Multimedia Tools and Applications (2020) 79:355–374
https://doi.org/10.1007/s11042-019-08079-x

Image encryption based on a new 2D logistic adjusted logistic map

Madhu Sharma[1]

Received: 30 September 2018 / Revised: 8 June 2019 / Accepted: 2 August 2019 /
Published online: 20 August 2019

Abstract
This paper proposes an encryption scheme based on a new 2-dimensional chaotic map. The new 2D chaotic map is derived from the idea of giving the two outputs of a 2D logistic map to two separate 1-dimensional logistic maps. The resulting 2D chaos based pseudo-random number generator is demonstrated to have significantly better randomness and unpredictability characteristics in terms of Lyapunov exponents as well as trajectory plots, in comparison to some recently proposed schemes based on other 2D chaotic maps. This new 2D chaotic map is then used to implement encryption of images. The proposed encryption scheme is demonstrated to be significantly better in terms of the required computational effort. For the proposed scheme, the commonly used measures of security, unpredictability and sensitivity to initial states are successfully established with the help of a set of standard simulation results.

Keywords Image encryption · New 2D chaotic map · Logistic map · Lyapunov exponent

1 Introduction

In the present age of high speed Internet becoming increasingly affordable to the masses, image security has been recognized as an important area of investigation. Images are one of the most significant components of data being transacted. The most important examples of digital image based documents needing to be secured may be found in medical, legal and intellectual property domains [2, 4, 8, 20, 30].

Image data has certain specific characteristics like 2-dimensional nature, correlation among adjacent pixels and specific image formats (viz. png, jpg, tiff etc.) which necessitate the development of encryption schemes [7, 38] meant specifically for images, different from other kinds of data - say, for example, textual data.

✉ Madhu Sharma
madhuashishsharma@gmail.com

[1] DIT University, Dehradun, PIN-248 001, India

Image encryption schemes based on chaos maps have found a lot of interest among researchers [11, 24, 26–28]. This is because of the fact that the pseudo-random number sequence generated on the basis of chaos maps are computationally simple and efficient, in addition to being unpredictable due to their extreme sensitivity to initial conditions. With the initial condition being defined by a secret key, the chaos map based methods naturally lead to a secure encryption systems.

In spite of being so powerful, the chaos based encryption schemes have been frequently cryptanalyzed Li et al. [22], Li and Lo [23] and Zhang et al. [37] – thus necessitating newer schemes to be regularly devised. Guan et al. [16] had presented a chaos based image encryption scheme. They utilized Arnold's cat map [1] for shuffling the pixels and Chen's chaotic system [9] for encrypting the gray scale value. However, Cokal and Solak [12] successfully recovered the secret parameters of this scheme utilizing the chosen plaintext and known plaintext attacks.

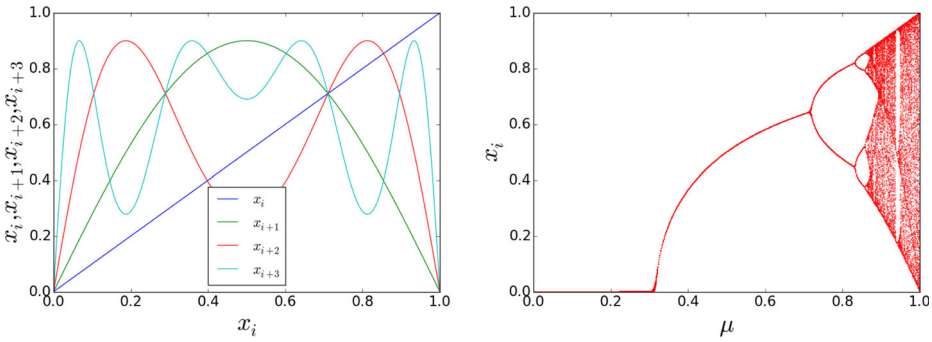
Chen et al. [10] extended Arnold's cat map [1] to a 3D cat map in order to develop an encryption scheme which decorrelated the 2D pixel arrangement due to its own 3D nature. The control parameters of most the chaos based schemes are first initialized and then they remain constant during the rest of the encryption-decryption process. However, Wang et al. [31] proposed an algorithm in which the control parameters themselves are two chaos maps obtained as a function of the plain image. Fridrich [13] demonstrated adaptable invertible two-dimensional chaotic maps for developing block encryption schemes and showed their application to image encryption. Yavuz et al. [36] applied two simultaneous 1D logistic maps for image encryption. While one map was used for shuffling the pixels, the other was used for changing the pixel values. Hsiao and Lee [17] proposed an image encryption scheme based on the chaos generated from a chaotic amplitude phase frequency model nonlinear adaptive filter. An asymmetric color image encryption algorithm based on chaotic systems and elliptic curves was demonstrated by Wu et al. [35]. Cao et al. [6] presented a bit-level image encryption algorithm based on a two-dimensional Logistic "ICMIC" cascade map derived from a "cascade modulation couple model".

Hua and Zhou [19] have employed a 2D Logistic Adjusted Sine Map (2D-LASM) to implement encryption of binary, gray scale and color images. The 2D-LASM basically involved giving the output of a slightly modified 2D logistic map as input to the sine map. A similar, and yet, significantly different work was reported by Hua et al. [18] using a 2D logistic map modulated by the sine map.

It can, thus, be clearly seen that apparently small variations in the mathematical formulation lead to significantly new and effective encryption algorithms [6, 10, 17–19, 35, 36]. Inspired by the work of Hua and Zhou [19], the present work first notes and demonstrates the operational similarity of the 1D logistic map with the 1D sine map. Based on this observation, in this work, instead of feeding the 2D logistic map's output to two sine maps [19], we give that output to two 1D logistic maps. The pseudo-random number sequence generated from the new 2D chaotic map is subjected to standard verifications in comparison to the sequences generated from previously proposed 2D chaotic maps. The new 2D map is then used to implement an image encryption scheme. Finally, this new scheme is demonstrated to be having comparably good/better security characteristics besides being computationally less demanding and significantly different too.

2 Review of chaotic maps

Here, we briefly review the background information which lead to the present work.



(a) Iteration diagram, $\mu = 0.9$. (b) Bifurcation diagram.

Fig. 1 Sine map

2.1 1D chaotic maps

Sine map and logistic map are among the few 1D chaotic maps commonly used in cryptography. Each such chaotic map leads to an iterate x_{i+1} from the previous one x_i as a function of a parameter μ as,

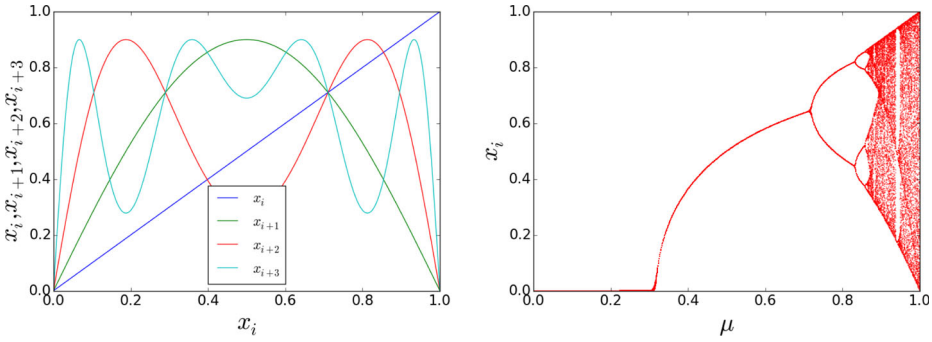
$$x_{i+1} = f(\mu, x_i) \tag{1}$$

In order to visualize the behavior of 1D chaotic maps, following two plots are used.

(a) Bifurcation diagram: To show the distribution of iterates x_i verses the control parameter μ.
(b) Iteration diagram: To show the variation of successive iterates $(x_{i+1}, x_{i+2}, x_{i+3})$ verses the 'initial state' x_i — for a constant value of the parameter μ.

