A Novel Piecewise Chaotic Map for Image Encryption

Nikolaos Charalampidis*, Christos Volos*, Lazaros Moysis*, Hector E. Nistazakis†, Ioannis Stouboulos*
* Laboratory of Nonlinear Systems - Circuits & Complexity, Physics Department, Aristotle University of Thessaloniki, Thessaloniki, Greece. {nicharala, volos, lmousis, stouboulos}@physics.auth.gr
† Section of Electronic Physics and Systems, Physics Deparment, National and Kapodistrian University of Athens, Athens, Greece. enistaz@phys.uoa.gr

***Abstract*—This paper investigates the issue of chaos-based image encryption. A new one-dimensional piecewise chaotic map based on the *z*-shaped fuzzy number is proposed and investigated, exhibiting regions of constant chaos and high Lyapunov exponent values. A pseudorandom bit generator based on the novel chaotic map is designed and successfully passes the National Institute of Standards and Technology (NIST) statistical tests. This PRBG is applied to the problem of image encryption, introducing a new image encryption scheme by using row and column permutation, chaotic pixel shuffling, and the exclusive OR operation. To demonstrate the robustness of this image encryption technique, various tests, such as differential attack and correlation, entropy and histogram analysis are performed.**

***Index Terms*—Fuzzy Numbers, Chaos, Cryptography, Image Encryption, Random Bit Generator**

I. Introduction

In recent years, chaos-based cryptography undeniably has started being a prominent member in the discipline of information security, over classic encryption algorithms like DES, AES, RSA [1], with wide-ranging applicability [2] [3].

The determinism, the dubious behaviour, the low computational cost of chaotic systems, renders chaos-based cryptography a promising field. Therefore, chaotic systems provide an excellent foundation for implementing a plethora of encryption schemes. Furthermore, the design of new chaotic maps, such as discrete maps of low dimension that display regions of constant chaotic behavior [4] [5], is a key feature in this field.

Motivated by this, and the research approach of [6], a new piecewise chaotic map [7] is proposed based on the *z*-shaped fuzzy number [8]. The proposed map has one basic feature that is not usually found on chaotic maps in conjunction with fuzzy numbers (see the well-known tent map [9]), which makes it a good candidate for cryptography-related applications. This is the fact that this map has large areas of constant chaos with high Lyapunov exponent values. Its bifurcation diagrams and corresponding Lyapunov exponent diagrams are computed to investigate this.

The proposed map is then used to build a pseudo-random bit generator (PRBG) [10] [11]. The goal is to generate a stream of bits that has properties similar to a random series. The National Institute of Standards and Technology (NIST) statistical test suite is used to validate the randomness of this bitstream [12].

Next, the proposed PRBG is applied to the problem of image encryption [13], which is based on the following three parts:

1) Initially, with the help of the proposed map a row and column shuffling is performed to the original image, and the shuffled image is partitioned to blocks [14].
2) Then, with the use of the proposed map a pixel shuffling is conducted on each block, by adapting the map's initial conditions [15], which yields a new shuffled image. After that, its pixels are converted into 8bits.
3) Lastly, the obtained sequence is combined with the pseudo-random bit generator through the exclusive OR operation to acquire the encrypted image.

This image encryption technique's security is then evaluated using tests such as entropy, correlation, and histogram analysis.

The outline of this work is as follows. The proposed chaotic map and its dynamical behavior are presented in Section 2. Section 3 depicts the proposed chaotic map-based pseudo-random bit generator and its statistical tests. The application of image encryption is outlined in Section 4. Section 5 contains the conclusions as well as suggestions for future work.

II. The Proposed Chaotic Map

Consider the z-shaped fuzzy number given by,

$$f(x) = \begin{cases} 1, & x \leq a \\ 1 - 2(\frac{x-a}{b-a})^2, & a \leq x \leq \frac{a+b}{2} \\ 2(\frac{x-b}{b-a})^2, & \frac{a+b}{2} \leq x \leq b \\ 0, & x \geq b \end{cases} \tag{1}$$

and let $a = 0$, then the proposed map is of the following form,

$$h(x) = \mod\left(b\left(\frac{\pi - \cos(\eta(x^3 + zx))}{\pi}\right)^3 \cos\left(\pi \arccos\left(\pi x\right)\right),\ b\right) \tag{2}$$

$$k(x) = \mod\left(b \cos\left(\frac{\eta}{x^z}\right)\left(z(x^3 + x) + \eta\right), b\right) \tag{3}$$

$$f_1(x) = \begin{cases} 1 - 2\left(\dfrac{h(x)}{b}\right)^2, & \text{if } 0 \leq h(x) \leq \dfrac{b}{2} \\ 2\left(\dfrac{h(x) - b}{b}\right)^2, & \text{if } \dfrac{b}{2} \leq h(x) \leq b \end{cases} \tag{4}$$

