Ankita Bisht, Mohit Dua*, Shelza Dua and Priyanka Jaroli

A Color Image Encryption Technique Based on Bit-Level Permutation and Alternate Logistic Maps

https://doi.org/10.1515/jisys-2018-0365
Received September 7, 2018; previously published online March 5, 2019.

Abstract: The paper presents an approach to encrypt the color images using bit-level permutation and alternate logistic map. The proposed method initially segregates the color image into red, green, and blue channels, transposes the segregated channels from the pixel-plane to bit-plane, and scrambles the bit-plane matrix using Arnold cat map (ACM). Finally, the red, blue, and green channels of the scrambled image are confused and diffused by applying alternate logistic map that uses a four-dimensional Lorenz system to generate a pseudorandom number sequence for the three channels. The parameters of ACM are generated with the help of Logistic-Sine map and Logistic-Tent map. The intensity values of scrambled pixels are altered by Tent-Sine map. One-dimensional and two-dimensional logistic maps are used for alternate logistic map implementation. The performance and security parameters histogram, correlation distribution, correlation coefficient, entropy, number of pixel change rate, and unified averaged changed intensity are computed to show the potential of the proposed encryption technique.

Keywords: ACM, LSM, LTM, TSM, Lorenz system, alternate logistic map.

1 Introduction

In this information age, with the advent of technology, it becomes easier for an individual to send processed data. The processed data are classified into different categories as digital images, text, audio, videos, etc. Digital images are also one of the types of processed data, which are often sent through the various media like networking sites, electronic media, and social media. These are used in many application areas like video conversation, medical science, online albums, and military images database. The color images draw more attention as these incorporate more data than grayscale images [12, 38, 39, 42]. Hence, their security and authenticity is the major issue of research interest [30]. A number of image encryption algorithms are being developed for the security and safety of digital image data while transferring them through the networking medium. The classical algorithms like Data Encryption Standard [10], Advanced Encryption Standard [32], International Data Encryption Algorithm [16], etc. are incapable of encrypting digital images as there is strong correlation among adjacent pixels and redundant data and bulk capacity of data is present in digital images. Also, the classical algorithms require higher processing time and power [3, 24, 30, 46, 47]. The chaotic encryption techniques acquired the consideration of researchers because of their ubiquitous nature like sensitivity to seed value and control parameters, ergodic state, quasi-randomness, complex dynamics, etc. [1, 18, 24, 27, 30, 46].

The chaotic map is an integral part of the chaotic cryptosystem. It helps in implementing the chaotic behavior within the cryptosystem. By using seed value and control parameters, chaotic maps produce fractals of pseudorandom numbers. Chaotic maps are also used as key streams generator for the various encrypted

***Corresponding author: Mohit Dua,** Department of Computer Engineering, NIT Kurukshetra, Haryana, India, e-mail: er.mohitdua@nitkkr.ac.in. https://orcid.org/0000-0001-7071-8323
Ankita Bisht and Priyanka Jaroli: Department of Computer Science, Banasthali Vidyapith Banasthali, Newai, India
Shelza Dua: Department of Electronics and Communication Engineering, NIT Kurukshetra, Haryana, India

systems [6, 23, 26]. In [23, 26], the Chebyshev polynomial and one-dimensional piecewise monotonic maps are used for the same. In [13, 28], the seed value of implemented logistic map (LM) and control parameters of Arnold map are used as the encryption key. The chaotic maps are categorized into one-dimensional [14, 26, 27] and multi-dimensional [5, 7, 11, 22, 27, 28, 41, 43, 48] as described in the research works of [11, 17]. The one-dimensional chaos has basic architecture, is easy to implement, and has low processing cost, but their chaotic orbit is quite smooth and one can predict it effortlessly [2]. The multi-dimensional chaos map has multiple parameters, sophisticated architecture, and higher processing cost [2, 17, 28]. Hence, a number of new chaos maps have been proposed by combining established chaotic maps. These provide a wider chaotic range, better chaotic behavior, and uniform distribution of density function as compared to traditional chaotic maps. Techniques that use new chaos maps are described in the research works of [2, 17, 36].

The confusion and diffusion process is an essential part of the chaotic cryptosystem [40, 46]. In confusion phase, there is the position interchanging between the pixels of an image while keeping values of all pixels same. In diffusion phase, the pixels' values of the image are changed by employing any chaotic map. Hua and Zhou [15] integrated the confusion-diffusion to one step to make the algorithm more efficient. In [28] Patidar et al. described the new shuffling-diffusion technique. The work in [21] is enhanced by applying a modified shuffling-diffusion technique which gave better results [45]. Until now, the substitution-diffusion is applied at pixel level, but it can also be applied at the bit level. The bit-level permutation changes the pixel value of the image as well as modifies its intensity value. So it is preferred over pixel level. To make the statistical characteristics of confusion and diffusion process more robust, the bit-level permutation techniques are employed in the algorithms [8, 20, 46]. In [48], Zhao et al. uses Arnold cat map (ACM) for bit-level permutation which illustrates the high security and efficiency of the technique. The ACM is also used to reduce the relationship between the pixels of an image via shuffling [13, 33]. Inspired by the these observations, the proposed scheme of this paper uses the ACM for bit-level permutation. The bit-level permutation is followed by the XOR operation which is used to change the intensity values of the image pixels. In [35], for encrypting partially an image, the XOR operations and wavelet transform was applied. The XOR operator is used in encryption process due to its efficient nature in diffusing the image pixels which changes the intensity values of the pixel [25, 31].

The high-dimensional chaotic maps are used for secure communication and image encryption technique [29, 41]. The security of an algorithm is directly proportional to its key space. To provide a larger key space, the high-dimensional differential equation is used. To increase the encryption speed and security of a system, the four-dimensional Lorenz system is developed by classical three-dimensional Lorenz system [9, 19, 49]. Inspired by the work, in the proposed scheme, the four-dimensional chaotic Lorenz system is employed as a key generator for the encryption algorithm.

