2014 IEEE International Conference on Systems, Man, and Cybernetics
October 5-8, 2014, San Diego, CA, USA

Image encryption using 2D Logistic-Sine chaotic map

Zhongyun Hua, Yicong Zhou*, Chi-Man Pun, C. L. Philip Chen
Department of Computer and Information Science
University of Macau, Macau, China 999078
Email: * yicongzhou@umac.mo

***Abstract*—This paper introduces a new two-dimensional Logistic-Sine map (2D-LSM). It has excellent chaotic performance and its outputs are difficult to predict. Using 2D-LSM, this paper proposes a new image encryption algorithm. Simulation results and security analysis demonstrate that the proposed algorithm is able to protect different kinds of images with a high security level.**

Keywords—chaotic map, cascade chaotic system, pseudo-random number generator (PRNG), data encryption.

I. INTRODUCTION

With the fast development of the computer network and digital technology, digital images become more and more important information carriers in recent years. So image security attracts increasing researchers' attentions. The image encryption is a straightforward technology of image security that can transform a digital image into an unrecognized noise image.

In the research of image encryption technologies, one feasible way is to treat digital image as a binary stream and then use the data encryption technology to encrypt it. But a digital image with pixels usually represented as a few binary bits, high correlations and data redundancy exist between neighboring pixels. This results in the fact that traditional digital data encryption algorithms may not be suitable for image encryption. Thus, many image encryption algorithms considering the properties of image were developed. These algorithms include the P-Fibonacci-based algorithm [1], wave transform-based algorithm [2], chaos-based algorithm [3], and so on.

Among all image encryption algorithms, chaos-based algorithms show high encryption performance because chaotic maps have properties of unpredictability, ergodicity and can generate different random outputs with specific initial conditions. In recent years, many new chaotic maps have been proposed [4, 5] and many chaos-based image encryption algorithms have been developed successfully [6]. When chaotic maps are used for image encryption, their chaotic orbits are usually used to obtain random-like encrypted images. However, with the development of computational ability of computers and chaos discern theory, chaotic orbits with simple structures could be predicted [7, 8] and the encryption keys can be estimated. Thus, some chaos-based image encryption algorithms can be broken [9, 10].

To develop a secure chaos-based image encryption algorithm, this paper proposes a new 2D-LSM with complex chaotic behaviors and good chaotic performance. We then propose a new image encryption algorithm using the 2D-LSM. Simulations and several security evaluations are provided. The results show that the proposed algorithm can protect different types of images with a high security level.

The rest of paper is organized as follows: Section II will introduce 2D-LSM and analyze its chaotic performance; Section III will introduce the new image encryption algorithm. Its simulation results and security analysis will be demonstrated in Section IV; Section V will give a conclusion.

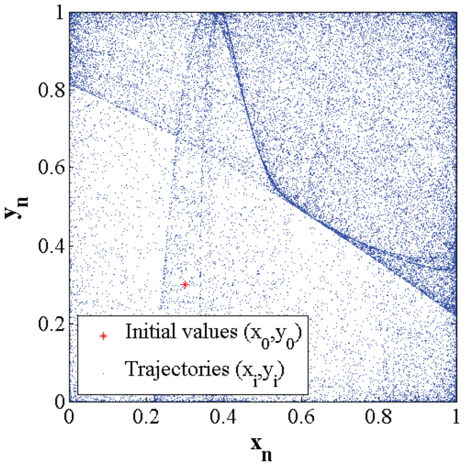
II. THE 2D LOGISTIC-SINE MAP (2D-LSM)

The 2D Logistic-Sine map (2D-LSM) is defined by E-qn. (1)

$$\begin{cases} x_{n+1} = \sin(\pi a(y_n + 3)x_n(1 - x_n)) \\ y_{n+1} = \sin(\pi a(x_{n+1} + 3)y_n(1 - y_n)) \end{cases} \tag{1}$$

where a is a control parameter and $a \in [0, 1]$, x_n, y_n are output values.