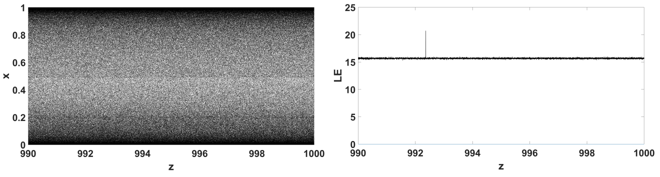
$$f_2(x) = \begin{cases} 1 - 2\left(\dfrac{k(x)}{b}\right)^2, & \text{if } 0 \leq k(x) \leq \dfrac{b}{2} \\ 2\left(\dfrac{k(x) - b}{b}\right)^2, & \text{if } \dfrac{b}{2} \leq k(x) \leq b \end{cases} \tag{5}$$

$$x_i = \begin{cases} f_1(x_{i-1}) & \text{, if } x_{i-1} \leq 0.5 \\ f_2(x_{i-1}) & \text{, otherwise} \end{cases} \tag{6}$$

where the parameters $(\eta,\ z,\ b) \in \mathbb{R}^+$ are used to control the behavior of the system.

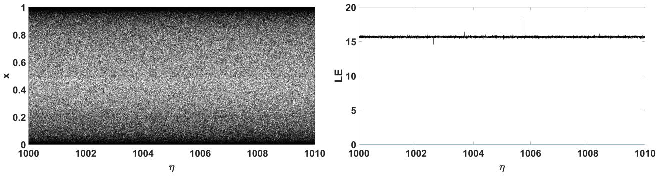
To investigate the system's dynamical behavior, bifurcation diagrams and their corresponding Lyapunov exponent diagrams are plotted with respect to parameters η, z, and b, with initial condition set to $x_0 = 0.1$.

The bifurcation diagrams with respect to parameters z, η are depicted in Figs. 1(a), and 2(a). It can be seen that large regions of constant chaos are present in both bifurcation diagrams. Furthermore, the aforementioned behavior can be verified in Figs. 1(b), and 2(b), which present the Lyapunov exponent diagrams with respect to parameters z, and η, respectively. The maximal Lyapunov exponents in these cases are 20.71, and 18.32. Note that the same behavior, of constant chaos is also present with respect to parameter b.



(a) (b)

Fig. 1. (a) The bifurcation diagram, and (b) its corresponding Lyapunov exponent diagram with respect to z, for $\eta = 1005$ and $b = 2$.



(a) (b)

Fig. 2. (a) The bifurcation diagram, and (b) its corresponding Lyapunov exponent diagram with respect to η, for $z = 990$ and $b = 2$.

III. Pseudo Random Bit Generator

The construction of a pseudo-random bit generator based on the chaotic map (6) will be presented in this Section.

Consider $x,\ v$ two sequences with initial conditions x_0, v_0 and length m of the map (6), and let the pairs (η_x, z_x, b_x) and (η_x, z_x, b_x) be their corresponding parameters, then the random bit generator takes the form [10],

$$p_i = \mod (10^{10} x_i, 1) \tag{7}$$

$$q_i = \mod (10^{10} v_i, 1) \tag{8}$$

$$pbit_i = \begin{cases} 1, & \text{if } p_i \geq 0.5 \\ 0, & \text{otherwise} \end{cases} \tag{9}$$

$$qbit_i = \begin{cases} 1, & \text{if } q_i \geq 0.5 \\ 0, & \text{otherwise} \end{cases} \tag{10}$$

$$bit_i = xor(pbits_i,\ qbits_i) \tag{11}$$

with the resulting bitstream, $\mathcal{B} = \{bit_1,\ bit_2,\ \ldots,\ bit_m\}$.

To assess the randomness of the suggested technique, a set of $100 \cdot 10^6$ bits is generated and tested using the National Institute of Standards and Technology's (NIST) Federal Information Processing Standards (FIPS) tests. This package contains a collection of 15 statistical tests, each of which is used to validate the randomness of a bitstream. To be considered successful, the computed *P*-value in each of the tests must be greater than the significance level, which is set by default at 0.01. If and only if all tests are successful, a generator is considered random. Table I shows the results for the parameters $(\eta_x, z_x, b_x) = (1005, 990, 2)$ and $(\eta_y, z_y, b_y) = (1008, 997, 8)$, as well as the initial conditions $x_0 = 0.111$ and $v_0 = 0.777$, where it can be seen that the generator passes all tests and can thus be used for encryption-related schemes.

