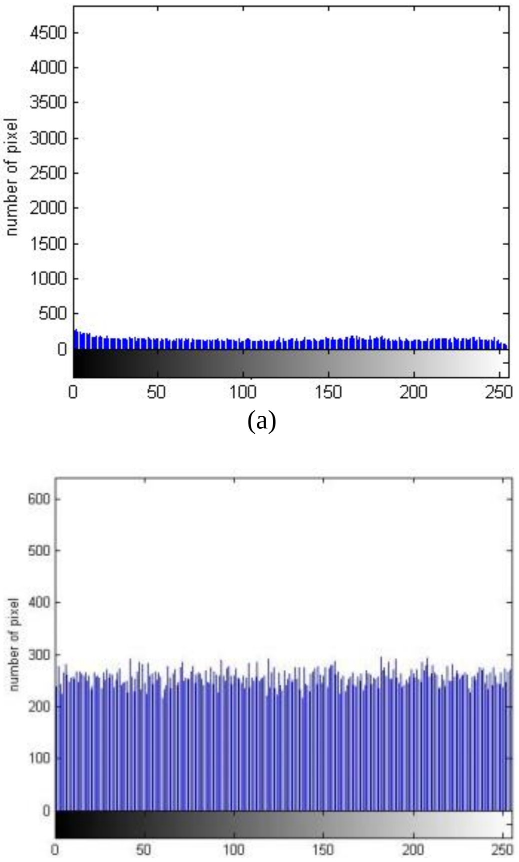
The histogram serves as a valuable visualization tool, revealing the tonal distribution and frequency of gray scale values within an image, a key metric for assessing randomness quality. Fig. 7 contrasts the images generated by the classic CA-based generator in (a) and the proposed new generator in (b). Fig. 8 shows their respective histograms.



(a)

(b)

Figure. 7 Generated images: (a) with classic CA-based generator and (b) based on new proposed process



(a)

(b)

Figure. 8 Histograms results: (a) data generated with classic CA-based generator and (b) data generated based on the proposed new generator

From histogram 1 unveils a uniform distribution spanning the full 0-255 intensity range for the new generator's output, indicating an ideal uniform probability distribution. These results demonstrate the generator's ability to produce data exhibiting the desired chaotic behaviour. Conversely, the classic generator fails this critical test.

4.1.3. Information entropy analysis

According to [29], Shannon entropy is a mathematical function used to measure uncertainty associated with random variables in an information source. For a source of discrete random variable X with n symbols, each symbol x_i has a probability P_i of occurring. The entropy H of source X can be calculated using Eqs. (17) and (18), where n is the image dimension (in our case, $n = 256 * 256 = 65536$), i indicates the symbol value ranging from 0 to 255, and k_i corresponds to the occurrence frequency of each value i. Ideally entropy is critical to ensure true randomness of a sequence. Consider a source with an alphabet of 256 characters: if all are equiprobable, the entropy is 8 bits.

Table 2. Entropy analysis

Nature of Source	Entropy in bits/symbols	The Rate in ideal source
Proposed work	7.9991	99.99 %
Generator based rule 30	5.1604	41.28%