2D-LSM is derived from two 1D chaotic maps: the Sine and Logistic maps. We use the output of the Logistic map as the input of the Sine map, and then extend to 2D. 2D-LSM has good chaotic behaviors. Here, we use the trajectory and Shannon entropy [11] to estimate the chaotic performance of 2D-LSM.



(a)

Fig. 1. The trajectory of 2D-LSM with the initial values $(0.3, 0.3)$ and parameter $a = 0.93$.

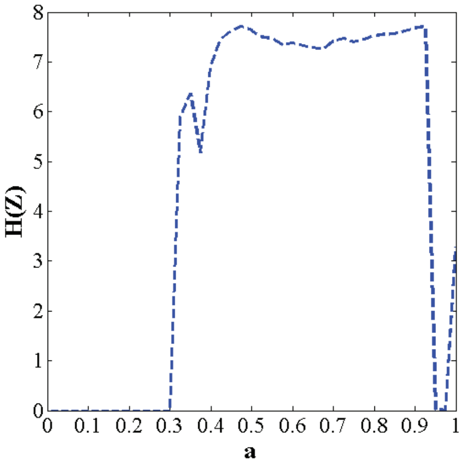
978-1-4799-3840-7/14/$31.00 ©2014 IEEE

Fig. 1 shows the trajectory of 2D-LSM. As can be seen, the points (x_i, y_i) randomly distribute in the whole phase plane. This means that the outputs of 2D-LSM has good properties of randomness and ergodicity.

Shannon entropy is a test standard to evaluate the randomness of a collection of data or a signal. It can be used to describe the distribution of a chaotic signal. Mathematically, Shannon entropy is defined as Eqn. (2)

$$H(Z) = -\sum_{i=1}^{n} Pr(z_i) \log_2 Pr(z_i) \tag{2}$$

where Z is a collection of data or a signal, z_i is the i-th possible value in Z and $Pr(z_i)$ is the probability of z_i. A bigger Shannon entropy value means better randomness, and the maximum of Shannon entropy value $H(Z)_{max} = \log_2 W$, where W is the number of possible values in Z.



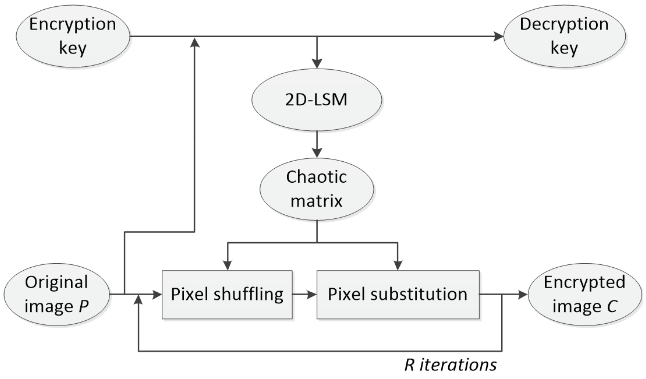
(a)

Fig. 2. The Shannon entropy distributions of the output sequences generated by 2D-LSM with different parameter settings.

To evaluate 2D-LSM using Shannon entropy, we first divide its outputs into 256 different levels, and calculate their Shannon entropy values. Fig. 2 plots the Shannon entropy distributions of the output sequences generated by 2D-LSM with different parameter settings. As can be seen, when parameter $a \in [0.35, 0.9]$ (approximately), the output sequences of 2D-LSM have large Shannon entropy values. This means that 2D-LSM has good randomness in this range.

III. New image encryption algorithm

Using the proposed 2D-LSM, this section introduces a new image encryption algorithm. The structure of the algorithm is shown in Fig. 3. The initial conditions of 2D-LSM are generated by the encryption key and original image P. The decryption key is a combination of the encryption key and some features of the original image P. The pixel shuffling process is to randomly change pixel positions within the image while the pixel substitution process is to randomly change pixel values. The iteration number R defined in Eqn. (3) is determined by the original image P. The entire image encryption process is described as $C = En(P, K_e)$ while the corresponding decryption process is represented by $P = De(C, K_d)$.



(a)

Fig. 3. The structure of the proposed image encryption algorithm.