TABLE I
NIST Test Results

No.	*Test*	*Chi-square, P-value*	*Rate*
1	Frequency	0.883171	99/100
2	Block Frequency	0.115387	100/100
3	Cumulative Sums	0.162606	99/100
4	Runs	0.834308	100/100
5	Longest Run	0.026948	99/100
6	Rank	0.678686	100/100
7	Fast Fourier Transform	0.090936	100/100
8	NonOverlapping Template	0.023545	100/100
9	Overlapping Template	0.699313	98/100
10	Universal	0.798139	99/100
11	Approximate Entropy	0.383827	100/100
12	Random Excursions	0.262249	58/58
13	Random Excursions Variant	0.013569	58/58
14	Serial	0.834308	99/100
15	Linear Complexity	0.334538	100/100

IV. The Proposed Image-Encryption Scheme

In this Section, the proposed pseudo-random bit generator is applied to the problem of image encryption. In addition, various statistical tests are run to assess the security of the proposed image-encryption.

A. Encryption & Decryption Processes

The encryption consists of three parts, a row and columns shuffling, a pixel shuffling on each block of a partition, and an application of the PRBG followed by an exclusive OR operation.

Step 1: Consider an $n \times m$ image source $\mathcal{A}$ and perform row and column shuffling to it. To do that the $n \times n$ permutation matrix $\mathcal{L}$ for the rows, and $m \times m$ permutation matrix $\mathcal{R}$ for the columns, are constructed. Consider two maps q and w of (6) with initial conditions $q_0 = \mod (10^2 H(\mathcal{A})^{-1}, 1)$ and $w_0 = \mod (10^6 H(\mathcal{A})^{-1}, 1)$ of length n and m, and control parameters (η_q, z_q, b_q) and (η_w, z_w, b_w) respectively, where $H(\mathcal{A})$ is the entropy of image $\mathcal{A}$. Then, consider two $1 \times n$ and $1 \times m$ vectors L and R which the elements fall within the integer interval $[1, n]$ and $[1, m]$, and the following rules generate each element $l_i = \lceil \mod (10^8 q_i, n) \rceil$ and $r_i = \lceil \mod (10^8 w_i, m) \rceil$ respectively. Note, that $\lceil \cdot \rceil$ is the ceiling operator. If an integer $l_i = l_j$ with $i \neq j$ for $i, j = 1, \ldots, n$ then, l_j is removed until all positions in vector L are uniquely defined. The same holds true for vector R. Then L gives the permutation order for the rows and R gives the permutation order for the columns. The permutation matrices $\mathcal{L}$ and $\mathcal{R}$ are then obtained by $\mathcal{L}(i, l_1) = 1$, $\mathcal{R}(r_1, i) = 1$ and zeros elsewhere. For example, if $l_1 = 100$, meaning that $\mathcal{L}(1, l_1) = 1$, then, by pre-multiplying $\mathcal{L}$ to $\mathcal{A}$, the 100th row of $\mathcal{A}$ will move to 1st row. Alternatively, if $r_1 = 10$, i.e. $\mathcal{R}(r_1, 1) = 1$, then, by post-multiplying $\mathcal{R}$ to $\mathcal{A}$, the 10th column of $\mathcal{A}$ will move to the 1st column. After, a pre-multiplication of $\mathcal{L}$ and post-multiplication of $\mathcal{R}$ with $\mathcal{A}$ yields a permuted image $\bar{\mathcal{A}} = \mathcal{L}\mathcal{A}\mathcal{R}$. The Fig. 3(b) is an example of how $\bar{\mathcal{A}}$ looks like. The shuffled image $\bar{\mathcal{A}}$ is then partitioned into c block matrices $\bar{A}_i$ of dimensions $n_i \times m_i$ for $i = 1, \ldots, c$, where each $n_i \in [1, n]$ and $m_i \in [1, m]$. For example, let $c = 1024$ and $\bar{A}_i$ be of dimensions $n_i \times m_i = 16 \times 16$ for all $i = 1, \ldots, c$.