Figures 1 and 2 give these two diagrams for Sine and Logistic maps, respectively. Where, Sine map, mapping $(0, 1) \rightarrow (0, 1)$, is described as

$$x_{i+1} = \mu \sin(\pi x_i) \tag{2}$$



(a) Iteration diagram, $\mu = 0.9$. (b) Bifurcation diagram.

Fig. 2 Logistic map

and, is known to be chaotic for $\mu \in [0.87, 1]$. And, Logistic map, mapping $(0, 1) \rightarrow (0, 1)$, is described as

$$x_{i+1} = 4\mu x_i(1 - x_i) \tag{3}$$

and, has been observed to be chaotic for $\mu \in [0.89, 1]$. This is evident from the above mentioned bifurcation diagrams.

2.2 2D chaotic maps

Like the 1D maps, each 2D chaotic map leads to 2D-iterate (x_{i+1}, y_{i+1}) from the previous one (x_i, y_i) as a function of a parameter μ. Here, some of the recently proposed 2D chaotic maps are presented.

The 2D Logistic Map (2D-LM), used by Wu et al. [32] for image encryption, is defined as,

$$x_{i+1} = \mu(3y_i + 1)x_i(1 - x_i) \tag{4}$$
$$y_{i+1} = \mu(3x_{i+1} + 1)y_i(1 - y_i) \tag{5}$$

The 2D Sine Logistic Modulation Map (2D-SLMM) presented by Hua et al. [18] is,

$$x_{i+1} = \mu[\sin(\pi y_i) + 3]x_i(1 - x_i) \tag{6}$$
$$y_{i+1} = \mu[\sin(\pi x_{i+1}) + 3]y_i(1 - y_i) \tag{7}$$

where, the parameter $\mu \in (0, 1]$.

The 2D-LASM demonstrated by Hua and Zhou [19],

$$x_{i+1} = \sin[\pi\mu(y_i + 3)x_i(1 - x_i)] \tag{8}$$
$$y_{i+1} = \sin[\pi\mu(x_{i+1} + 3)y_i(1 - y_i)] \tag{9}$$

where, again, the parameter $\mu \in (0, 1]$.

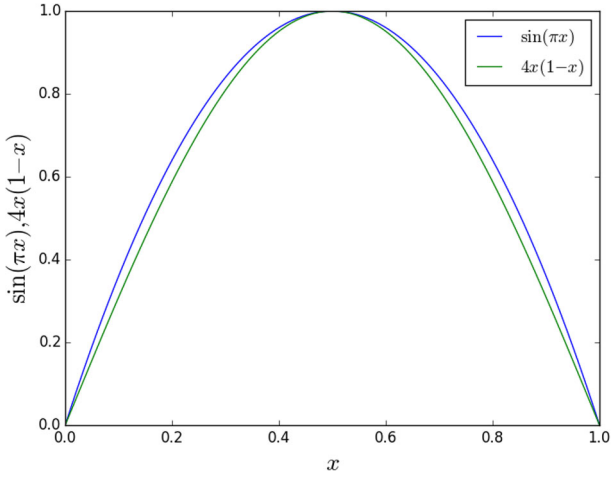


Fig. 3 Comparison of Sine and Logistic maps' kernel functions

3 Proposed 2D chaotic map

Figure 3 shows the comparison of kernel functions of Sine and Logistic maps. Looking at the similarity between these two plots, the present work proposes to replace the Sine function in the definition of 2D-LASM (8) and (9) with the kernel of Logistic function. Consequently, the proposed 2D chaotic map is named 2D Logistic Adjusted Logistic Map (2D-LALM), and, is defined as,

$$\bar{x}_{i+1} = \mu(y_i + 3)x_i(1 - x_i) \tag{10}$$

$$x_{i+1} = 4\bar{x}_{i+1}(1 - \bar{x}_{i+1}) \tag{11}$$

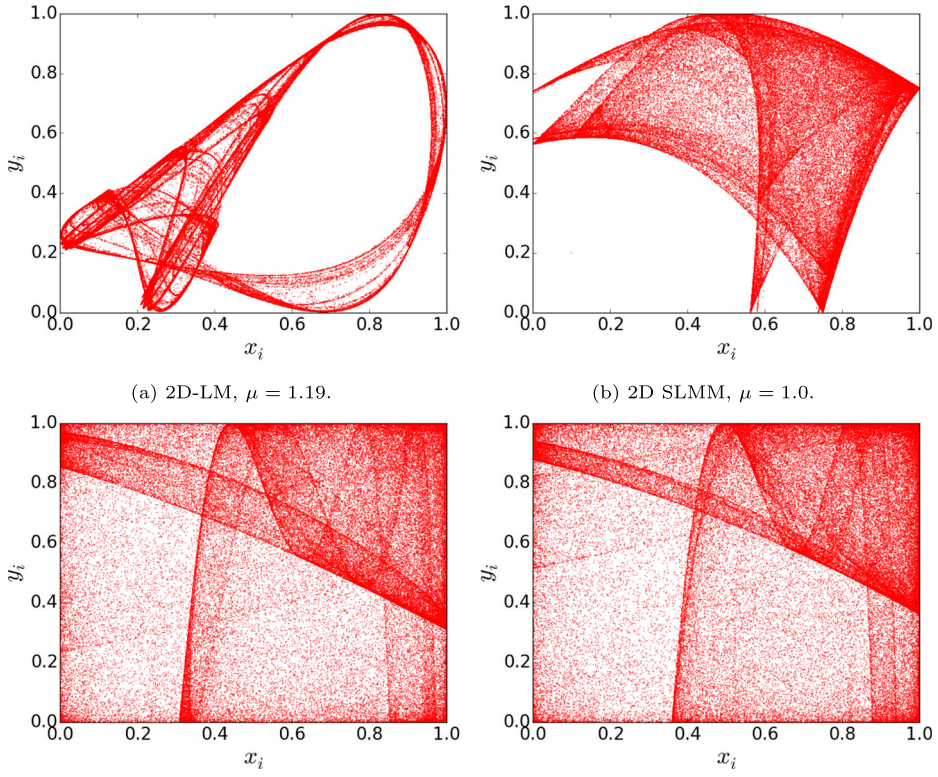
$$\bar{y}_{i+1} = \mu(x_{i+1} + 3)y_i(1 - y_i) \tag{12}$$

$$y_{i+1} = 4\bar{y}_{i+1}(1 - \bar{y}_{i+1}) \tag{13}$$

where, yet again, the parameter $\mu \in (0, 1]$. And, $(\bar{x}_{i+1}, \bar{y}_{i+1})$ are just intermediate variables.

3.1 Comparison among trajectories

Figure 4 shows the trajectory plot (x_i *vs* y_i) of 2D-LALM along with those of the other three 2D-maps mentioned above (2D-LM, 2D-SLMM and 2D-LASM). Each of these four plots starts with the initial condition ($x_i = 0.1$, $y_i = 0.2$). Among the four maps, the 2D map being proposed in this work (2D-LALM) has an almost identical distribution to the best



(a) 2D-LM, $\mu = 1.19$.

(b) 2D SLMM, $\mu = 1.0$.

(c) 2D-LASM, $\mu = 0.9$.

(d) 2D-LALM, $\mu = 0.9$.