A. Security key design

The encryption key $K_e = \{x_0,\ y_0,\ a\}$, in which three components are in the range of $[0, 1]$. To enhance security of the proposed algorithm, information of the original image P is combined with the encryption key to generate initial conditions for 2D-LSM in each iteration. First, we define a control parameter T and iteration number R in Eqn. (3)

$$\begin{aligned} T &= \sum_{i=1}^{M}\sum_{j=1}^{N} P(i,j) \mod 128 \\ R &= (\sum_{i=1}^{M}\sum_{j=1}^{N} P(i,j) \mod 3) + 2 \end{aligned} \tag{3}$$

In each iteration, the initial values and parameter of 2D-LSM can be generated from the control parameter T, iteration number R and encryption key K_e, as defined by Eqn. (4)

$$\begin{aligned} x_{0i} &= T \times i \times x_0 \mod 1 \\ y_{0i} &= T \times i \times y_0 \mod 1 \\ a_i &= (T \times i \times a \mod 0.5) + 0.4 \end{aligned} \tag{4}$$

where $a_i,\ x_{0i}$ and y_{0i} are the parameter and initial values of 2D-LSM in the i-th iteration, $1 \le i \le R$.

In each iteration, 2D-LSM uses the initial conditions $(x_{0i},\ y_{0i},\ a_i)$ to generate a chaotic matrix with the same size of the original image P. The chaotic matrix is used for the pixel shuffling and substitution processes for image encryption. The decryption key consists the control parameter T, iteration value R, and encryption key, $K_d = \{x_0,\ y_0,\ a,\ T,\ R\}$.

B. Pixel shuffling

Pixel shuffling is designed to change both the row and column positions of all pixels within an image. Because the permutation results are determined by the chaotic matrix, a pixel has the probability to be moved to any location within the image. Pixel shuffling is to break high correlations between neighboring pixels in the original image. It is described in Algorithm 1.

Algorithm 1. Pixel shuffling

```
Input: The original image $P$ and chaotic matrix $S$ with the size of $M \times N$.
$S(i,j) = floor(S(i,j) \times 2^{20}) \mod 2^{12}$, where $i \in [1,M]$ and $j \in [1,N]$
Initialize four empty matrixes $Te$, $I$, $IR$ and $IC$ with the size of $M \times N$
Convert $S(i,j)$ into a 12-bit stream
for $m = 1$ to $M$ do
    for $n = 1$ to $N$ do
        Convert $n$ into a 20-bit stream $D_n$
        Add $D_n$ to the rear of $S(m,n)$ to get a 32-bit stream $IR(m,n)$
        Convert $m$ into a 20-bit stream $D_m$
        Add $D_m$ to the rear of $S(m,n)$ to get a 32-bit stream $IC(m,n)$
    end for
end for
Sort each row of $IR$, and get $IR'$
Sort each column of $IC$ and get $IC'$
For $IR'(i,j)$ and $IC'(i,j)$, truncate their last 20 bits to make them as 20-bit streams. Then convert all the values into decimal integers
$Te(i,j) = P(i, IR'(i,j))$
$I(i,j) = Te(IC'(i,j), j)$
Output: The shuffled image $I$
```

C. Pixel substitution

For an ideally encrypted image, its pixels distribute uniformly to resist the statistical attacks. Pixel substitution changes all pixel values randomly such that the numbers of pixels in each intensity levels are approximately equal. The attackers is unable to use statistical methods to obtain any useful information from the encrypted image. Pixel substitution is defined by Eqn. (5)

$$C(i,j) = (\lfloor S(i,j) \times 2^{32} \rfloor + I(i,j)) \mod F \qquad (5)$$

where $i \in [1,M]$, $j \in [1,N]$, $\lfloor \cdot \rfloor$ is the floor operation. S is the chaotic matrix with the same size of the input image I, C is the substitution result, and F is the maximum number of possible pixel values in the original image. For example, $F = 256$ if the original image pixels are represented by 8 bits. The decryption procedure performs the inverse pixel substitution operation using the same chaotic matrix S.