The paper presents a novel approach to encrypt a color image using bit-level permutation and alternate LM. The proposed method initially segregates the color image into red, green, and blue channels, transposes the segregated channels from the pixel plane to the bit plane, and scrambles the bit-plane matrix using ACM. Finally, the red, green, and blue channels of the scrambled image are confused and diffused by applying an alternate LM that uses a four-dimensional Lorenz system to generate a pseudorandom number sequence for the three channels. The parameters of ACM are generated with the help of Logistic-Sine map (LSM) and Logistic-Tent map (LTM). The intensity values of scrambled pixels are altered by the Tent-Sine map (TSM). One-dimensional and two-dimensional LMs are used for alternate LM implementation.

The structure of the paper is as follows: Section 2 describes the chaotic maps used in the proposed technique, Sections 3 and 4 explain the working behind the proposed scheme and the experimental results obtained from it, respectively, and Section 5 concludes the paper.

2 Chaotic Maps

Chaotic maps play an important role in a chaotic cryptosystem. The initial and control parameter can be used as the encryption key and also for generating the pseudorandom number series. The chaotic maps used in this paper are described as follows.

2.1 Logistic Map

This is one of the oldest, simple, and extensively used chaotic maps that have been used in many chaos-based cryptography systems.

2.1.1 One-Dimensional LM

It can be easily implemented through hardware and software because of its simplicity. It has less processing overhead and is more balanced. Hence, it is a candidate for characterizing the sophisticated dynamic behavior. Equation (1) gives the mathematical expression of the one-dimensional LM.

$$p_{j+1} = B_k \, p_j (1 - p_j) \tag{1}$$

where p_j lies in the range between 0 and 1 for n number of iterations. The dynamic nature of the function p depends on the control parameter B_k . The function shows complete chaotic nature within the range of $3.56 < B_k \leq 4$. In the proposed scheme B_k is taken as 3.9989. The initial value of p_0 is also called seed value [42].

2.1.2 Two-Dimensional LM

The two-dimensional LM is the extension of one-dimensional LM. Chaotic cryptosystems usually employ the chaotic maps of more than one dimension to provide better key space and dependency on the control parameter. Hence, two-dimensional LM has a more complicated structure [28]. It shows more chaotic behavior than the one-dimensional LM. So it becomes harder for an attacker to breach the cryptosystem [42]. It can be mathematically expressed by Eq. (2).

$$\begin{cases} q_{j+1} = q_j \, B_{k1}(1 - q_j) + \; L_{y1} {r^2}_j \\ r_{j+1} = r_j \, B_{k2}(1 - r_j) + L_{y2}\left({q^2}_j + q_j r_j\right) \end{cases} \tag{2}$$

where $2.75 < B_{k1} \leq 3.4$, $2.75 < B_{k2} \leq 3.45$, $0.15 < \mathrm{L}_{y1} \leq 0.21$, and $0.13 < \mathrm{L}_{y2} \leq 0.15$. The values of q and r lie in the interval (0, 1). The values of B_{k1}, B_{k2}, L_{y1} , and L_{y2} and the initial values of the q_0, r_0 are provided by the secret key (Lorenz system) used for image encryption in alternate one-dimensional and two-dimensional LM [42, 49].

2.2 Four-Dimensional Chaotic Map (Lorenz System)

The Lorenz system is the four-dimensional differential chaotic equation, an extended version of the three-dimensional differential equations. Equation (3) represents the mathematical expression of the four-dimensional Lorenz system [45].

$$\begin{cases} \dfrac{dz_1}{dt} = a\,(z_2 - z_1) \\ \dfrac{dz_2}{dt} = bz_4 + cz_2 + dz_1 - z_1 z_3 - z_3 z_4 \\ \dfrac{dz_3}{dt} = fz_3 + z_2 z_4 + z_1 z_2 \\ \dfrac{dz_4}{dt} = gz_2 - \; ez_4 - 0.05 z_1 z_3 \end{cases} \tag{3}$$

Here z_1, z_2, z_3, and z_4 are state variables and a, b, c, d, e, f, and g are the constant parameters of the system. In [19] and [9], two Lorenz systems have been combined, and the performance analysis of new four-dimensional chaotic systems has been done. The developed differential equation system provides a set of dynamic solutions.

2.3 ACM

The ACM is performing the permutation within the pixels of an image. It is used to decrease the interrelationship between the pixels of the color image. Equation (4) gives the mathematical expression for the iterative form of ACM [24].

$$\begin{bmatrix} a' \\ b' \end{bmatrix} = \begin{bmatrix} 1 & Y_i \\ Z_i & Y_i Z_i + 1 \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} mod\ N \tag{4}$$

Here, a' and b' become the new coordinate positions of the pixel for the original positions of a and b. Y_i and Z_i are the control parameters of ACM.

2.4 LSM

This chaotic map is a combination of one-dimensional logistic and one-dimensional sine map. It provides greater chaotic range as compared to individual logistic and sine map. Equation (5) gives the mathematical expression for the LSM [6].

$$P_{m+1} = \left(B_k \star P_m \star (1 - P_m) + (4 - B_k) \star \frac{\sin(\pi \star P_m)}{4} \right) mod\ 1 \tag{5}$$

where $B_k\ \epsilon (0, 4)$. B_k is the control parameter, and its value is taken as 3.9999 in the implemented work.

2.5 LTM

This chaotic map is the combination of the one-dimensional LM and one-dimensional tent map. Equation (6) gives the mathematical expression for the LTM [6].

$$P_{m+1} = \begin{cases} \left(B_k \star (1 - P_m) \star P_m + (4 - B_k) \star \frac{P_m}{2} \right) mod\ 1\ P_i < 0.5 \\ \left(B_k \star (1 - P_m) \star P_m + (4 - B_k) \star \frac{(1 - P_m\ )}{2} \right) mod\ 1\ P_i \geqslant 0.5 \end{cases} \tag{6}$$

where $B_k\ \epsilon (0, 4)$. B_k is the control parameter, and its value is taken as 3.9999 in the implemented work.