Fig. 4 Trajectory Plots. In each case, initial condition is $(x_0,\ y_0) = (0.1,\ 0.2)$

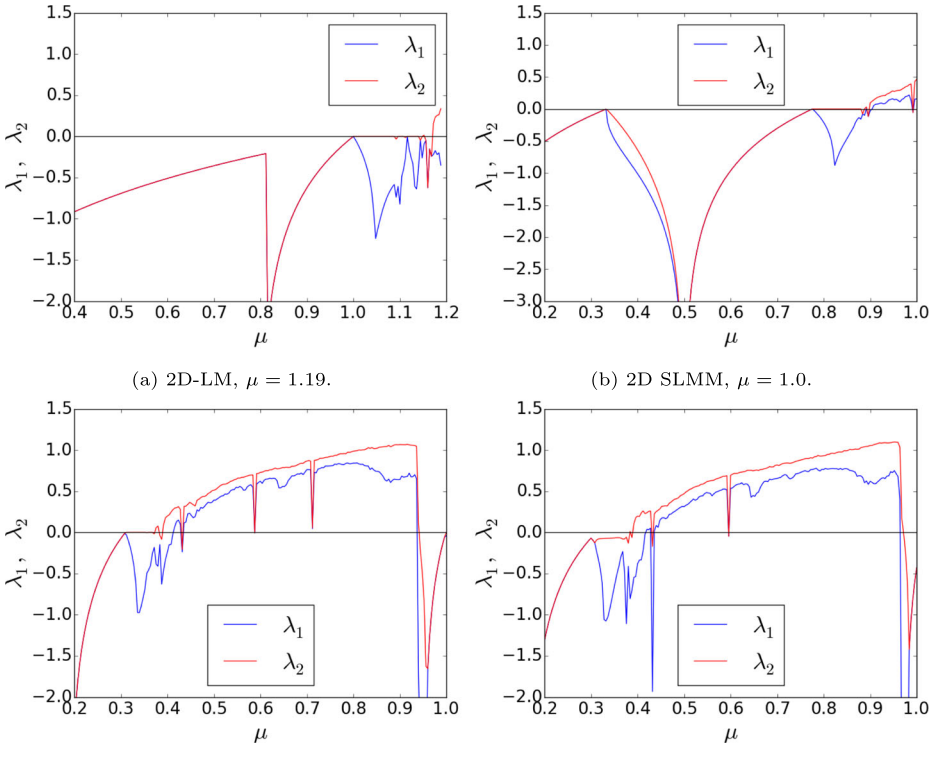
distribution provided by previously used three maps - 2D-LASM. Both, 2D-LASM and 2D-LALM, have their successive iterates $(x_i,\ y_i)$ much more uniformly distributed in the 2-D space in comparison to the other two.

3.2 Comparison among Lyapunov exponents

Lyapunov Exponents [3, 5, 19] are one of the standard measures of unpredictability of time series. Here, the Lyapunov exponents are computed using the pull back method attributed to Benettin et al. [3]. Figure 5 shows the variation of Lyapunov exponents of all the four 2-dimensional chaotic maps. The chaotic behavior of a time series is characterized by both positiveness as well as positivity of the Lyapunov Exponents. A multi-dimensional system is said to be hyperchaotic if it has more than one positive Lyapunov Exponents. As can be seen from Fig. 5, the newly proposed 2D-LALM has a well defined hyperchaotic behavior over a wide range - matched only by that of the 2D-LASM. The present methodology is, thus, designed to utilize μ in the range $(0.5 \leq \mu \leq 0.9)$.

4 The Encryption/Decryption process based on 2D-LALM

The present image encryption/decryption scheme is same as described by Hua and Zhou [19]. The only exception is the fact that the controlled randomization is achieved using



(a) 2D-LM, $\mu = 1.19$. (b) 2D SLMM, $\mu = 1.0$.

(c) 2D-LASM, $\mu = 0.9$. (d) 2D-LALM, $\mu = 0.9$.

Fig. 5 Variation of Lyapunov Exponents $(\lambda_1,\ \lambda_2)$ with the parameter μ

the 2D-LALM in place of the 2D-LASM. The scheme utilizes two 2-D chaotic maps for the encryption/decryption process. For the sake of completeness, and also for providing another perspective, the step-wise details of the process are given below. For simplicity of presentation, the description is given only for a gray scale image, although, the method is similarly applicable for binary and RGB images.

1. *Image Matrix P*

   The process starts with an image P (of a width of N pixels and a height of M pixels) visualized as a matrix as follows:

   $$P = \left[p_{ij}\right] \forall i = 1, 2, ...M,\ j = 1, 2, ...N \tag{14}$$

   Here, each pixel's representation (p_{ij}) is according to the mode of the image - binary, gray scale, RGB etc. Hence, for an 8 bit gray scale image, p_{ij} is an integer in the range of 0 to 255 ($= 2^8 - 1$).

2. *Adding a randomized border to the image*

   A border of a definite size and having randomized pixel-values is added around the original image. As the rest of the entire image-encryption procedure is already designed in a manner so as to produce a different cipher-image for even the slightest change in the original image, this additional step ensures that, for every run of the encryption process (even with the same key), the resulting cipher-image is significantly different. Hence, if a border of m pixels each is added to the top and the bottom, and, a border of n pixels each is added to the left and right extremes of the original image, the original image-matrix P gets enhanced to a new image-matrix P' as follows:

   $$P' = \left[p'_{ij}\right]; i = 1, 2, ...M',\ j = 1, 2, ...N' \tag{15}$$

   where,

   $$M' = M + 2m, \text{ and, } N' = N + 2n \tag{16}$$

   Here,

   $$p'_{ij} = p_{ij}\ \forall\ m < i \leq (M + m),\ n < j \leq (N + n) \tag{17}$$

   , and,

   $$p'_{ij} = \text{random}\,[0, 1, 2, 3...255], \quad \text{otherwise} \tag{18}$$

   assuming that the gray scale image has 8 bits per pixel.

3. *Initializing the two 2-D chaotic maps*

   Maintaining consistency with Hua and Zhou [19] for the sake of ease and justifiability of comparison with their results, a 232 bit key is utilized to initialize the encryption/decryption process. The 232 bit key is visualized as an array of 232 bits - 'K'. Taking $K[i]$ to denote the i^{th} bit of K, a slice of K from the i^{th} bit to the j^{th} bit (both inclusive) is denoted as $K[i : j]$ and is seen to identify an integer as follows:

   $$K[i : j] = \sum_{k=i}^{j} K[i] \times 2^{k-1} \tag{19}$$

   The initial states ($x_0^{(i)}$, $y_0^{(i)}$) and the system parameters (μ_i) of the two 2-D chaotic maps (2-D LALM) mentioned above are established as follows:

   $$x_0^{(i)} = (x_0 + w\gamma^i) \bmod 1;\ i = 1, 2 \tag{20}$$

   $$y_0^{(i)} = (y_0 + w\gamma^i) \bmod 1;\ i = 1, 2 \tag{21}$$

   $$\mu_i = [(\mu + w\gamma^i) \bmod 0.4] + 0.5;\ i = 1, 2 \tag{22}$$

where,

$$x_0 = \frac{1}{2^{52}} K[1:52] \tag{23}$$

$$y_0 = \frac{1}{2^{104}} K[53:104] \tag{24}$$

$$\mu = \frac{1}{2^{156}} K[105:156] \tag{25}$$

$$w = \frac{1}{2^{208}} K[157:208] \tag{26}$$

$$\gamma_1 = \frac{1}{2^{208}} K[209:220] \tag{27}$$

$$\gamma_2 = \frac{1}{2^{220}} K[221:232] \tag{28}$$

Further, to avoid self-mapping, each of the four initial state variables $(x_0^{(i)}, y_0^{(i)})$ is reset to 0.4 when it comes out to be zero in (20) and (21). Also, as noted earlier with respect to Fig. 5, in (22), it is ensured that the two system parameters (μ_i) lie in the range of 0.5 to 0.9.

4. *Generating the two pseudo-image matrices* $(S^{(1)}, S^{(2)})$

Using the two pairs of initial states $(x_0^{(i)}, y_0^{(i)})$ and the system parameters (μ_i), two pseudo-image matrices $(S^{(1)}, S^{(2)})$ are generated. Each of these two pseudo-image matrices are of the same size and format as the enhanced image-matrix P'.

$$S^{(k)} = \left[s_{ij}^{(k)}\right]; i = 1, 2, ...M',\ j = 1, 2, ...N',\ k = 1, 2 \tag{29}$$

5. *Encryption/Decryption of the enhanced image matrix* P'

Using each of the two pseudo-image matrices $(S^{(k)}; k = 1, 2)$, the following two steps are carried out in proper sequence to encrypt the enhanced image matrix P'.