Step 2: Pixel shuffling is performed in each block matrix $\bar{A}_i$, $i = 1, \ldots, c$ separately. In order to do that, reshape a block matrix in a $1 \times n_i m_i$ vector V_i, and consider a sequence y of map (6) with control parameters (η_y, z_y, b_y) and initial conditions $y_0 = \mod \left(\left(\frac{9i}{n_i m_i} H(\mathcal{A}) + \ln i\right), 1\right)$. Then, a $1 \times n_i m_i$ vector S_i of integers in the interval $[1, n_i m_i]$ is produced by $s_j^i = \lceil \mod (10^8 y_j, n_i m_i)\rceil$. Note, that no two elements in S_i can have the same value in the interval $[1, n_i m_i]$. Then, S_i gives the permutation order of the vector V_i, which yield a new vector P_i. Then, P_i is reshaped into an $n_i \times m_i$ matrix $\widetilde{A}_i$. This is repeated until all block matrices have been shuffled, and a new permuted image $\widetilde{\mathcal{A}}$ is obtained. An example it can be seen in Fig. 3(c). The matrix $\widetilde{\mathcal{A}}$ is converted into a binary sequence $\widetilde{B}$, where all the pixels of $\widetilde{\mathcal{A}}$ are transformed to 8 bit sequences.

Step 3: Then, the sequence $\widetilde{B}$ is combined with a sequence B, of same dimensions, obtained by the PRBG, with initial conditions $x_0 = \mod (10^8 H(\mathcal{A})^{-1}, 1)$ and $v_0 = \mod (10^{10} H(\mathcal{A})^{-1}, 1)$, through an exclusive OR operation. This combination yields $\widetilde{C} = \widetilde{B} \oplus B$. The binary sequence $\widetilde{C}$ is converted into integer form and reshaped into $n \times m$ matrix, and the final encrypted image $\mathcal{C}$ is procured, as in Fig. 3(d).

The encrypted image can be transmitted along with the encryption key $\{x_0, v_0, y_0, q_0, w_0, \eta_x, z_x, b_x, \eta_v, z_v, b_v, \eta_y, z_y, b_y, \eta_q, z_q, b_q, \eta_w, z_w, b_w\}$. The moment the receiver has the encrypted image and the encryption key the decryption can start by reversing the aforementioned stages. First, by using the PRBG and solving $\widetilde{B} = \widetilde{C} \oplus B$. Second, by transforming the binary sequence $\widetilde{B}$ in double form to obtain $\widetilde{\mathcal{A}}$, computing $H(\widetilde{\mathcal{A}}) = H(\mathcal{A})$, splitting $\widetilde{\mathcal{A}}$ into c blocks of $n_i \times m_i$ dimensions and reversing the shuffling in each block, changing the initial condition y_0 for each block according to Step 2. This will yield $\bar{\mathcal{A}}$. Then, computing the permutation matrices $\mathcal{L}$ and $\mathcal{R}$ and reversing the row and column shuffling by pre and post multiplying with the inverse permutation matrices as $\mathcal{L}^{-1}\bar{\mathcal{A}}\mathcal{R}^{-1} = \mathcal{A}$, which is equal to the original image.

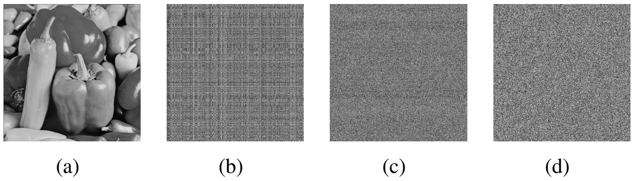

(a) (b) (c) (d)

Fig. 3. (a) The Original Image, (b) the Image after row and column shuffling, (c) the Image after adaptive shuffling on Fig.3 (b), and (d) the Encrypted Image.

B. Security Analysis

In order to showcase the proposed design a 512×512 image $\mathcal{A}$, depicted in Fig. 3(a), is considered. The initial conditions are defined as in Steps 1, 2, and 3 and triplets of parameters $(\eta_q, z_q, b_q) = (1003, 994, 5)$, $(\eta_w, z_w, b_w) = (1010, 999, 7)$, $(\eta_y, z_y, b_y) = (1002, 993, 10)$, $(\eta_x, z_x, b_x) = (1005, 990, 2)$, $(\eta_v, z_v, b_v) = (1008, 997, 8)$.

To evaluate the security of the encryption scheme, several tests are performed in the encrypted image.

1) *Histograms*: The encrypted and original image histograms are computed and shown in Fig. 4. In contrast to the original image, the encrypted one exhibits uniformity.