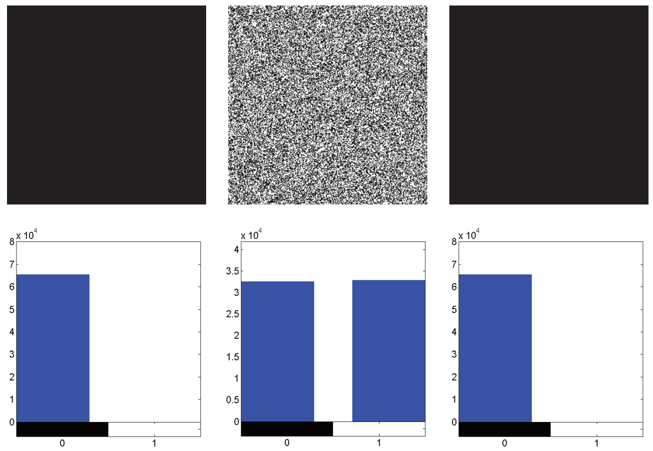
After R rounds of pixel shuffling and substitution operations using different chaotic matrices in each round, the proposed algorithm can encrypt a digital image into a random-like one. Because the iteration number R is determined by the original image, different original images may require different iterations in the encryption process. This further enhances the security level of the encrypted image.

IV. Simulations and security analysis

This section provides several simulation results to show the encryption performance of the proposed algorithm, and then analyzes its security performance.

A. Simulation results

The encryption results of binary and grayscale images are shown in Figs. 4 and 5, respectively. The binary image is an all-zero image. As can be seen, the encrypted images are noise-like and the reconstructed images are the same as the original images. The histograms of the encrypted images distribute uniformly. This ensures the attackers' difficulty of using statistics methods to obtain useful information from the encrypted images. The proposed algorithm shows the excellent encryption performance.



(a) (b) (c)

Fig. 4. Encryption of an all-zero image. (a) The original all-zero image with its histogram; (b) the encrypted image with its histogram; (c) the reconstructed image with its histogram.

B. Security analysis

An encryption algorithm is required to have excellent security performance such that it can resist different security attacks. Here, we use several security analysis methods to evaluate the proposed image encryption algorithm.

1) Key sensitivity: An encryption algorithm should be sensitive to its security keys [12]. The key sensitivity includes the encryption key sensitivity and decryption key sensitivity. The encryption key sensitivity means that a slight change in the encryption key will result in a completely different encrypted image. The decryption key sensitivity means that two slightly

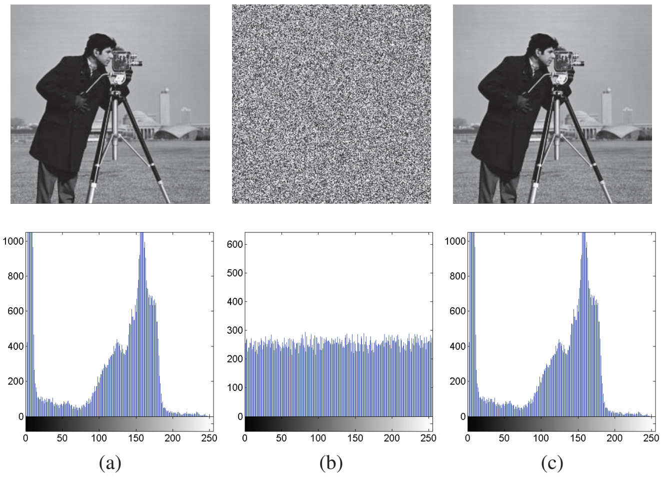


Fig. 5. Grayscale image encryption. (a) The original image with its histogram; (b) the encrypted image with its histogram; (c) the reconstructed image with its histogram.

different decryption keys yield two totally different decryption results.

Fig. 6 shows the results of the encryption key sensitivity analysis. As can be seen, two encryption keys K_{e1} and K_{e2} are of only one bit difference. When the original image P is encrypted by the proposed algorithm with K_{e1} and K_{e2} individually, the encrypted results C_1 and C_2 are totally different, as shown in their difference in Fig. 6(d). This means that the proposed image encryption algorithm is sensitive to its encryption key.