(A) *Shuffling of the pixels*

For the $M' \times N'$ matrix P', a pixel index matrix (D) is visualized as follows:

$$D = \left[d_{ij}\right]; i = 1, 2, ...M',\ j = 1, 2, ...N' \tag{30}$$

$$d_{ij} = (i-1)N' + j \tag{31}$$

Thus,

$$D = \begin{bmatrix} 1 & 2 & ... & (N'-1) & N' \\ (N'+1) & (N'+2) & ... & ... & 2N' \\ ... & ... & ... & ... & ... \\ (M'-1)(N')+1 & (M'-1)(N')+2 & ... & ... & M'N' \end{bmatrix} \tag{32}$$

Let l_D be the number of bits needed to represent d_{ij}. A new matrix $R^{(k)}$ is evaluated as,

$$R^{(k)} = 2^{(8+l_d)} S^{(k)} + 2^8 D + P' \tag{33}$$

Consequently, each element of $R^{(k)}$ will have corresponding elements of the three matrices on the right hand side (33) ordered bitwise as - $s_{ij}^{(k)}$ in the highest 8 bits, d_{ij} in the next l_d number of bits and p'_{ij} in the lowest 8 bits. The elements of $R^{(k)}$ are then shuffled using row-wise and column-wise sorting as,

$$R^{(k')} = \text{column-sort}(\text{row-sort}(R^{(k)})) \tag{34}$$

Here, row-sort(.) sorts the elements in each row of a matrix, and so on. The final output of pixel-shuffling operation is obtained as the image matrix $T^{(k)}$:

$$T^{(k)} = \left[R^{(k')}\right] \text{AND } (2^8 - 1) \tag{35}$$

Hence, each element of $T^{(k)}$ just has an integer represented by the lowest eight bits of the corresponding elements of $R^{(k')}$. In other words, the elements of the image matrix $T^{(k)}$ are the shuffled elements of the image matrix P'.

(B) *Xoring and diffusion of the randomization among pixels*

The final step of encryption process is the usual XORing operation accompanied with diffusion of the randomization process among all the pixels. Thus, the final encrypted image matrix

$$O^{(k)} = \left[o_{ij}^{(k)}\right]; i = 1, 2, ...M',\ j = 1, 2, ...N' \tag{36}$$

is obtained as,

$$o_{ij}^{(k)} = \begin{cases} t_{ij}^{(k)} \oplus t_{M'N'}^{(k)} \oplus s_{ij}^{(k)} & \text{for } i = 1,\ j = 1 \\ t_{ij}^{(k)} \oplus o_{(i-1)N'}^{(k)} \oplus s_{ij}^{(k)} & \text{for } i \neq 1,\ j = 1 \\ t_{ij}^{(k)} \oplus o_{i(j-1)}^{(k)} \oplus s_{ij}^{(k)} & \text{for } j \neq 1 \end{cases} \tag{37}$$

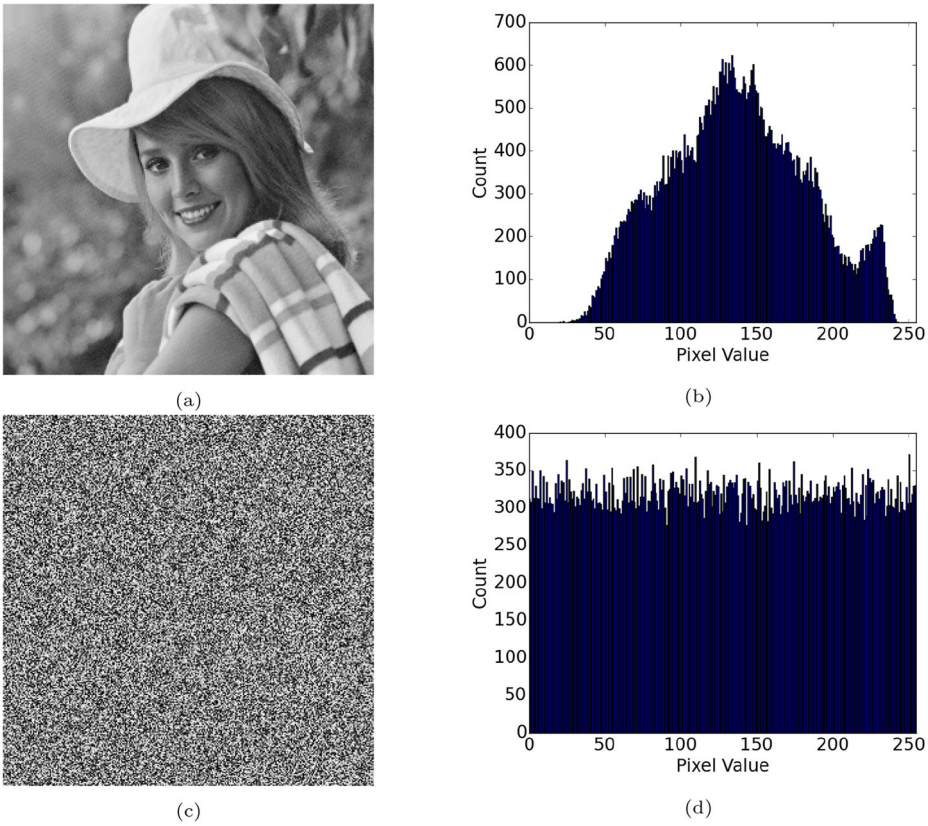


Fig. 6 Histograms of gray scale plain-image and cipher-image

Here, $t_{ij}^{(k)}$ are the elements of the matrix $T^{(k)}$, and, $(k = 1, 2)$.

The last step above (*Encryption/Decryption of the enhanced image matrix* P') is carried out first (for $k = 1$) with the pseudo-image matrix $S^{(1)}$, and then, repeated (for $k = 2$) with the second pseudo-image matrix $S^{(2)}$, taking as input $O^{(1)}$ in place of P'.

Hence, the complete encryption process can be summarized as generating as output the encrypted image in the form of $O^{(2)}$ taking as input the image P and the 232 bit key K. The decryption process involves the usual reversing of the above steps listed for the encryption process.

5 Simulation results

As mentioned earlier, a 232-bit key is used. The key used here is as follows (in hex):

$$K_1 = \text{0xAFE16E25A23D9D178D059526D0B5}$$
$$\text{C63471429DB435794F8A359004B490} \quad (38)$$

The results of encrypting a gray scale image are shown in Fig. 6 in the form of histograms of pixel values of plain and cipher images. The pattern seen in the histogram of the plain-image has been uniformly randomized in that of the cipher-image. Similar results for an