2.6 TSM

This map is also the combination of one-dimensional tent map and one-dimensional sine map. Equation (7) gives the mathematical expression for the TSM [6].

$$P_{m+1} = \begin{cases} \left(B_k \star P_m + (4 - B_k) \frac{\sin(\pi P_m)}{4} \right) mod\ 1\ P_i < 0.5 \\ \left(B_k \star (1 - P_m)/2 + (4 - B_k) \frac{\sin(\pi P_m)}{4} \right) mod\ 1\ P_i \geqslant 0.5 \end{cases} \tag{7}$$

3 Proposed Scheme

This section of the paper describes the color image encryption technique. The proposed scheme comprises bit-level permutation and alternate one-dimensional, two-dimensional LM. Firstly, the color image of $[M \star\ N \star\ 3]$ is taken as input and is separated into red (p^r), green (p^g), and blue (p^b) channels. Then

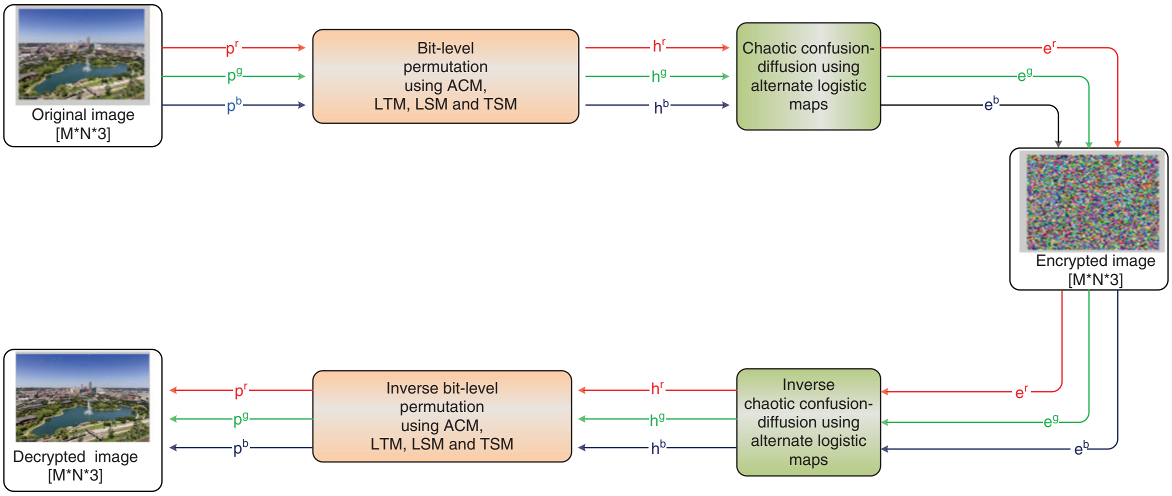
Original image
[M*N*3]
p^r
p^g
p^b
Bit-level permutation using ACM, LTM, LSM and TSM
h^r
h^g
h^b
Chaotic confusion-diffusion using alternate logistic maps
e^r
e^g
e^b
Encrypted image
[M*N*3]
Inverse chaotic confusion-diffusion using alternate logistic maps
Inverse bit-level permutation using ACM, LTM, LSM and TSM
Decrypted image
[M*N*3]

Figure 1: Block Diagram for Encyrption-Decryption Mechanism.

bit-level permutation is applied using ACM, LTM, LSM, and TSM, which gives scrambled red (h^r), green (h^g), and blue images (h^b) as output. In alternate LMs, the four-dimensional chaotic map (Lorenz system) is used as key generator to further confuse-diffuse the permutated image and to form the final encrypted images for the red (e^r), green (e^g), and blue channels (e^b). The decryption process is just reversal encryption steps inverse chaotic confusion-diffusion using alternate LMs followed by an inverse bit-level permutation using ACM, LTM, LSM, and TSM. The block diagram for the proposed method is shown by Figure 1.

3.1 Bit-Level Permutation

A color image of size $[M{\star}\ N{\star}\ 3]$ is taken and split it into its red (p^r), green (p^g), and blue (p^b) gray scale image channels. Each gray scale image is transposed from pixel plane to bit plane. The ACM is applied to scramble the bits of gray-scale images. The two chaotic functions LSM and LTM are used to generate a sequence of random numbers $s1$ and $s2$. The sequence is used to figure out the new location of the image pixels. The mathematical expression used to obtain a new location in ACM is given as follows:

$$Y \ = \ mod\ (\ (s1(i)\ \star\ (10^\wedge 9)\ ),\ 2 \star N) \tag{8}$$

$$Z = \ mod\ (\ (s2(i)\ \star\ (10^\wedge 9)\ ),\ 2 \star N) \tag{9}$$

Next, the TSM is used to generate pseudorandom number series (C) to diffuse the intensity value of the scrambled image by XORing with random numbers. The following expression is used to diffuse the intensity value of image pixel.

$$C(i) = mod\ (bitxor(bitxor(C(i), (s3(i))), C(i-1)),16) \tag{10}$$

3.2 Encryption Technique

The scrambled image is encrypted using one-dimensional and two-dimensional LMs one after another employing the four-dimensional Lorenz system as pseudorandom key generator.

3.2.1 Pseudorandom Key Generator

The Algorithm 1 below describes the steps for generating the encryption key using Lorenz system.

Algorithm 1: Pseudorandom Key Generator Using Lorenz System

Input: For generating keys for red, green, and blue channel, the initial values of constant parameters a, b, c, d, e, f, and g of the Lorenz system are taken as 16, 45, −2, 45, 16, −4, and 16.

Step 1: Create a four-dimensional chaotic sequence by taking the root values as $z01$, $z02$, $z03$, and $z04$ of chaotic system to iterate Eq. (3) $((M \times N)/3 + 10)$ times. Eliminate the first 10 group values of pseudorandom sequence and then obtain the four sequences z_1, z_2, z_3, and z_4.

Step 2: Quantize four-dimensional chaotic sequence by taking all the values of z_{j_i}, where $j = 1, 2, 3, 4$ and $i = 1, 2, 3 \ldots.(M \times N)/3$ and then obtain the decimal part of the four-dimensional chaotic sequence by using Eq. (11)

$$\emptyset z_{ji} = z_{j_i} - floor(z_{j_i}) \tag{11}$$

The decimal part of the chaotic sequence is quantized, and the integer sequence values z_1, z_2, z_3, and z_4 are obtained using Eq. (12).

$$z_{j_i} = mod(floor(\emptyset z_{j_i} \times 10^{14}), 65{,}536) \tag{12}$$

Step 3: Generate key sequence where z_{j_i} a sequence of integer values is and it is mathematically expressed by Eq. (13).

$$z_{j_i} = f_{j_{i15}} * 2^{15} + f_{j_{i14}} * 2^{14} + f_{j_{i13}} * 2^{13} + \,\ldots\ldots + f_{j_{i1}} * 2^{1} + f_{j_{i0}} \tag{13}$$

The following order reduces the correlation between three random bit sequences, and crosswise different bits of every sequence are taken.