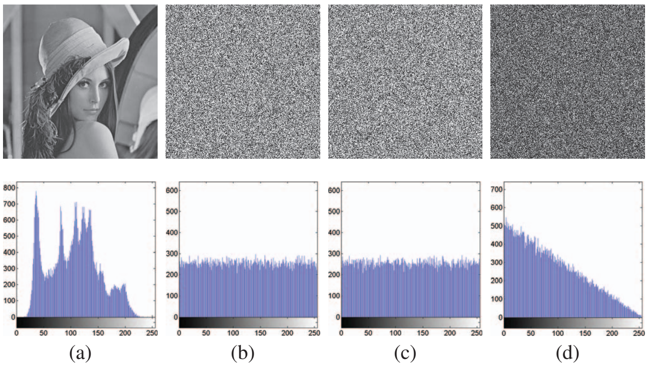


Fig. 6. Encryption key sensitivity analysis. (a) The original image P; (b) the encrypted result $C_1 = En(P, K_{e1})$; (c) the encrypted result $C_2 = En(P, K_{e2})$; (d) the difference between C_1 and C_2, $|C_1 - C_2|$.

The decryption key sensitivity analysis is shown in Fig. 7. We use Fig. 6(a) as the original image. K_{e1} is an encryption key and K_{d1} is its corresponding decryption key. K_{d2} and K_{d3} are two decryption keys different from K_{d1} with only one bit. From the results, we can see that the original image can be correctly reconstructed only by the corresponding decryption key as shown in Fig. 6(b). When the encrypted image (Fig. 7(a)) is decrypted by two decryption keys with only one bit difference from K_{d1}, the decrypted results in Figs. 7(c) and (d) are noise images. They are totally different as shown in Fig. 7(e). Therefore, the proposed algorithm is sensitive to its decryption key.

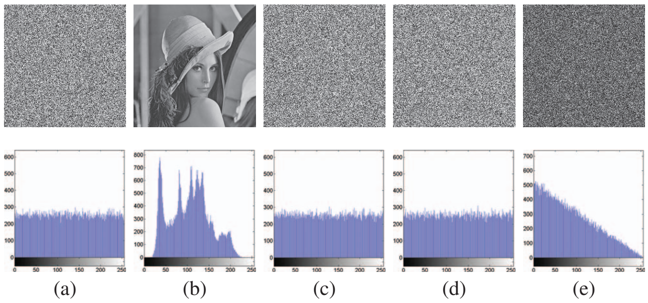


Fig. 7. Decryption key sensitivity analysis. The original image is Fig. 6(a). (a) The encrypted image $C_1 = En(P, K_{e1})$; (b) the decryption result $D_1 = De(C_1, K_{d1})$; (c) the decryption result $D_2 = De(C_1, K_{d2})$; (c) the decryption result $D_3 = De(C_1, K_{d3})$; (d) the difference between D_2 and D_3, $|D_2 - D_3|$.

2) Correlation analysis: For a natural image, high correlations may exist between its neighboring pixels. In image encryption, one important principle is to break this high correlations. Mathematically, the pixel correlation can be described by Eqn. (6)

$$Co = \frac{E[(X - \mu_X)(Y - \mu_Y)]}{\sigma_X \sigma_Y} \tag{6}$$

where X and Y are collections of pixels. μ is the mean value and σ is the standard derivation. When X and Y have high correlations, the test value is close to 1; when they have low correlations, the test value is close to 0.

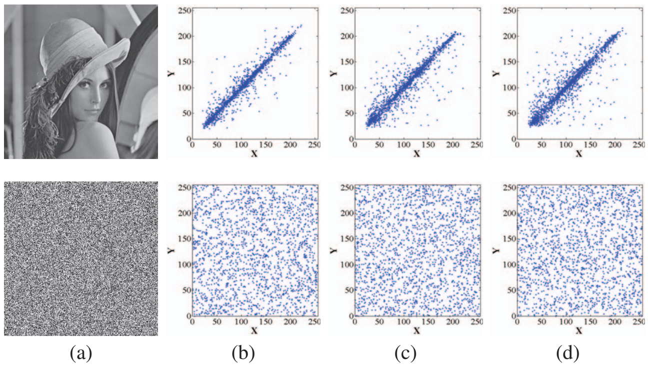


Fig. 8. Distributions of neighboring pixel pairs of (a) the original image and its encryption result at the (b) horizontal, (c) vertical and (d) diagonal directions, respectively.