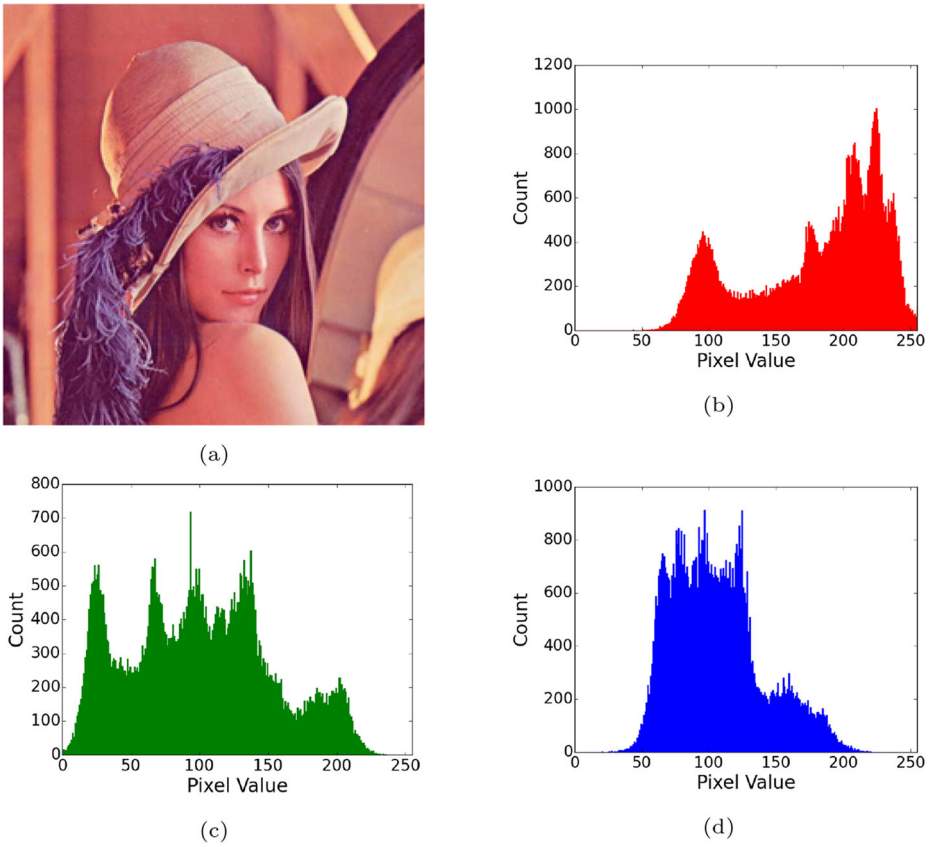


Fig. 7 Histograms of RGB plain-image

RGB colour image are shown in Figs. 7 and 8. Uniform randomization is again seen across all the three colour bands.

6 Security analysis

This section presents some standard security analysis results for the proposed image encryption scheme.

6.1 Sensitivity to the key

In addition to one 232-bit key given earlier (K_1), we consider two more such keys (K_2 and K_3) differing, in comparison to K_1, in only one bit.

$$K_2 = \text{0xAFE16E25A23D9D178D059526D0B5} \\ \text{C63471429DB435794F8A359004B491} \tag{39}$$

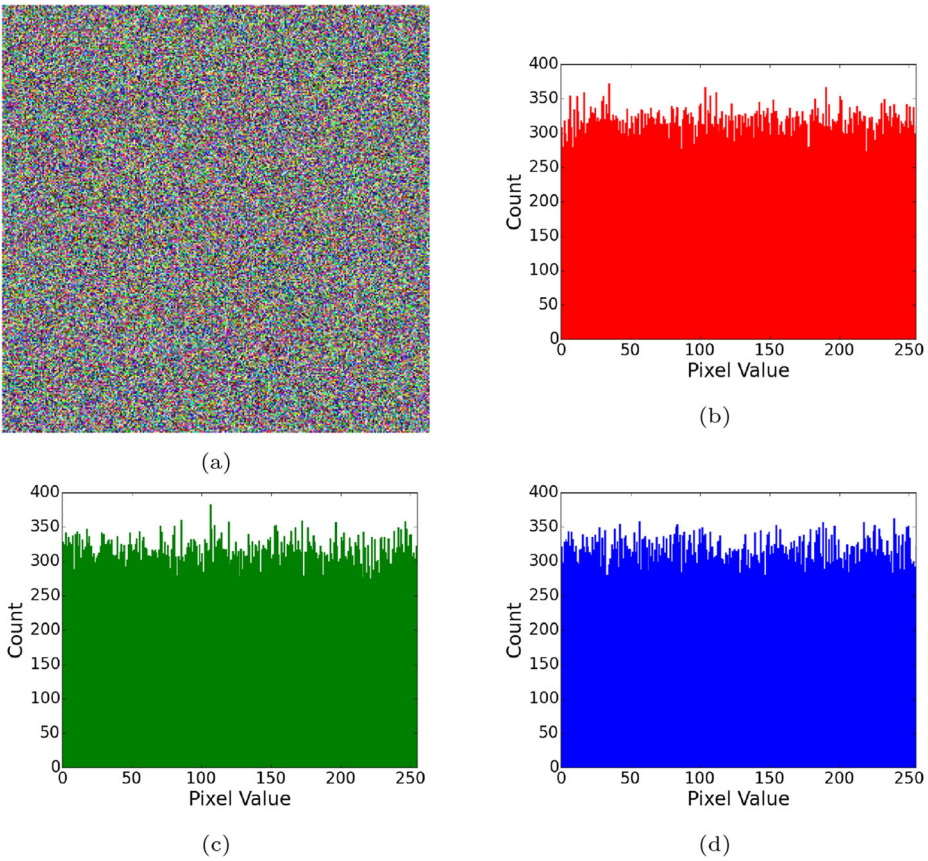


Fig. 8 Histograms of RGB cipher-image

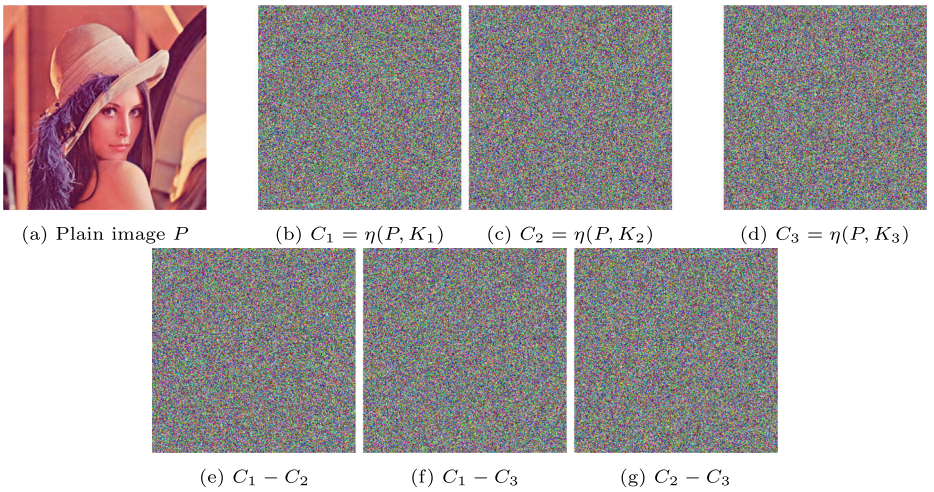


Fig. 9 Sensitivity of the encryption process to the encryption key

$$K_3 = \text{0xAFE16E25A23D9D178D059526D0B5}$$
$$\text{C63471429DB435794F8A359004B492} \tag{40}$$

Let the process of encryption of a plain image P in to a cipher image C using K as the encryption key be represented as,

$$C = \eta(P, K) \tag{41}$$

Similarly, let the decryption of the cipher image C in to the plain image P_d using K as the decryption key be,

$$P_d = \zeta(C, K) \tag{42}$$

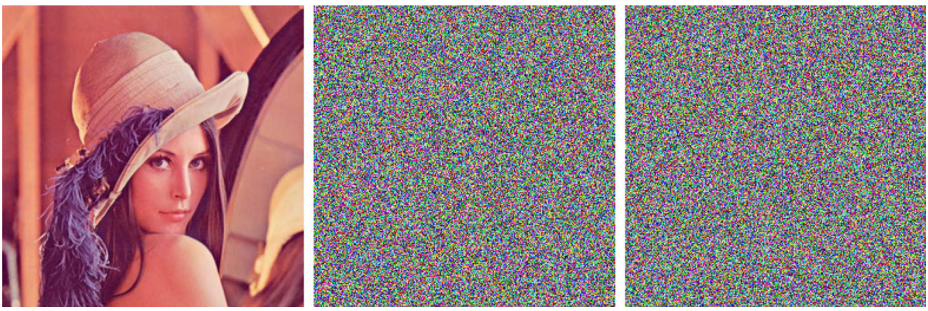
The sensitivity of the encryption process to a small variation in the encryption key is depicted in Fig. 9. And, the sensitivity of the decryption process to a small variation in the decryption key is depicted in Fig. 10. It can be seen that the slight variation in the encryption key leads to entirely different cipher-image. Similarly, a slightly different decryption key leads to an image entirely different from the plain-image.