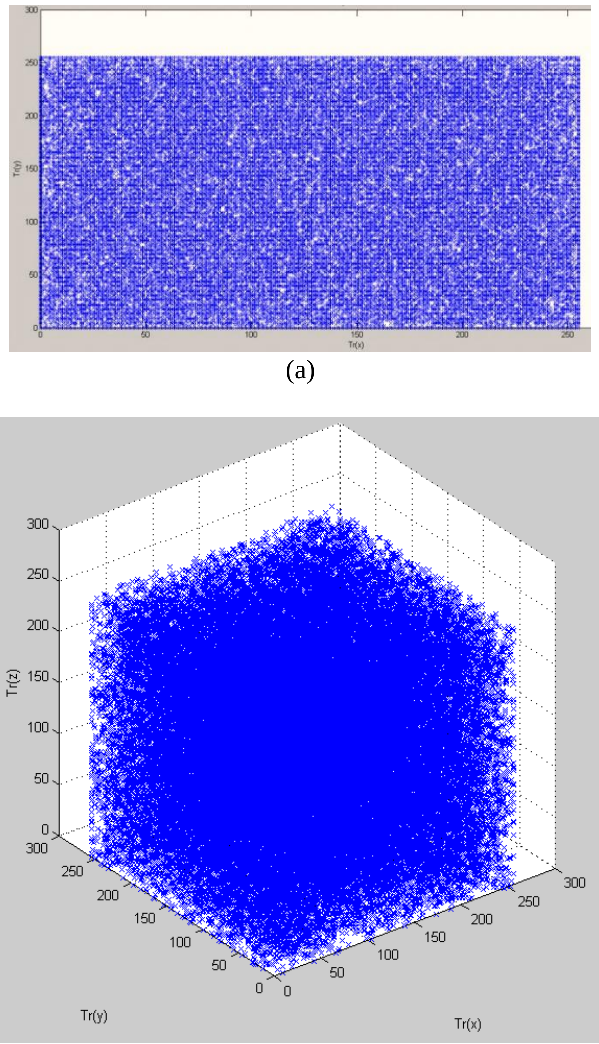
$$H(x) = -\sum_{i=1}^{n} P_i . log_2(P_i) \qquad (17)$$

$$P_i = \frac{k_i}{n} \qquad (18)$$

Table 2 shows entropy of 7.9991 bits with 99.99% of characters equiprobable, validating the proposed generator's ability to deliver random characters. In contrast, the classic generator produces non-equiprobable characters, failing this crucial test.

4.1.4. Spectral test

The spectral test holds immense significance by evaluating point distributions in 2D and 3D space and demonstrating the generator's effectiveness in discriminating between reliable and unreliable generators, as many inadequate ones fail this evaluation [30]. In Fig. 9 (a), the points exhibit a desirable uniform distribution within the square domain, indicating true randomness without patterns or clustering. The 3D representation in Fig. 9 (b) further validates this by illustrating a uniform cube distribution, eliminating higher-dimensional biases. Crucially, rotating the 3D cube confirms the absence of the Marsaglia effect. The lack of such artifacts affirms the randomness and independence of generated values, critical for cryptographic applications. As a result, the proposed generator successfully demonstrates its ability to produce random sequences suitable for security and privacy applications.



(a)

(b)

Figure. 9 Test spectral analysis: (a) 2D spectral test and (b) 3D spectral test

4.1.5. Frequency test

The frequency test is a fundamental evaluation in assessing the effectiveness of a pseudo-random generator. A high-quality pseudo-random sequence should exhibit a uniform distribution across its values, with each value occurring with equal frequency. Examining the graph in Fig. 10, it is apparent that the generated values between 0 and 255 are evenly distributed and occur with relatively equal frequencies. Consequently, the proposed generator effectively satisfies the crucial frequency test, a key requirement for producing truly random sequences suitable for various applications demanding robust randomness properties.