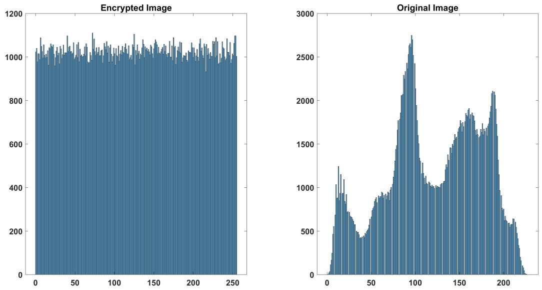


Fig. 4. Histograms of the Encrypted Image and the Original Image.

2) *Correlation Coefficient*: The correlation coefficient for neighboring characters in an encrypted image should be close to zero, indicating that their values are uncorrelated, and is calculated as in [6]. The diagonal correlation coefficient for the original, after the 1st shuffling, after the 2nd shufflling, and also for the encrypted image are 0.9639, -0.0033, -0.0012, and 0.0001 respectively. When compared to the original, the encrypted image has substantially lower correlation coefficients. As a result, the permutation and encryption of the image allow for the reduction of correlation, which is the desired result.

3) *Information Entropy*: A signal's global information entropy is a measure of its randomness and unpredictability. An image's information entropy is calculated as in [6]. Because the image is encrypted with 2^8 characters, it is deduced that the diffusion is good if the entropy is near or equal to 8. Computing the entropies for the original image, the 1st shuffling, the 2nd shuffling, and the encrypted image to be 7.5937, 7.5937, 7.5937, and 7.9993 respectively. So, it is observed that the entropies between the original image and the shuffled is identical, because no encryption is taking place, while the entropy of the encrypted image is closing to the maximum value the entropy can have in this scenario.

4) ***Differential Attack Analysis***: A minor change to the original image in an encryption scheme should result in a completely different encrypted. This makes the design resilient to known plaintext attacks, in which an attacker encrypts the same plaintext repeatedly, slightly changing it each time, and examining changes in the encrypted image to expose the encryption structure. The number of pixels change rate (NPCR) and the unified average changing intensity (UACI) are computed to determine the level of change between nearly identical images. To accomplish this, two identical images $\mathcal{A}_1$ and $\mathcal{A}_2$ are considered, with the second different from the first by only one pixel. According to [16] the optimum values are calculated to be 99.61% for the NPCR and 33.46% for the UACI. Because in the proposed scheme the initial conditions of the map (6) are plaintext dependent, a single change in the image will result in a different permuted and encrypted image. The corresponding encrypted images for the $\mathcal{A}_1$ and $\mathcal{A}_2$ images are generated, and the NPCR and UACI are computed to be 99.62 and 33.24 respectively [16]. Therefore, it becomes clear that the encrypted image performs well on both measures, as the outcomes are close to the ideal values.

5) ***Key Space & Sensitivity***: Every encryption system must be resistant to brute force attacks. As a result, the key space must be greater than 2^{100} [17]. In this image encryption scheme, six chaotic maps (6) are utilized, with initial conditions x_0, v_0, y_0, q_0, w_0 and control parameters $\eta_x, z_x, b_x, \eta_v, z_v, b_v, \eta_y, z_y, b_y, \eta_q, z_q, b_q, \eta_w, z_w, b_w$. Note, that only the first y_0 is included in the key space, the rest will be computed by the receiver as the decryption of the image is progressing. Then, the upper bound for the key space is computed as $10^{20\cdot 16} = 10^{320} = (10^2)^{160} \approx (2^7)^{160} = 2^{1120}$, which already is greater than the required minimum key space.

V. Conclusions

A new one-dimensional piecewise chaotic map is introduced in this work. This map demonstrates large regions of constant chaos and high Lyapunov exponent values. Furthermore, a PRBG on this map was presented, which was tested and passed all NIST tests. After that, an image encryption system was created. The system is built around a shuffling process that is derived from the proposed chaotic map. This is followed by the use of the PRBG and an XOR operation, which results in the encrypted image. Finally, a variety of tests and measures were used to demonstrate that the resulting encrypted image was random and secure. Future extensions of this work will include full analysis on the proposed map with Lyapunov exponent diagrams with respect to two parameters and phase diagrams. In addition, to robustify and generalize this encryption scheme the partitioning in Step 1 and the dimensions of each block will be chosen through a chaotic process. Furthermore, with the advancement of multimedia applications, the use of 3D objects is becoming increasingly popular, and its security has become an immediate issue [18]. As such, extensions to this work will also attempt to contribute in this direction.