6.2 Correlation of adjacent pixels

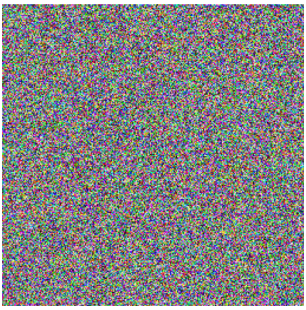
A natural (plain) image has strong correlation among its adjacent pixels. In contrast, due to the essential requirements of confusion and diffusion, the adjacent pixels of a cipher image are expected to have negligible correlation. This feature is quantified in terms of a parameter named as "Autocorrelation" (α) defined as follows:

$$\alpha = \frac{E[(X - E[X])(Y - E[X])]}{\sigma_X^2} \tag{43}$$

Here, X and Y are two adjacent sets (2D arrays) of pixels taken from a test image. The operator $E(.)$ evaluates the expected (mean) value of the operand array. σ_X is the standard deviation of the gray scale values of the pixels in X. The adjacency can be, as usual [19], horizontal, vertical or diagonal. In this study, the input plain image is "lena.bmp" [21] which is a 512×512 gray scale image. The present correlation results are shown in Table 1 along



(a) $P_{d11} = \zeta(C_1, K_1)$ (b) $P_{d12} = \zeta(C_1, K_2)$ (c) $P_{d13} = \zeta(C_1, K_2)$



(d) $P_{d12} - P_{d13}$

Fig. 10 Sensitivity of the decryption process to the decryption key

with those reported by earlier researchers. The correlation results of the present (2D-LALM) work are found to be good in comparison to the ones reported in literature.

6.3 Resistance to differential analysis

Attacks based on differential analysis involve subjecting the encryption-decryption system to varying inputs and studying how the corresponding outputs vary. To quantify the

Table 1 Autocorrelation results

	Horizontal	Vertical	Diagonal
Plain-image "Lena"	0.9690986	0.9840961	0.9556483
Image encryption schemes			
Chen et al. [10]	0.00024	0.24251	0.23644
Liao et al. [25]	0.0127	0.019	0.0012
Fu et al. [14]	0.0368	0.0392	0.0068
Wu et al. [32]	0.000215	0.0014913	0.0040264
Zhou et al. [39]	0.0054	0.0045	0.0031
Wu et al. [34]	0.0053365	0.0027616	0.0016621
LAS-IES [19]	0.0013174	0.0006427	0.0019122
2D-LALM	**0.000105**	**-0.0013968**	**-0.0004623**

influence of variation in one pixel change (in the input plain image) on the whole cipher image (output), two most common measures are NPCR (Number of Pixels Changing Rate) and UACI (Unified Average Changing Intensity).

First, we take two plain images P_1 and P_2. Here, P_2 differs in comparison to P_1 in terms of having only one bit different in only one of its pixels. Encryption using the same secret key (K) leads to cipher images C_1 and C_2, respectively. That is,

$$C_1 = \eta(P_1, K) \tag{44}$$

$$C_2 = \eta(P_2, K) \tag{45}$$

Table 2 NPCR scores of 8-bit gray scale images for different image encryption schemes

File name	Wu [32]	Zhou [39]	Liao [25]	Hua et al. [18]	Hua et al. [19]	2D-LALM
5.1.09	99.5804	99.60	49.8093	99.6658	99.6064	99.6139
5.1.10	99.5865	99.61	99.614	99.6475	99.6154	99.6335
5.1.11	99.5972	99.64	49.8138	99.6674	99.6244	99.6044
5.1.12	99.6201	99.60	49.828	99.5941	99.5703	99.6044
5.1.13	99.6414	99.63	99.5972	99.6445	99.6109	99.5954
5.1.14	99.5773	99.62	99.6368	99.5975	99.6364	99.6124
5.2.08	99.63	99.61	99.6208	99.6281	99.587	99.6079
5.2.09	99.6346	99.60	99.6174	99.6197	99.626	99.5911
5.2.10	99.6178	99.61	99.6292	99.6288	99.6124	99.606
7.1.01	99.5861	99.59	49.8005	99.6273	99.5992	99.6079
7.1.02	99.6178	99.62	49.8039	99.5892	99.6075	99.5963
7.1.03	99.6117	99.59	49.8096	99.6201	99.6079	99.5911
7.1.04	99.5808	99.62	99.6094	99.5894	99.5988	99.6079
7.1.05	99.5998	99.61	99.6063	99.6185	99.617	99.6079
7.1.06	99.6006	99.61	99.6048	99.6117	99.6272	99.6079
7.1.07	99.6059	99.60	99.6323	99.6223	99.5931	99.606
7.1.08	99.5918	99.58	99.6101	99.6151	99.6094	99.6079
7.1.09	99.601	99.61	49.81	99.6044	99.6162	99.5911
7.1.10	99.6002	99.63	49.8199	99.6101	99.6045	99.6103
Boat.512	99.6037	99.61	99.6037	99.6006	99.6154	99.606
Elaine.512	99.6082	99.6	99.6292	99.6128	99.6196	99.6079
Gray21.512	99.6075	99.61	99.6254	99.6082	99.6022	99.5911
Numbers.512	99.5995	99.6	99.612	99.6059	99.6141	99.6103
Ruler.512	99.6147	99.61	99.6304	99.6265	99.612	99.5961
5.3.01	99.6058	99.60	49.8086	99.6098	99.5931	99.6135
5.3.02	99.6005	99.62	99.6163	99.6119	99.6128	99.6138
7.2.01	99.6073	99.61	49.8199	99.6156	99.6156	99.6122
Testpat.1k	99.6117	99.62	99.6108	99.6124	99.6072	99.6178
Mean	99.605	99.6093	81.8196	99.618	99.6094	99.6061
Std	0.0157	0.013	24.302	0.0196	0.0133	0.0095

Then, we can calculate NPCR and UACI using the relations [19],

$$NPCR(C_1, C_2) = \sum_{i,j} \frac{A(i,j)}{G} \times 100\% \tag{46}$$

where,

$$A(i,j) = \begin{cases} 0 \ \forall \ C_1(i,j) = C_2(i,j), \\ 1 \ \forall \ C_1(i,j) \neq C_2(i,j) \end{cases} \tag{47}$$

and,

$$UACI(C_1, C_2) = \sum_{i,j} \frac{|C_1(i,j) - C_2(i,j)|}{(L-1) \times G} \times 100\% \tag{48}$$

Here, L is the number of possible gray scale levels of each pixel of the image, and, G represents the total number of pixels in the image.

As per the details provided by Fu et al. [15], for 8-bit gray scale images, the expected values of NPCR and UACI are 99.6094 % and 33.4635 %, respectively.

Using the image dataset of USC-SIPI "Misc." [29], which contains 28 images, the NPCR and UACI scores for different image encryption schemes are listed in Tables 2 and 3, respectively. For comparison, five other sets of results, as reported by Hua and Zhou [19], are also presented in the said two tables. Among all the six image encryption schemes, 2D-LALM achieves the average scores of NPCR and UACI closest to the expected ones. Thus, we can conclude that the 2D-LALM based scheme has a good ability of resisting differential attacks.

6.4 Resistance to chosen plain-text and cipher-text attacks

As is known, the chosen plain-text attack involves the determination of an encryption-decryption process's secret details by subjecting the process to a known plain-text and analysing the resulting cipher-text. The reverse attacking process happens in the case of chosen cipher-text attacks.

The present encryption scheme [19] involves randomizing every run of the encryption process (even for the same key and plain image) via the addition of a unique random border pixels. Further, the important attributes of confusion and diffusion of pixels in the cipher image have also been ensured. Due to these two reasons, the present scheme can be seen to be resistant to chosen plain-text and cipher-text attacks.