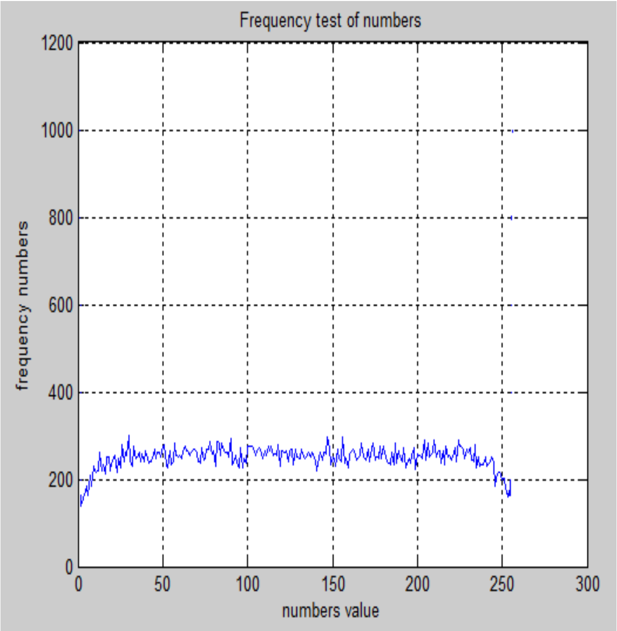
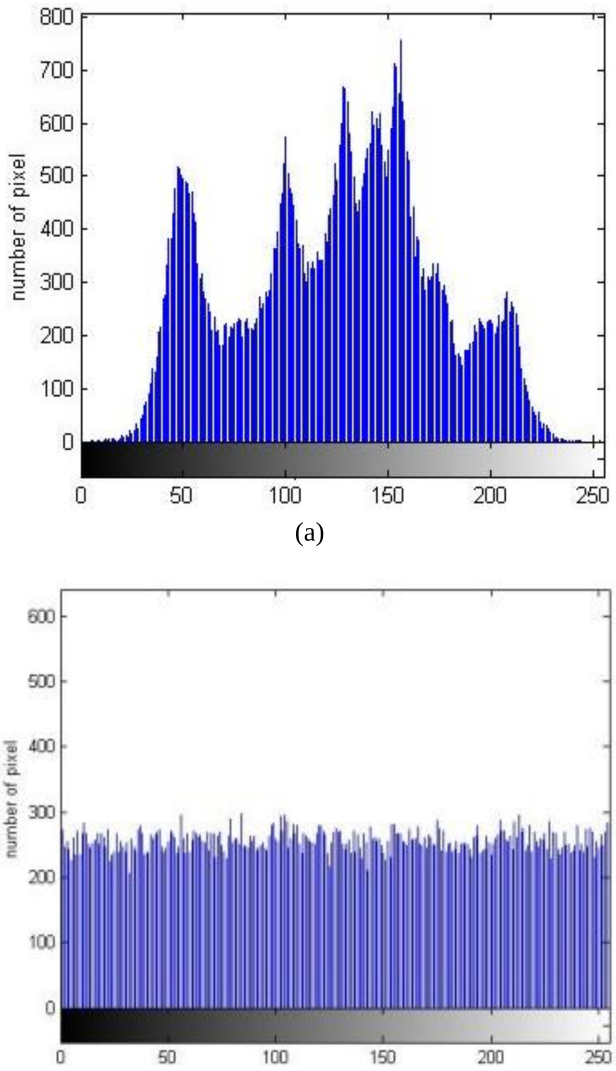


Figure. 10 The frequency test numbers

4.2 Image encryption metrics

4.2.1. Histogram analysis

The histogram test plays a crucial role in evaluating encryption strength against statistical attacks [25]. Ideally, encrypted pixel values should spread evenly over the full 0-255 range, obfuscating any patterns from the original plaintext. Fig. 11 presents the encrypted image's histogram exhibiting near-uniform distribution, signifying a balanced spread of pixel values lacking discernible biases a key requirement to thwart statistical analysis attempts. Consequently, this histogram analysis validates the proposed encryption's ability to statistically conceal all visually discernible information about the original plaintext image, underscoring its robustness against cipher text-only attacks exploiting pixel value distributions.



(a)

(b)

Figure. 11 Histogram results: (a) plain image and (b) cipher image

4.2.2. Key space analysis

In cryptography, a crucial requirement is ensuring the encryption system's key space sufficiently large to withstand brute force attacks [25], with industry standards dictating a minimum of 128 bits. In the proposed method, the secret key comprises the discrete 2D Henon map's control parameters each contributing 128 bits, collectively forming a 256-bit key space. Additionally, the CA's secret initial key adds another 256 bits. Consequently, the combined 512-bit key space employed vastly exceeds theoretical minimums, rendering exhaustive searches computationally infeasible even with foreseeable technological advancements.

4.2.3. Entropy analysis

As entropy approaches the ideal maximum of 8, it statistically indicates that encrypted images become exponentially more challenging to decode by unauthorized parties [25]. Table 3 presents the entropy values for the plain Lena image and its encrypted counterpart. The entropy value closely approximates the ideal 8, on par with state-of-the-art approaches, demonstrating the proposed scheme's ability to produce random, uniformly distributed cipher text. Also validates the method's resilience against attacks which exploit non-uniform textures to compromise security.