Key 1 obtained from the first to eighth bits of z_{1_i}, z_{2_i}, and z_{3_i} is shown in Eq. (14).

$$\begin{cases} k1(k) = f_{1i_7} \times 2^7 + f_{1i_6} \times 2^6 + f_{1i_5} \times 2^5 + f_{1i_4} \times 2^4 + f_{1i_3} \times 2^3 + f_{1i_2} \times 2^2 + f_{1i_1} \times 2^1 + f_{1i_0} \\ k1(k+1) = f_{2i_7} \times 2^7 + f_{2i_6} \times 2^6 + f_{2i_5} \times 2^5 + f_{2i_4} \times 2^4 + f_{2i_3} \times 2^3 + f_{2i_2} \times 2^2 + f_{2i_1} \times 2^1 + f_{2i_0} \end{cases} \tag{14}$$

Key 2 obtained from the fifth to twelfth bits of z_{3_i}, z_{4_i}, and z_{1_i} is shown in Eq. (15).

$$\begin{cases} k2(k) = f_{3i_{11}} \times 2^7 + f_{3i_{10}} \times 2^6 + f_{3i_9} \times 2^5 + f_{3i_8} \times 2^4 + f_{3i_7} \times 2^3 + f_{3i_6} \times 2^2 + f_{3i_5} \times 2^1 + f_{3i_4} \\ k2(k+1) = f_{4i_{11}} \times 2^7 + f_{4i_{10}} \times 2^6 + f_{4i_9} \times 2^5 + f_{4i_8} \times 2^4 + f_{4i_7} \times 2^3 + f_{4i_6} \times 2^2 + f_{4i_5} \times 2^1 + f \end{cases} \tag{15}$$

Key 3 obtained from the ninth to sixteenth bits of z_{2_i}, z_{3_i}, and z_{4_i} is shown in Eq. (16).

$$\begin{cases} k3(k) = f_{2i_{15}} \times 2^7 + f_{2i_{14}} \times 2^6 + f_{2i_{13}} \times 2^5 + f_{2i_{12}} \times 2^4 + f_{2i_{11}} \times 2^3 + f_{2i_{10}} \times 2^2 + f_{2j_9} \times 2^1 + f_{2i_8} \\ k3(k+1) = f_{3i_{15}} \times 2^7 + f_{3i_{14}} \times 2^6 + f_{3i_{13}} \times 2^5 + f_{3i_{12}} \times 2^4 + f_{3i_{11}} \times 2^3 + f_{3i_{10}} \times 2^2 + f_{3j_9} \times 2^1 + f_{3i_8} \end{cases} \tag{16}$$

Output: k1, k2, and k3 are the encryption keys for red, green, and blue channels obtained as output of this algorithm.

The time taken to generate different lengths of multi-dimensional chaotic sequence is 5.181768 by this technique. A good pseudorandom key generator efficiently minimizes the length of chaotic sequence and hence largely reduce the time for encrypting the image.

3.2.2 Alternate LMs

The scrambled color image (h) of size $M \times N$ obtained through the bit-level permutation is further encrypted. The image is divided into R, G, and B channels. Each channel is represented by a pixel matrix of size $M \times N$, and the range of pixels is from 0 to 255. Each channel pixel matrix is denoted as h^r, h^g, and h^b, respectively. Algorithm 2 describes the steps involved in alternate LM.

Algorithm 2: Alternate Logistic Maps

Input: The scrambled image of size [$M * N * 3$] after bit-level permutation is performed is given as input.

Step1: Divide the scrambled image into R, G, and B channels matrix h^r, h^g, and h^b, respectively, and obtain the matrix A.

Step 2: H_1, H_2, and H_3 are determined for each channel of the image h by employing Eqs. (17) to (19), respectively. After that some permutations are done on each channel.

$$H_1 = \sum_{j\epsilon[1-255]} h_j^r/(M \times N \times 255) \tag{17}$$

$$H_2 = \sum_{j\epsilon[1-255]} h_j^g/(M \times N \times 255) \tag{18}$$

$$H_3 = \sum_{j\epsilon[1-255]} h_j^b/(M \times N \times 255) \tag{19}$$

Step 3: Repeat Eqs. (5) and (6) using B_k, B_{k1}, B_{k2}, L_{y1}, and L_{y2} as the initial values of the p_0, q_0, and r_0 are used. In each repetition the new values p_j, q_j, and r_j are obtained.

Step 4: Assign the value of keys $k1$, $k2$, and $k3$ should be to p_i, q_i, and r_i, and continue repeating two-dimensional equations and one-dimensional equation of logistic map alternately.

Step 5: Diffuse the values of red, green, and blue channels to obtain the final encrypted image A.

$$A(p, r) = (D(p, r) + A(p, r - 1))mod256, z\epsilon[2,\ M \times N] \tag{20}$$

where $A(p, r - 1)$ is the previous value of the channels and ($D(p,r)$) is the current value of the channels which are being processed.

Output: The final encrypted image with its red, green, and blue channels is obtained by the algorithm.

4 Experimental Setup and Security Analysis

This section of the paper discusses the tool used for implementing the color image encryption technique and results obtained after applying the technique. To implement the technique, MATLAB R2013a simulation tool, Operating System Windows 10 Pro, and processor intel core i-5 were used. The seed values for LSM, LTM, and TSM are set to 0.19235188279821, 0.73457891876543, and 0.56399882091176, respectively. The value of bifurcation parameter is 3.9999 for all the three maps. The implementation of the alternate LM with Lorenz system as the key generator is done by using the parameters, and initial values are $v = 3.99$, $v_1 = 3.39$, $v_2 = 3.4489$, $L_1 = 0.21$, $L_2 = 0.15$, $x_0 = 0.345$, $y_0 = 0.365$, and $z_0 = 0.537$. Figure 2A shows the input "Aerial" image of size 170 × 170 taken for testing the proposed work. Figure 2B–D show its red, green, and blue channels. Figure 3A–D show the final encrypted color image with the red, green, and blue channels, respectively. Figure 4A–D show the final decrypted image with red, green, and blue channels, respectively.