6.5 Randomness - Localized Shannon Entropy

Another important measure of the quality of an encryption algorithm is the overall randomness in the resulting cipher image. This randomness is quantified in terms of a parameter termed as "Local Shannon Entropy" (LOCSE, [19, 33]). Let the "Shannon Entropy" H of a gray scale image X be defined as Wu et al. [33],

$$H(X) = -\sum_{i=1}^{L} Pr(x_i) \log_2 Pr(x_i) \tag{49}$$

Here, $Pr(x_i)$ is the probability of occurrence of i^{th} possible pixel value x_i among all the possible L number of gray levels in the image X. Using the Shannon Entropies $H(X_j)$ of k

Table 3 UACI scores of 8-bit gray scale images for different image encryption schemes

File name	Wu [32]	Zhou [39]	Liao [25]	Hua et al. [18]	Hua et al. [19]	2D-LALM
5.1.09	33.5253	33.14	16.6687	33.598	33.4456	33.4215
5.1.10	33.3938	33.24	33.5374	33.5366	33.4946	33.5922
5.1.11	33.86	33.24	16.7015	33.4398	33.5541	33.4524
5.1.12	33.615	33.56	17.0621	33.4228	33.4302	33.4086
5.1.13	33.725	33.56	33.6419	33.4205	33.4438	33.2789
5.1.14	33.4491	33.21	34.2965	33.46967	33.4655	33.3903
5.2.08	33.3933	33.31	33.4267	33.472	33.4008	33.5400
5.2.09	33.5346	33.62	33.4553	33.4921	33.4804	33.4434
5.2.10	33.5265	33.31	33.4993	33.4914	33.4563	33.4927
7.1.01	33.4789	33.25	16.8228	33.5212	33.5037	33.5239
7.1.02	33.5416	33.27	16.8126	33.4846	33.4237	33.4130
7.1.03	33.4062	33.27	16.7308	33.4647	33.4291	33.4517
7.1.04	33.4845	33.21	33.4778	33.5202	33.4739	33.5085
7.1.05	33.4852	33.3	33.4581	33.54	33.4362	33.5045
7.1.06	33.4453	33.3	33.4489	33.5254	33.3954	33.5313
7.1.07	33.4535	33.15	33.5216	33.5205	33.4073	33.4876
7.1.08	33.476	33.26	33.4496	33.5678	33.4332	33.5327
7.1.09	33.4875	33.26	16.768	33.5223	33.4177	33.4777
7.1.10	33.4754	33.23	16.8557	33.4325	33.4344	33.4741
Boat.512	33.4994	33.42	33.6291	33.5097	33.4654	33.4589
Elaine.512	33.4355	33.37	33.4419	33.5477	33.4225	33.4742
Gray21.512	33.3743	33.37	33.477	33.393	33.4608	33.4485
Numbers.512	33.415	33.36	33.4503	33.3993	33.424	33.4959
Ruler.512	33.3807	33.43	34.0635	33.5129	33.4262	33.4670
5.3.01	33.4714	33.42	49.8086	33.4532	33.4585	33.4656
5.3.02	33.464	33.29	99.6163	33.4853	33.4605	33.4299
7.2.01	33.4917	33.59	33.4685	33.4965	33.4556	33.5093
Testpat.1k	33.5025	33.43	33.4786	33.4455	33.4347	33.5022
Mean	33.492	33.335	28.177	33.489	33.448	33.471
Std	0.1021	0.1289	7.9729	0.0507	0.03372	0.0587

number of random non-overlapping image blocks ($j = 1, 2....k$) taken from an image S, its "Local Shannon Entropy" (LOCSE) $\overline{H_{k,T_B}}$ is defined as Wu et al. [33],

$$\overline{H_{k,T_B}} = \frac{1}{k}\sum_{j=1}^{k} H(X_j) \tag{50}$$

Each one of the image blocks X_j has T_B number of pixels (non-overlapping). With ($k = 30$, $T_B = 1936$) for 8-bit gray scale images [33], the lower bound, upper bound and ideal values of $\overline{H_{k,T_B}}$ are 7.901515698, 7.903422936 and 7.902469317, respectively. Conducting the simulation on the 28 gray scale images (USC-SIPI "Misc." [29]), the results are presented in Table 4. Results from six other sources as reported by Hua and Zhou [19]

Table 4 LOCSE scores of 8-bit gray scale plain and cipher images for different image encryption schemes

File name	Plain Image	Wu [32]	Zhou [39]	Liao [25]	Hua et al. [18]	Hua et al. [19]	2D-LALM
5.1.09	5.948253	7.901985	7.903354	*7.904191*	7.902127	7.902521	7.902501
5.1.10	7.009960	7.902731	7.902443	7.902371	7.903402	7.902215	7.902278
5.1.11	4.913895	7.902446	7.902756	*7.900799*	7.902687	*7.901470*	7.902218
5.1.12	5.181903	7.902556	7.901526	7.903374	7.901906	*7.904045*	7.901929
5.1.13	1.403060	7.902688	*7.904563*	*7.904566*	7.902825	7.902184	7.902493
5.1.14	6.737685	*7.903474*	7.902954	7.903111	7.902340	*7.905557*	7.902476
5.2.08	5.818099	*7.903953*	7.902356	7.901762	7.903327	7.903328	7.902197
5.2.09	6.384914	7.902233	*7.899853*	*7.905854*	7.901765	7.902551	7.902620
5.2.10	4.904788	*7.900714*	7.902654	7.902768	7.902748	7.902888	7.901990
7.1.01	5.432175	7.902173	7.902634	7.902145	*7.901305*	7.902014	7.902494
7.1.02	2.384175	*7.900879*	7.901634	7.902157	7.901578	7.902254	7.902506
7.1.03	4.848621	7.902543	*7.905423*	*7.900645*	7.903099	*7.903894*	7.902883
7.1.04	5.193038	*7.901126*	7.902125	*7.904141*	7.902607	7.902539	7.902456
7.1.05	5.966493	*7.903579*	*7.883653*	*7.900027*	*7.905305*	7.902851	7.902422
7.1.06	6.018822	7.901930	7.902356	7.901736	7.902695	7.901960	7.902496
7.1.07	5.625370	7.903000	7.902364	*7.900802*	7.902896	7.901658	7.902400
7.1.08	4.405719	7.903197	*7.904456*	*7.900944*	7.901632	7.902129	7.902094
7.1.09	5.446080	7.902308	7.903012	*7.905658*	7.903173	7.903018	7.902685
7.1.10	5.307269	*7.899542*	7.901598	*7.893848*	7.901524	*7.901114*	7.902447
Boat.512	6.255248	7.901908	7.901879	*7.900712*	7.903088	7.902407	7.902334
Elaine.512	6.104411	*7.901122*	7.902989	7.902030	7.901720	7.901703	7.902648
Gray21.512	0.376627	*7.900170*	*7.905107*	7.902149	7.902688	7.901959	7.902329
Numbers.512	5.947982	*7.903615*	*7.892351*	*7.903579*	7.901657	7.901664	7.902126
Ruler.512	0.492257	7.903265	7.903001	*7.901428*	7.903052	7.901596	7.902292
5.3.01	5.680905	7.902727	7.902647	*7.901040*	7.901772	7.902751	7.902450
5.3.02	5.689569	7.903182	*7.910474*	*7.900981*	7.903328	7.901552	7.902184
7.2.01	4.857594	7.902772	7.901989	*7.904525*	7.902454	7.902452	7.902229
Testpat.1k	1.255093	7.902806	7.901681	7.903343	7.902752	7.902663	7.902589
Pass rate		18/28	20/28	11/28	26/28	23/28	28/28
Mean		7.902308	7.901923	7.902167	7.902552	7.902462	7.902384
Std		0.001060	0.004427	0.002222	0.000820	0.000905	0.000212

are also given for comparison. Ten number of simulations are carried out with each of the 28 images and the best results are reported. Simulation results falling outside the specified bounds are taken as failures and are highlighted by underlining. It can be seen that the present 2D-LALM based results have 100 % pass rate.