TABLE I. NEIGHBORING PIXELS CORRELATION VALUES AT DIFFERENT DIRECTIONS.

	Horizontal	Vertical	Diagonal
Original image in Fig. 8(a)	0.945633	0.952352	0.925432
Encrypted image in Fig. 8(a)	0.001353	-0.007542	0.013453

Fig. 8 shows the distributions of neighboring pixel pairs of the original image and its encrypted result by the proposed algorithm. As can be seen, in the original image, the neighboring pixel pairs at different directions are distributed on or close to the diagonal axis in the coordinate system. This means that their values are close or equal and that the pixel pairs have high correlations. However, in its encrypted result, the pairs of neighboring pixels are randomly distributed in the entire data range. This means they have extremely low correlations. Table. I shows the quantitative results of correlation analysis. The correlations of the original image are close to 1 while those of the encrypted image are close to 0. Therefore, the neighboring pixels have high correlations in the original image and low correlations in the encrypted image.

3) Differential attack: The differential attack is a commonly used and effective cryptanalysis method. The encryption algorithm with an excellent diffusion property can resist the differential attack. The diffusion property is described as follows: when two images with a slight difference are encrypted by an encryption algorithm with the same encryption key, two encryption results are totally different. The diffusion property can be tested by the number of pixel change rate (NPCR) and unified averaged changed intensity (UACI). The NPCR and UACI are defined by Eqns. (7) and (8) [13]

$$NPCR(C_1, C_2) = \sum_{i=1}^{M}\sum_{j=1}^{N}\frac{D(i,j)}{L} \times 100\% \tag{7}$$

$$UACI(C_1, C_2) = \sum_{i=1}^{M}\sum_{j=1}^{N}\frac{|C_1(i,j) - C_2(i,j)|}{B \times L} \times 100\% \tag{8}$$

$$D(i,j) = \begin{cases} 0, & \text{if } C_1(i,j) = C_2(i,j) \\ 1, & \text{if } C_1(i,j) \neq C_2(i,j) \end{cases} \tag{9}$$

where L is the total number of pixels and B is the largest allowed pixel value in the image. C_1 and C_2 are two encrypted images.

The straightforward results of the differential attack are shown in Fig. 9. Two original images in Figs. 9(a) and (b) are of one pixel difference, which is shown in Fig. 9(c). When they are encrypted by the proposed algorithm with the same encryption key, the two encryption results in Figs. 9(d) and (e) are totally different, as shown in Fig. 9(f). The NPCR and UACI test results are 99.5956% and 33.4477%, respectively. They are close to the theoretically ideal values of NPCR and UACI (99.609% and 33.464%) reported in [13]. These show that the proposed algorithm has good diffusion property to resist the differential attack.

V. Conclusion

To overcome the weaknesses of existing 1D chaotic maps in being simple structures and easy to be predicted, this paper has proposed a new 2D Logistic-Sine map (2D-LSM). It is able to generate random and unpredictable chaotic sequences. The analysis results of trajectory and Shannon entropy have demonstrated its good randomness and ergodicity properties.

Based on the 2D-LSM, a new image encryption algorithm has been proposed. Simulation results and several security evaluations have shown that the proposed algorithm can encrypt different types of images with a high security level.