Figure 2A–D show a final encrypted color image with the red, green, and blue channels, respectively.

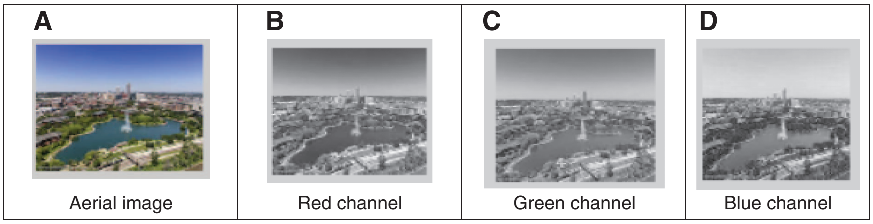

Figure 2: (A–D) Original "Aerial" Image with R, G, and B channels.

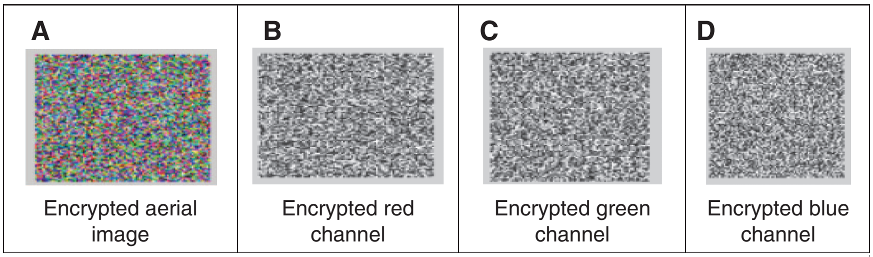

Figure 3: (A–D) Encrypted “Aerial” Image with R, G, and B channels.

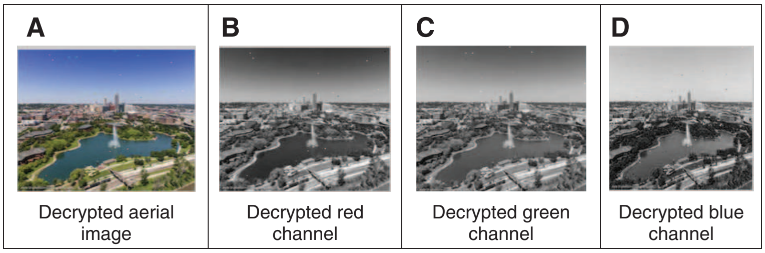

Figure 4: (A–D) Decrypted “Aerial” Image with R, G, and B channels.

4.1 Key Space Analysis

The key space analysis demonstrates the probability of finding an encryption key by applying all the possible keys [29]. The key space of the implemented technique increases exponentially with increase in key size. It also depends upon the initial value of chaotic maps used in the system [45]. In the proposed scheme, the three chaotic maps are used which has 14 digits after the decimal points. The key space for them is equal to $10^{14\times3}$, which is more than 2^{186}, and the parameters v, v_1, v_2, L_{y1}, L_{y2}, x_0, y_0, and $z_0$are used as encryption keys in alternate one-dimensional and two-dimensional map. The precision for them is 10^{-15}. Since the total key space becomes almost 10^{165}. Hence, the overall key space is large enough to resist brute force attack [42].

4.2 Statistical Analysis

It demonstrates the nearness between the plaintext image and the cipher image. There must be no similarity between the cipher and the plaintext image. The two measures of statistical analysis are image histogram and correlation between the adjacent pixels [40]. The histogram illustrates the distribution of image pixels by plotting the graph of the intensity level of each pixel present in the image. Usually, for original plaintext, the graph formed is steeper, raised, and fluctuating in nature, while for encrypted images the graph is consistently scattered and much different from the original plaintext. Hence, it shows no statistical similarity between the original and the encrypted images. Figure 5A–C show histograms of R, G, and B channels of the original image, respectively. Figure 6A–C show the histograms of R, G, and B channels of the encrypted image, respectively.

The correlation among the adjacent pixels of an original plaintext image is stronger, whereas in the encrypted image the correlation must be weaker such that the original images cannot be easily retrieved from the encrypted images. The correlation distribution of the red, green, and blue channels of the “Aerial” color image in each direction is shown in Figure 7. Figure 7A–C show the correlation distribution of the red channel of the plain color image horizontally, vertically, and diagonally. Figure 7D–F show the correlation distribution of the green channel of the plain color image horizontally, vertically, and diagonally. Figure 7G–I show the correlation distribution of the blue channel of the plain color image horizontally, vertically, and diagonally.

The correlation distribution of the red, green, and blue channels of the encrypted color image in each direction is shown in Figure 8. Figure 8A–C show the correlation distribution of the red channel of the

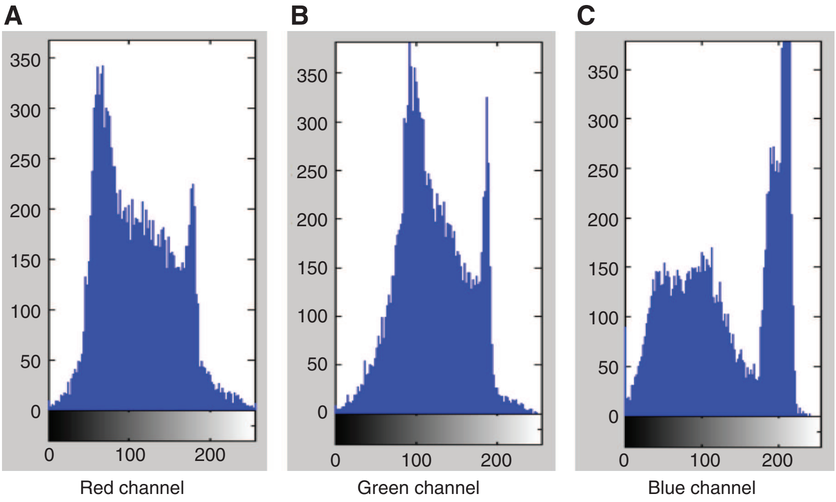

Figure 5: Histogram Distribution of the Original Image.