4.2.4. Correlation coefficient analysis

The correlation coefficient, r_{xy}, between adjacent pixels x and y is computed by the following formula:

$$r_{xy} = \frac{Con(x,y)}{\sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \frac{1}{N}\sum_{i=1}^{N} x_i))^2}\ \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \frac{1}{N}\sum_{i=1}^{N} y_i)^2}} \quad (19)$$

$$Con(x,y) = \frac{1}{N}\sum_{i=1}^{N}\left(x_i - \frac{1}{N}\sum_{i=1}^{N} x_i\right)\left(y_i - \frac{1}{N}\sum_{i=1}^{N} y_i\right) \quad (20)$$

Table 3. The entropy and Sensitivity analysis

Nature of Source	Entropy in bits/symbols	NPCR %	UACI %
Lena Plain image	7.4748	-	-
Encrypted image	7.9994	99.6768	33.7492
Ref [11]	7.99927	99.6103	33.4540
Ref [12]	7.9978	99.61	33.45
Ref [16]	7.9977	99.4386	33.3250
Ref [7]		99.4004	33.9512

Table 4. Correlation coefficients

Images	Plain image			Cipher image		
	H	V	D	H	V	D
Lena	0.9335	0.9592	0.9087	0.0019	0.0027	0.0020
Ref [7]	0.9715	0.9751	0.9534	0.0099	0.0144	0.0151
Ref [11]	0.9850	0.9782	0.9633	0.0065	0.0051	−0.0005
Ref [12]	0.9588	0.9260	0.9291	0.0008	0.0031	−0.0058
Ref [16]	-	-	-	0.0012	0.0025	0.0008

Table 4 illustrates that proposed scheme reducing the correlation among pixels in the encrypted image to near-zero values in all directions. That demonstrates the algorithm’s effectiveness in randomizing the cipher image and eliminating any exploitable statistical relationships between neighboring pixels. Such a lack of correlation is crucial for withstanding statistical and structural attacks that leverage pixel redundancies.

4.2.5. Differential attack analysis

The proposed encryption algorithm exhibits a good property of high sensitivity to minute changes in the original image and encryption keys. This sensitivity is evaluated through two criteria: NPCR (Number of Pixels Change Rate) which measures the percentage of different pixels between two cipher images, and UACI (Unified Average Changing Intensity) which quantifies the average intensity difference. NPCR and UACI are defined by Eqs. (21)-(23), respectively.

$$NPCR = \frac{\sum_{i,j} D(i,j)}{M \times N} \times 100 \tag{21}$$

$$D(i,j) = \begin{cases} 0 \ \ if \ C1(i,j) = C2(i,j) \\ 1 \ if \ \ C1(i,j) \neq C2(i,j) \end{cases} \tag{22}$$

$$UACI = \frac{1}{M \times N}\left[\sum_{i,j} \frac{C1(i,j) - C2(i,j)}{255}\right] \times 100 \tag{23}$$

The results in Table 3 demonstrate that a single-pixel alteration in the plaintext image leads to substantial changes in the ciphertext, with NPCR and UACI values close to their ideal values and comparable to other state-of-the-art encryption schemes reported in the literature. This sensitivity ensures the algorithm’s robustness against various cryptanalytic attacks.

5. Conclusion

This paper presented a new pseudo-random sequence generator that integrates Hénon’s 2D chaotic map into the evolutionary process of Elementary Cellular Automata (ECA). By introducing dynamism into selected neighborhoods during evolution, the proposed generator exhibits improved efficiency and robust random properties compared to conventional static neighborhood-based generators in Cellular Automata (CA) evolution. The experimental analysis demonstrated the chaotic attributes of the generator, meeting all criteria defined by the NIST test. The new generator was successfully applied to the One-Time Pad (OTP) encryption algorithm for image data. Encrypted image achieved a near-maximum entropy value of 7.9994 bits per symbol, ultra-low correlation coefficients close to zero, and desirable Number of Pixels Change Rate (NPCR) and Unified Average Changing Intensity (UACI) values close to ideal thresholds underline the robustness and suitability of the encryption system for images. This work represents a scientific breakthrough by introducing an innovative evolution CA method to generate good pseudo-random sequences perfectly integrated with chaos-based encryption.

Conflicts of Interest

The authors declare no conflict of interest.