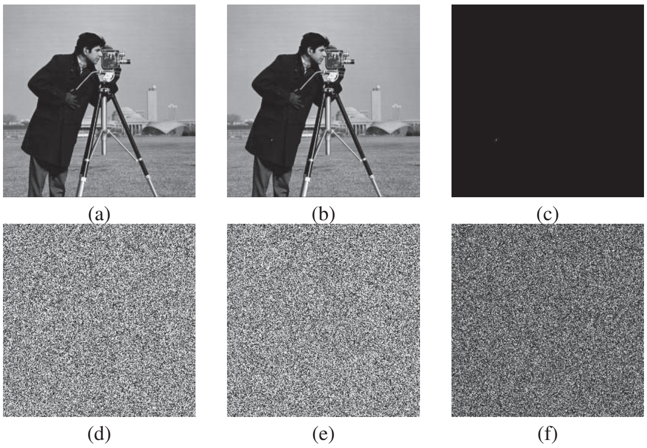


Fig. 9. Differential attack. (a) the original image P_1; (b) the original image P_2, which is obtained by randomly setting one pixel of P_1 as 255; (c) the difference between P_1 and P_2, $|P_1 - P_2|$; (d) the encrypted image $C_1 = En(P_1, K_e)$; (e) the encrypted image $C_2 = En(P_2, K_e)$; (f) the difference between C_1 and C_2, $|C_1 - C_2|$.

Acknowledgement

This work was supported in part by the Macau Science and Technology Development Fund under Grant FDCT/017/2012/A1 and by the Research Committee at University of Macau under Grants MYRG2014-00003-FST, MRG017/ZYC/2014/FST, MYRG113(Y1-L3)-FST12-ZYC and MRG001/ZYC/2013/FST.

References

[1] Y. Zhou, K. Panetta, S. Agaian, and C. L. P. Chen, “Image encryption using P-Fibonacci transform and decomposition,” *Optics Communications*, vol. 285, no. 5, pp. 594–608, 2012.

[2] X. Liao, S. Lai, and Q. Zhou, “A novel image encryption algorithm based on self-adaptive wave transmission,” *Signal Processing*, vol. 90, no. 9, pp. 2714–2722, 2010.

[3] R. Enayatifar, A. H. Abdullah, and I. F. Isnin, “Chaos-based image encryption using a hybrid genetic algorithm and a DNA sequence,” *Optics and Lasers in Engineering*, vol. 56, no. 0, pp. 83–93, 2014.

[4] Z. Hua, Y. Zhou, C.-M. Pun, and C. P. Chen, “A new 1D parameter-control chaotic framework,” in *IS&T/SPIE Electronic Imaging*. International Society for Optics and Photonics, 2014.

[5] Z. Hua, Y. Zhou, and C. L. P. Chen, “A new series-wound framework for generating 1D chaotic maps,” in *2013 IEEE Digital Signal Processing and Signal Processing Education Meeting (DSP/SPE)*, 2013, pp. 118–123.

[6] Y. Zhou, L. Bao, and C. L. P. Chen, “A new 1D chaotic system for image encryption,” *Signal Processing*, vol. 97, no. 0, pp. 172–182, 2014.

[7] G. Chen, Y. Chen, and H. Ogmen, “Identifying chaotic systems via a Wiener-type cascade model,” *IEEE Control Systems*, vol. 17, no. 5, pp. 29–36, 1997.

[8] H. C. Papadopoulos and G. W. Wornell, “Maximum-likelihood estimation of a class of chaotic signals,” *IEEE Transactions on Information Theory*, vol. 41, no. 1, pp. 312–317, 1995.

[9] S. Li, X. Mou, and Y. Cai, *Pseudo-random Bit Generator Based on Couple Chaotic Systems and Its Applications in Stream-Cipher Cryptography*, ser. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2001, vol. 2247, ch. 30, pp. 316–329.

[10] G. Alvarez, F. Montoya, M. Romera, and G. Pastor, "Cryptanalysis of a chaotic encryption system," *Physics Letters A*, vol. 276, no. 14, pp. 191–196, 2000.

[11] C. E. Shannon, "A mathematical theory of communication," *SIGMOBILE Mobile Computing and Communications Review*, vol. 5, no. 1, pp. 3–55, 2001.

[12] A. J. Menezes, P. C. Van Oorschot, and S. A. Vanstone, *Handbook of applied cryptography*. CRC press, 1996.

[13] C. Fu, J. Chen, H. Zou, W. Meng, Y. Zhan, and Y. Yu, "A chaos-based digital image encryption scheme with an improved diffusion strategy," *Optics Express*, vol. 20, no. 3, pp. 2363–2378, 2012.