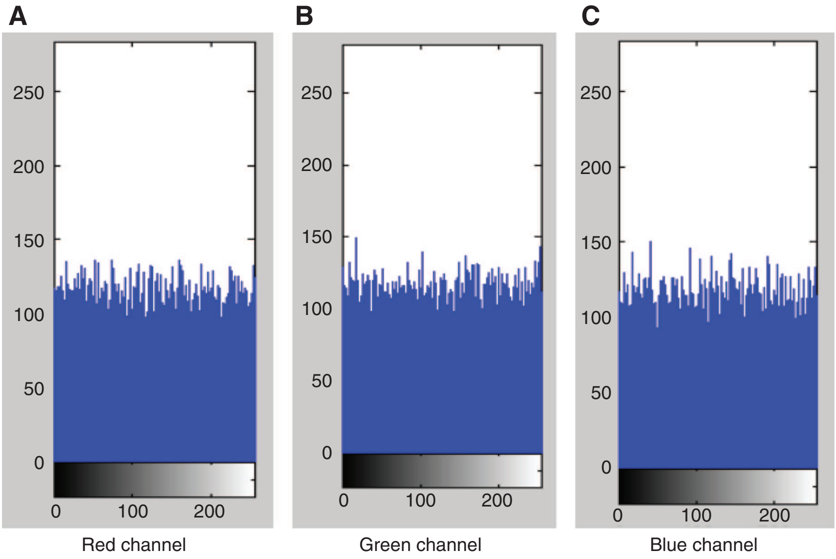

Figure 6: Histogram Distribution of the Encrypted Image.

encrypted color image horizontally, vertically, and diagonally. Figure 8D–F show the correlation distribution of the green channel of the encrypted image horizontally, vertically, and diagonally. Figure 8G–I show the correlation distribution of the blue channel of the encrypted image horizontally, vertically, and diagonally.

4.3 Correlation Coefficient Analysis

The correlation among the pixels of the original plaintext is high in each direction whether it is horizontal, vertical, or diagonal, while for the encrypted image the correlation should be very small in each direction. When the correlation is low, it implies that the algorithm has a better ability to resist statistical attack [40]. In plain image and encrypted image, 3000 pairs of adjacent pixels are selected to calculate the correlation coefficient. It is calculated by using the following equations [5]:

$$A_{y,z} = \frac{\operatorname{cov}(y, z)}{\sqrt{D(y)}\sqrt{D(z)}} \tag{21}$$

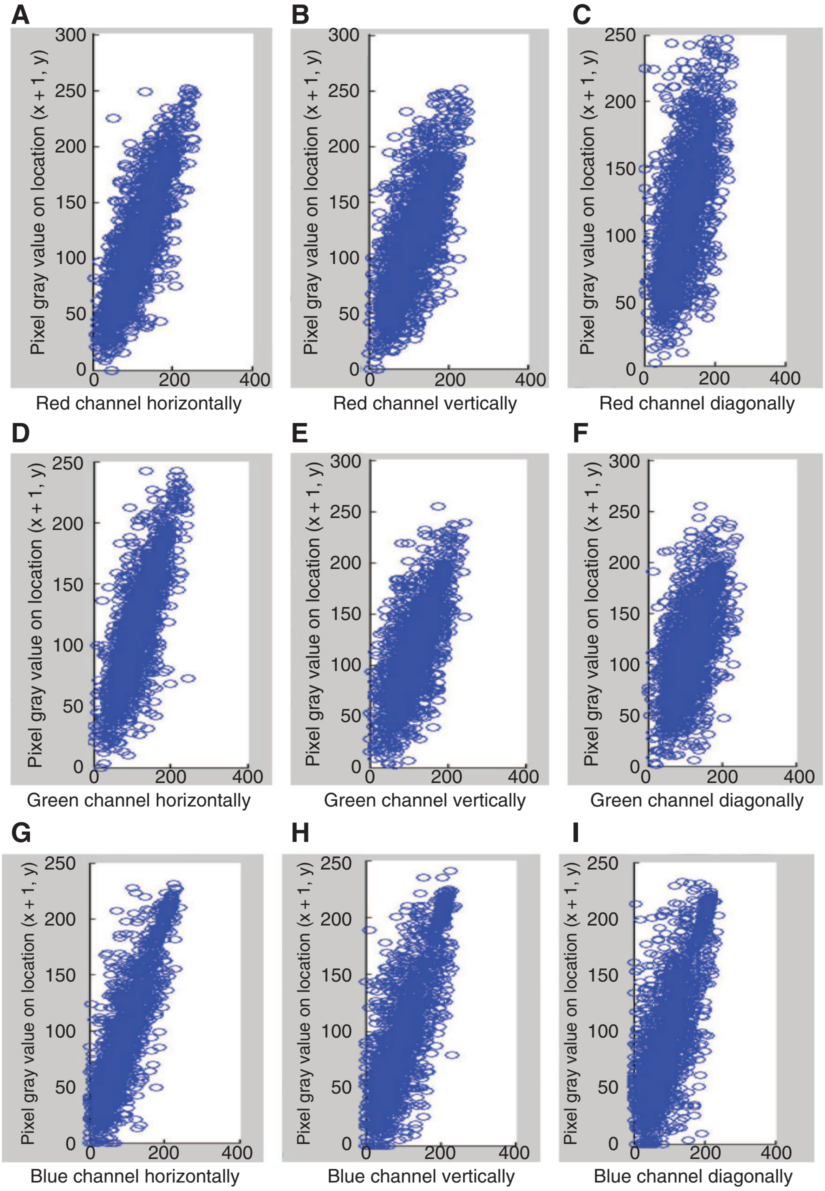

Figure 7: Correlation Distribution of Red, Green, and Blue Channels of the Original Color Image.

where

$$\operatorname{cov}(y,z)=\frac{1}{N}\sum_{i=1}^{n}(y_i-E(y)(z_i-E(z))) \tag{22}$$

$$E(y)=\frac{1}{N}\sum_{i=1}^{n}(y_i) \tag{23}$$

$$D(y)=\frac{1}{N}\sum_{i=1}^{n}(y_i-E(y))^2 \tag{24}$$

Table 1 shows the correlation coefficients between the R, G, and B channels of the encrypted color image.