7 Conclusion

This work proposes a new 2D chaotic map - 2D Logistic Adjusted Logistic Map (2D-LALM). The 2D-LALM basically involves feeding the two outputs of an underlying 2D

logistic map to two respective 1D logistic maps. Using the trajectory plots and Lyapunov Exponents, this new 2D chaotic map is shown to have chaotic behavior better than or equal to those of the ones previously reported. This chaotic map is then assembled in to an image encryption scheme given by Hua et al. [19]. Among other significant characteristics, the encryption procedure produces a completely different cipher image for every run even with the same key. While using the computationally less demanding logistic map in comparison to the sine map used by the earlier researchers [19], this new image encryption scheme is shown to pass all the commonly used tests of security, robustness and effectiveness. Further, this demonstration points to possibilities of developing similar, and yet, effectively unique newer encryption procedures using simple modifications and/or refinements of the existing ones.

Compliance with Ethical Standards

Disclosure of potential conflicts of interest The author declares that there is no conflict of interest related to this work.

Research involving Human Participants and/or Animals This work did not involve any human participants or animals.

Informed Consent This work does not have any content needing any informed consent.

References

1. Arnold VI, Avez A (1968) Ergodic problems in classical mechanics. Benjamin, New York
2. Bellinii P, Mesitii M, Nesii P, Perlasca P (2018) Protection and composition of crossmedia content in collaborative environments. Multimed Tools Appl 77(2):2083–2114
3. Benettin G, Galgani L, Strelcyn JM (1976) Kolmogorov entropy and numerical experiments. Phys Rev A 14(6):2338–2345
4. Bhardwaj R, Aggarwal A (2019) Hiding clinical information in medical images: an encrypted dual-image reversible data hiding algorithm with base-3 numeral framework. Optik 181:1099–1112
5. Reggie B, Paul B, Abarbanel HDI (1991) Computing the Lyapunov spectrum of a dynamical system from an observed time series. Phys Rev A 43(6):2787–2806
6. Cao C, Sun K, Liu W (2018) A novel bit-level image encryption algorithm based on 2d-LICM hyperchaotic map. Signal Process 143:122–133
7. Chai X (2017) An image encryption algorithm based on bit level Brownian motion and new chaotic systems. Multimed Tools Appl 76(1):1159–1175
8. Chandrika BK, Aparna P, Sumam DS (2017) Perceptually lossless coder for volumetric medical image data. J Vis Commun Image Represent 46:23–32
9. Guanrong C, Tetsushi U (1999) Yet another chaotic attractor. Int J Bifurcation Chaos 9(7):1465–1466
10. Chen G, Mao Y, Chui CK (2004) A symmetric image encryption scheme based on 3D chaotic cat maps. Chaos Solitons Fractals 21(3):749–761
11. Chen J, Zhu Z, Fu C, Yu H, Zhang L (2015) A fast chaos-based image encryption scheme with a dynamic state variables selection mechanism. Commun Nonlinear Sci Numer Simul 20(3):846–860
12. Cokal C, Solak E (2009) Cryptanalysis of a chaos-based image encryption algorithm. Phys Lett A 373(15):1357–1360
13. Fridrich J (1998) Symmetric ciphers based on two-dimensional chaotic maps. Int J Bifurcation Chaos 8(6):1259–84
14. Fu C, Lin B, Miao Y, Liu X, Chen J (2011) A novel chaos-based bit-level permutation scheme for digital image encryption. Opt Commun 284(23):5415–5423
15. Fu C, Chen J, Zou H, Meng W, Zhan Y, Yu Y (2012) A chaos-based digital image encryption scheme with an improved diffusion strategy. Opt Express 20(3):2363–2378
16. Guan ZH, Huang F, Guan W (2005) Chaos-based image encryption algorithm. Phys Lett A 346(1-3):153–157

17. Hsiao H, Lee J (2015) Color image encryption using chaotic nonlinear adaptive filter. Signal Process 117:281–309
18. Hua Z, Zhou Y, Pun C, Chen CLP (2015) 2D Sine Logistic modulation map for image encryption. Inf Sci 297:80–94
19. Hua Z, Zhou Y (2016) Image encryption using 2D Logistic-adjusted-Sine map. Inf Sci 339:237–253
20. Kindt EJ (2018) Having yes, using no? about the new legal regime for biometric data. Comput Law Secur Rev 34(3):523–538
21. "lena512.bmp". https://www.ece.rice.edu/~wakin/images/lena512.bmp. Accessed 2 July 2018
22. Li S, Li C, Lo KT, Chen G (2008) Cryptanalysis of an image scrambling scheme without bandwidth expansion. IEEE Trans Circ Syst Video Technol 18(3):338–349
23. Li C, Lo KT (2011) Optimal quantitative cryptanalysis of permutation-only multimedia ciphers against plaintext attacks. Signal Process 91(4):949–954
24. Li C, Luo G, Qin K, Li C (2016) An image encryption scheme based on chaotic tent map. Nonlinear Dyn 87(1):127–133
25. Liao X, Lai S, Zhou Q (2010) A novel image encryption algorithm based on self-adaptive wave transmission. Signal Process 90(9):2714–2722
26. Liu W, Sun K, Zhu C (2016) A fast image encryption algorithm based on chaotic map. Opt Lasers Eng 84:26–36
27. Mao Y, Chen G, Lian S (2004) A novel fast image encryption scheme based on 3d chaotic baker maps. Int J Bifurcation Chaos 14(10):3613–3624
28. Sheela SJ, Suresh KV, Tandur D (2018) Image encryption based on modified Henon map using hybrid chaotic shift transform. Multimed Tools Appl 77(19):25223–25251
29. USC-SIPI Image Database Website. http://sipi.usc.edu/database/database.php?volume=misc. Accessed 2 July 2018
30. Villena S, Vega M, Mateos J, Rosenberg D, Katsaggelos AK (2018) Image super-resolution for outdoor digital forensics. Usability and legal aspects. Comput Ind 98:34–47
31. Wang Y, Wong K, Liao X, Xiang T, Chen G (2009) A chaos-based image encryption algorithm with variable control parameters. Chaos Solitons Fractals 41(4):1773–1783
32. Wu Y, Yang G, Jin H, Noonan JP (2012) Image encryption using the two-dimensional logistic chaotic map. J Electron Imaging 21(1):013–014
33. Wu Y, Zhou Y, Saveriades G, Agaian S, Noonan JP, Natarajan P (2013) Local Shannon entropy measure with statistical tests for image randomness. Inf Sci 222(10):323–342
34. Wu Y, Zhou Y, Noonan JP, Agaian S (2014) Design of image cipher using latin squares. Inf Sci 264(0):317–339
35. Wu J, Liao X, Yang B (2017) Color image encryption based on chaotic systems and elliptic curve ElGamal scheme. Signal Process 141:109–124
36. Erdem Y, Rifat Y, Cem KM, Ezgi Y (2016) A chaos-based image encryption algorithm with simple logical functions. Comput Electr Eng 54:471–483
37. Zhang Y, Xiao D, Wen W, Li M (2014) Breaking an image encryption algorithm based on hyper-chaotic system with only one round diffusion process. Nonlinear Dyn 76(3):1645–1650
38. Zhang W, Yu H, Zhao Y, Zhu Z (2015) Image encryption based on three-dimensional bit matrix permutation. Signal Process 118:36–50
39. Zhou Y, Bao L, Chen CLP (2013) Image encryption using a new parametric switching chaotic system. Signal Process 93(11):3039–3052

Madhu Sharma She completed her Bachelor of Engineering in Computer Science and Engineering from the Rajiv Gandhi Technological University, Bhopal, India in 2001. She received her Master of Engineering degree in Software Engineering from the Thapar University, Patiala, India in 2005. She is currently working as an Assistant Professor in the Department of Computer Science & Engineering of the DIT University, Dehradun. Her research interests include information security and IoT (Internet of Things).