Author Contributions

Conceptualization, methodology, software, validation, formal analysis, investigation, resources, data curation, writing—original draft preparation, writing—review and editing, visualization, the first and second authors; supervision, project administration, the third author.

References

[1] A. Setyono and D. Setiadi, “An Image Watermarking Method Using Discrete Tchebichef Transform and Singular Value Decomposition Based on Chaos Embedding”, *International Journal of Intelligent Engineering and Systems*, Vol. 13, No. 2, pp. 140-150, 2020, doi: 10.22266/ijies2020.0430.14.

[2] J. Zeng and C. Wang, “A Novel Hyperchaotic Image Encryption System Based on Particle Swarm Optimization Algorithm and Cellular

Automata", *Security and Communication Networks*, Vol. 2021, pp. 1-15, 2021.

[3] W. Zhang, Z. Zhu, and H. Yu, "A Symmetric Image Encryption Algorithm Based on a Coupled Logistic-Bernoulli Map and Cellular Automata Diffusion Strategy", *Entropy*, Vol. 21, No. 5, pp. 504, 2019.

[4] A. Pourjabbar Kari, A. Habibizad Navin, A. M. Bidgoli, and M. Mirnia, "A novel multi-image cryptosystem based on weighted plain images and using combined chaotic maps", *Multimedia Systems*, Vol. 27, No. 5, pp. 907-925, 2021.

[5] J. Liu, Y. Wang, Q. Han, and J. Gao, "A Sensitive Image Encryption Algorithm Based on a Higher-Dimensional Chaotic Map and Steganography", *International Journal of Bifurcation and Chaos*, Vol. 32, No. 1, 2022.

[6] R. V. Ravi, S. B. Goyal, S. M. N. Islam, V. Kumar, "Color Image Encryption Employing Cellular Automata and Three-Dimensional Chaotic Transforms", In: *Proc. of International Conf. On Innovative Technologies in Intelligent Systems and Industrial Applications*, Kuala Lumpur, Malaysia, and Sydney, Australia, pp. 473-481, 2023.

[7] M. kordestani, "An Image Cryptosystem based on Elementary Cellular Automata with Integrity Checking", *Intelligent Knowledge Exploration and Processing*, Vol. 3, No. 11, 2024.

[8] H. Ali, M. S. Khan, M. Driss, J. Ahmad, W. Buchanan, and N. Pitropakis, "CellSecure: Securing Image Data in Industrial Internet-of-Things via Cellular Automata and Chaos-Based Encryption", In: *Proc. of International IEEE 98th Vehicular Technology Conference (VTC2023-Fall)*, Hong Kong, Hong Kong, pp. 1-6, 2023.

[9] A. Mazen, Y. Korayem, M. Gabr, and W. Alexan, "Three Layered Image Encryption: An Application of Hyperchaos and Cellular Automata", In: *Proc. of International Telecommunications Conference (ITC-Egypt)*, Alexandria, Egypt, pp. 615-620, 2023.

[10] A. Kumar and N. S. Raghava, "An efficient image encryption scheme using elementary cellular automata with novel permutation box", *Multimedia Tools and Applications*, Vol. 80, No. 14, pp. 21727-21750, 2021.

[11] L. Li, Y. Luo, S. Qiu, X. Ouyang, L. Cao, and S. Tang, "Image encryption using chaotic map and cellular automata", *Multimedia Tools and Applications*, Vol. 81, No. 28, pp. 40755-40773, 2022.

[12] X. Ma and C. Wang, "Hyper-chaotic image encryption system based on $\mathrm{N}+2$ ring Joseph algorithm and reversible cellular automata", *Multimedia Tools and Applications*, Vol. 82, No. 25, pp. 38967-38992, 2023.

[13] S. Khaled, M. Gabr, Y. Korayem, and W. Alexan, "Image Encryption Through Cellular Automata, S-Box and Tent Chaotic Map", In: *Proc. of 2023 International Conf. On Telecommunications (ITC-Egypt)*, pp. 601-606, 2023.

[14] K. Kumar, S. Roy, U. Rawat, and A. Shandilya, "SOCIET: Second-order cellular automata and chaotic map-based hybrid image encryption technique", *Multimedia Tools and Applications*, Vol. 83, No. 10, pp. 29455-29484, 2023.