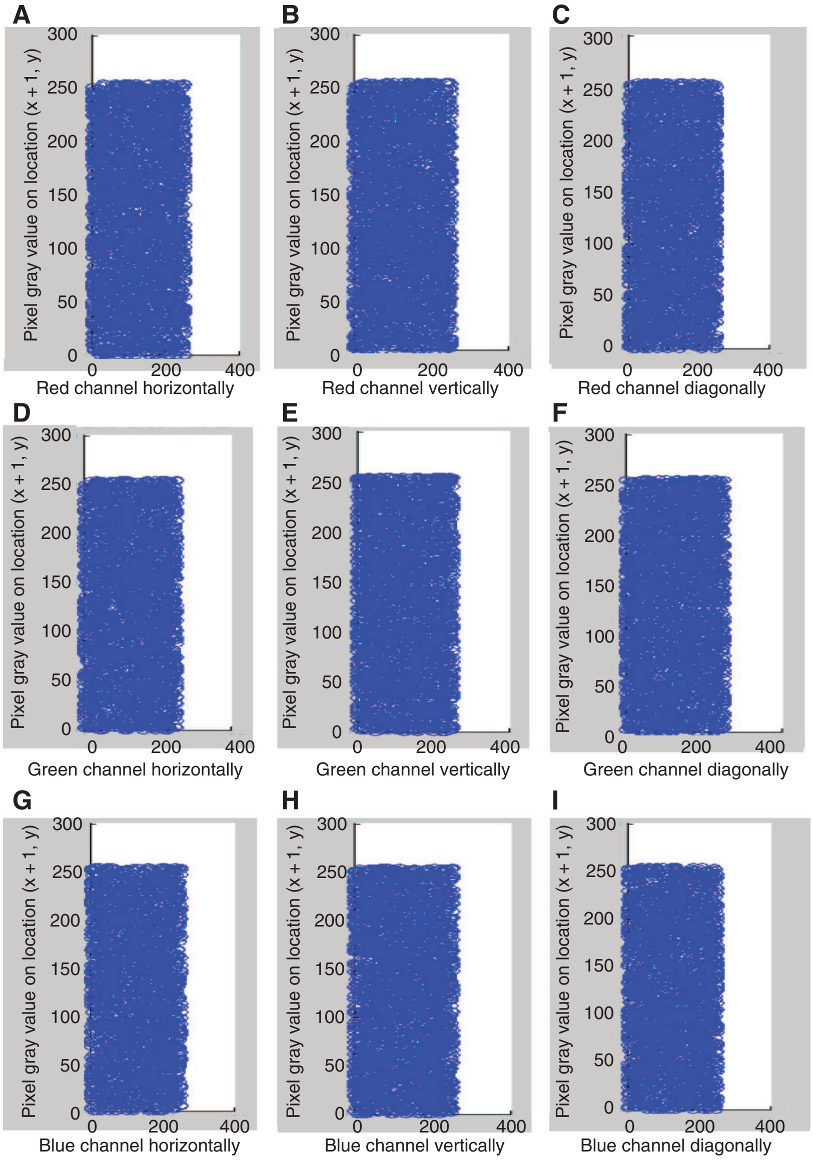

Figure 8: Correlation Distribution of Encrypted Red, Green, and Blue Channel of the Encrypted Image.

Table 1: Correlation Coefficients of R, G, and B Channels of the Encrypted Color Image of "Aerial".

Component	Horizontal	Vertical	Diagonal
R component	0.0043	−0.0162	0.0117
G component	0.0046	−0.0067	0.0113
B component	−0.0126	−0.0283	0.0213

4.4 Information Entropy

Information entropy is used for measuring the robustness of the algorithm. The entropy value of the ciphered image is relatively equivalent to 8; then the algorithm has the capability of resisting against the entropy attack. Equation (42) gives mathematical expression for information entropy [19, 29, 40].

$$E(r) = -\sum_{i=1}^{2^N-1} P(r_j)\log_2 P(r_j) \tag{25}$$

Table 2: NPCR, UACI, and Entropy Values of Encrypted R, G, and B Channels.

Channels	NPCR	UACI	Entropy
Red	99.6955	28.9173	7.9939
Green	99.7024	27.9796	7.9939
Blue	99.5329	31.6502	7.9920

Here $P(r_j)$ represents the probability of the symbol. Table 2 shows the entropy of the R, G, and B components of the original plaintext and encrypted image.

4.5 Differential Analysis

This differential analysis illustrates the sensitivity of the encryption technique towards the negligible changes [37]. If an intruder does small changes in the original plaintext (for example, 1 pixel) to notice its impact on results, this interruption would result in a major change in the ciphered image. Then, the intruder could not get the connection between the original plaintext and the encrypted image. Thus, the differential attack by an intruder fails. Two measures, NPCR and UACI, are used to test against differential attack. NPCR is an acronym for a number of pixel change rate with respect to a one-pixel change in the original image. UACI is an acronym for unified average changing intensity which demonstrates the average intensity of differences between two encrypted images, corresponding to plain images having only one-pixel difference between them. The NPCR and UACI are expressed as follows:

$$NPCR = \frac{\sum_{i,j} C(i,j)}{A \times B} \tag{26}$$

where $C(i,j)$ is a two-dimensional array which has been of the same size as the encrypted image and $A \times B$ shows the total number of pixels in the original image.

$$UACI = \frac{1}{A \times B}\left[\sum_{i,j} \frac{|C_1(i,j) - C_1(i,j)|}{255}\right] \times 100 \tag{27}$$

where A and B show the width and height of the image, and C_1 and C_2 are the encrypted image before and after one pixel of the color plain image is changed. Table 2 shows the NPCR and UACI of R, G, and B components of the original plaintext and encrypted image.

4.6 PSNR and MSE

The parameters used for measuring encryption and decryption efficiency are mean squared error (MSE) and peak signal-to-noise ratio (PSNR) [4, 44]. MSE measures error present in images. The mathematical expression for calculating mean squared error is defined as follows in Eq. (28).

$$MSE = \frac{1}{N}\sum_{i=1}^{n}(z_i' - z_i) \tag{28}$$

Here, z' is the original plain image, and z is an encrypted image. N is the size of the image.