Acknowledgement

The authors acknowledge funding from National and Kapodistrian University of Athens (NKUA), Special Account for Research Grants (SARG).

References

[1] D. Xiao, X. Liao, and K. Wong, "An efficient entire chaos-based scheme for deniable authentication," *Chaos, Solitons & Fractals*, vol. 23, no. 4, pp. 1327–1331, 2005.

[2] L. Moysis, I. Kafetzis, C. Volos, A. V. Tutueva, and D. Butusov, "Application of a hyperbolic tangent chaotic map to random bit generation and image encryption," in *2021 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (ElConRus)*. IEEE, 2021, pp. 559–565.

[3] X. Tang and S. Mandal, "Encrypted physical layer communications using synchronized hyperchaotic maps," *IEEE Access*, vol. 9, pp. 13 286–13 303, 2021.

[4] Z. Hua, B. Zhou, and Y. Zhou, "Sine chaotification model for enhancing chaos and its hardware implementation," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 2, pp. 1273–1284, 2018.

[5] L. Liu and S. Miao, "A new simple one-dimensional chaotic map and its application for image encryption," *Multimedia Tools and Applications*, vol. 77, no. 16, pp. 21 445–21 462, 2018.

[6] L. Moysis, C. Volos, S. Jafari, J. M. Munoz-Pacheco, J. Kengne, K. Rajagopal, and I. Stouboulos, "Modification of the logistic map using fuzzy numbers with application to pseudorandom number generation and image encryption," *Entropy*, vol. 22, no. 4, p. 474, 2020.

[7] H. Zang, Y. Yuan, and X. Wei, "Research on pseudorandom number generator based on several new types of piecewise chaotic maps," *Mathematical Problems in Engineering*, vol. 2021, 2021.

[8] M. Barman and J. P. Chaudhury, "A framework for selection of membership function using fuzzy rule base system for the diagnosis of heart disease," *International Journal of Information Technology and Computer Science*, vol. 5, no. 11, pp. 62–70, 2013.

[9] S. Zhu, X. Deng, W. Zhang, and C. Zhu, "A new one-dimensional compound chaotic system and its application in high-speed image encryption," *Applied Sciences*, vol. 11, no. 23, p. 11206, 2021.

[10] A. V. Tutueva, E. G. Nepomuceno, A. I. Karimov, V. S. Andreev, and D. N. Butusov, "Adaptive chaotic maps and their application to pseudorandom numbers generation," *Chaos, Solitons & Fractals*, vol. 133, p. 109615, 2020.

[11] P. Ayubi, S. Setayeshi, and A. M. Rahmani, "Deterministic chaos game: a new fractal based pseudo-random number generator and its cryptographic application," *Journal of Information Security and Applications*, vol. 52, p. 102472, 2020.

[12] A. Rukhin, J. Soto, J. Nechvatal, M. Smid, and E. Barker, "A statistical test suite for random and pseudorandom number generators for cryptographic applications," Booz-allen and hamilton inc mclean va, Tech. Rep., 2001.

[13] X. Liu, X. Tong, Z. Wang, and M. Zhang, "A new n-dimensional conservative chaos based on generalized hamiltonian system and its' applications in image encryption," *Chaos, Solitons & Fractals*, vol. 154, p. 111693, 2022.

[14] X. Chen and C.-J. Hu, "Adaptive medical image encryption algorithm based on multiple chaotic mapping," *Saudi journal of biological sciences*, vol. 24, no. 8, pp. 1821–1827, 2017.

[15] X. Huang and G. Ye, "An efficient self-adaptive model for chaotic image encryption algorithm," *Communications in Nonlinear Science and Numerical Simulation*, vol. 19, no. 12, pp. 4094–4104, 2014.

[16] Y. Wu, J. P. Noonan, S. Agaian *et al.*, "Npcr and uaci randomness tests for image encryption," *Cyber journals: multidisciplinary journals in science and technology, Journal of Selected Areas in Telecommunications (JSAT)*, vol. 1, no. 2, pp. 31–38, 2011.

[17] G. Alvarez and S. Li, "Some basic cryptographic requirements for chaos-based cryptosystems," *International journal of bifurcation and chaos*, vol. 16, no. 08, pp. 2129–2151, 2006.

[18] X. Wang, M. Xu, and Y. Li, "Fast encryption scheme for 3d models based on chaos system," *Multimedia Tools and Applications*, vol. 78, no. 23, pp. 33 865–33 884, 2019.