[15] A. Boudali, N. H. Said, and A. Ali-Pacha, "A new symmetrical cryptosystem based cellular automata and chaotic map function", *Journal of Discrete Mathematical Sciences and Cryptography*, Vol. 25, No. 5, pp. 1435-1455, 2022.

[16] B. Rezaei, H. Ghanbari, and R. Enayatifar, "An image encryption approach using tuned Henon chaotic map and evolutionary algorithm", *Nonlinear Dynamics*, Vol. 111, No. 10, pp. 9629-9647, 2023.

[17] S. Jamil, N.T. Hiramony, T. Tahreen Tanisha, and R. Chakraborty, "Circuit Implementation of Vernam Cipher-based Data Encryption Using Cellular Automata", In: *Proc. of 25th International Conf. On Computer and Information Technology (ICCIT)*, pp. 763-767, 2022.

[18] E. Formenti, K. Imai, B. Martin, and J. Yunès, "Advances on Random Sequence Generation by Uniform Cellular Automata", In: *Proc. of Computing with New Resources. Lecture Notes in Computer Science*, Vol. 8808, Springer, Cham, 2014.

[19] S. Bilan, M. Bilan, and S. Bilan, "Research of the method of pseudo-random number generation based on asynchronous cellular automata with several active cells", In: *Proc. of 21st International Conf. on Circuits, Systems, Communications and Computers*, Heraklion, Crete, Grece, pp. 14-17, 2017.

[20] U. S. Choi, H. D. Kim, S. W. Kang, and S. J. Cho, "Design of Key Sequence Generators Based on Symmetric 1-D 5-Neighborhood CA", *The Journal of the Korea institute of electronic communication sciences*, Vol. 16, No. 3, pp. 533-540, 2021.

[21] S. Wolfram, "Random sequence generation by cellular automata", *Advances in Applied Mathematics*, Vol. 7, No. 2, pp. 123-169, 1986.

[22] S. Wolfram, “Statistical mechanics of cellular automata”, *Reviews of Modern Physics*, Vol. 55, No. 3, pp. 601-644, 1983.

[23] R. White and G. Engelen, “Cellular Automata and Fractal Urban Form: A Cellular Modelling Approach to the Evolution of Urban Land-Use Patterns”, *Environment and Planning A: Economy and Space*, Vol. 25, No. 8, pp. 1175-1199, 1993.

[24] A. Kanshi, R. Soundrapandiyan, V. S. Anita Sofia, and V. R. Rajasekar, “Hybridized Cryptographic Encryption and Decryption Using Advanced Encryption Standard and Data Encryption Standard”, *Cybernetics and Information Technologies*, Vol. 23, No. 4, pp. 63-78, 2023.

[25] X. Gao, M. Miao, and X. Chen, “Multi-Image Encryption Algorithm for 2D and 3D Images Based on Chaotic System”, *Frontiers in Physics*, Vol. 10, 2022.

[26] Y. Nakar, and D. Ron, “The Structure of Configurations in One-Dimensional Majority Cellular Automata: From Cell Stability to Configuration Periodicity”, In: *Proc. of International Conf. on Cellular Automata for Research and Industry*, Geneva, Switzeland, pp 63-72, 2022.

[27] P. Parida *et al.*, “Elliptic curve cryptographic image encryption using Henon map and Hopfield chaotic neural network”, *Multimedia Tools and Applications*, Vol. 82, No. 22, pp. 33637-33662, 2023.

[28] A. Rukhin, J. Soto, J. Nechvatal, M. Smid, E. Barker, S. Leigh, S. Leigh, M. Levenson, M. Vangel, D. Banks, N. Heckert, J. Dray, S. Vo, and L. Bassham, “Nist special publication 800-22: A statistical test suite for the validation of random number generators and pseudo random number generators for cryptographic applications”, *NIST Special Publication*, Vol. 800, No. 22, 2010.

[29] C. E. Shannon, “A Mathematical Theory of Communication”, *Bell System Technical Journal*, Vol. 27, No. 4, pp. 623-656, 1948.