PSNR is defined as the ratio between peak signal to MSE. The mathematical formula for calculating the PSNR is as follows, defined in Eq. (29).

$$PSNR = 20 \times \log_{10}\left(\frac{I_{max}}{\sqrt{MSE}}\right) \tag{29}$$

Table 3: Algorithm Efficiency Using PSNR and MSE Parameter for R, G, and B Channels.

Channels	PSNR			MSE		
	R	G	B	R	G	B
Encryption efficiency	9.1395	9.4931	8.2573	7.9273e + 03	7.3076e + 03	9.7129e + 03
Decryption efficiency	38.2513	38.9967	35.3901	9.7264	8.1925	8.7962

Table 4: Encryption Time of R, G, and B Components.

Plain image component	Time (s)
Red component	34.4616
Green component	34.4648
Blue component	34.4616
Total encryption time	34.4626

4.6.1 Encryption Efficiency

For encryption process efficiency, the PSNR and MSE are calculated for the original image and the encrypted image. Lower PSNR value and higher MSE value demonstrate the more efficient image encryption. The encryption efficiency is shown in Table 3.

4.6.2 Decryption Efficiency

For decryption process efficiency, the value of PSNR should be higher and MSE is lower between the original and the decrypted images. Table 3 shows the decryption efficiency of the algorithm.

4.7 Encryption Time Analysis

Time is the most important parameter in image encryption [40]. The proposed image encryption algorithm has been implemented on a personal computer with operating system Windows 10 Pro, with processor Intel core i-5 and using MATLAB R2013 as a simulation tool. Table 4 represents the time taken for encryption of the red, green, and blue components of the image.

5 Conclusion

The proposed scheme presents a color image encryption technique using bit-level permutation and four-dimensional Lorenz system. The bit-level permutation is done with help of ACM. The three channels go through the confusion-diffusion process by four-dimensional Lorenz system and ultimately are transformed to an encrypted color image. Also, the tested performance metrics like key space analysis, entropy analysis, and differential analysis reveal that the developed system is highly resistant towards brute-force attack, is more secure, and is highly sensitive towards differential attack, respectively.

The PSNR and MSE are used to calculate efficiency of encryption and decryption process.

Bibliography

[1] A. Akgul, H. Calgan, I. Koyuncu, I. Pehlivan and A. Istanbullu, Chaos-based engineering applications with a 3D chaotic system without equilibrium points, *Nonlinear Dyn.* **84** (2016), 481–495.

[2] A. Akgul, I. Moroz, I. Pehlivan and S. Vaidyanathan, A new four-scroll chaotic attractor and its engineering applications, *Optik* **127** (2016), 5491–5499.

[3] S. Aljawarneh and M. B. Yassein, A resource-efficient encryption algorithm for multimedia big data, *Multimed. Tools Appl.* **76** (2017), 22703–22724.

[4] A. Bisht, M. Dua and S. Dua, A novel approach to encrypt multiple images using multiple chaotic maps and chaotic discrete fractional random transform, *J. Amb. Intel. Hum. Comp.* (2018), 1–13.

[5] G. Chen, Y. Chen and X. Liao, An extended method for obtaining S-boxes based on three-dimensional chaotic Baker maps, *Chaos Soliton. Fract.* **31** (2007), 571–579.

[6] J. X. Chen, Z. L. Zhu, C. Fu, H. Yu and L. B. Zhang, A fast chaos-based image encryption scheme with a dynamic state variables selection mechanism, *Commun. Nonlinear Sci. Numer. Simul.* **20** (2015), 846–860.

[7] A. V. Diaconu, Circular inter–intra pixels bit-level permutation and chaos-based image encryption, *Inf. Sci.* **355** (2016), 314–327.

[8] A. V. Diaconu, V. Ionescu, G. Iana and J. M. Lopez-Guede, A new bit-level permutation image encryption algorithm, in: *Communications (COMM), 2016 International Conference on* (IEEE), pp. 411–416, June 2016.

[9] C. Fu, B. B. Lin, Y. S. Miao, X. Liu and J. J. Chen, A novel chaos-based bit-level permutation scheme for digital image encryption, *Opt. Commun.* **284** (2011), 5415–5423.

[10] A. Gambhir and R. Arya, Performance analysis and implementation of DES algorithm and RSA algorithm with image and audio steganography techniques, in:*Computing, Communication and Signal Processing*, pp. 1021–1028, Springer, Singapore, 2019.

[11] T. Gao and Z. Chen, A new image encryption algorithm based on hyper-chaos, *Phys. Lett. A* **372** (2008), 394–400.

[12] Q. Guo, Z. Liu and S. Liu, Color image encryption by using Arnold and discrete fractional random transforms in IHS space, *Opt. Laser. Eng.* **48** (2010), 1174–1181.

[13] Z. Han, W. X. Feng, L. Z. Hui, L. Da Hai and L. Y. Chou, A new image encryption algorithm based on chaos system, in: *Robotics, intelligent systems and signal processing, 2003. Proceedings. 2003 IEEE international conference on* (IEEE), Vol. 2, pp. 778–782, October 2003.

[14] F. Han, X. Liao, B. Yang and Y. Zhang, A hybrid scheme for self-adaptive double color-image encryption, *Multimed. Tools Appl.* **77** (2017), 1–20.

[15] Z. Hua and Y. Zhou, Image encryption using 2D Logistic-adjusted-Sine map, *Inf. Sci.* **339** (2016), 237–253.

[16] M. Jayashree, I. Poonguzhali and S. S. Agnes, An efficient high throughput implementation of idea encryption algorithm using VLSI, *Aust. J. Basic Appl. Sci.***10** (2016), 337–344.

[17] A. Kanso and M. Ghebleh, A novel image encryption algorithm based on a 3D chaotic map, *Commun. Nonlinear Sci. Numer. Simul.* **17** (2012), 2943–2959.

[18] D. Lambić, A novel method of S-box design based on discrete chaotic map, *Nonlinear Dyn.* **87** (2017), 2407–2413.

[19] C. Li, T. Xie, Q. Liu and G. Cheng, Cryptanalyzing image encryption using chaotic logistic map, *Nonlinear Dyn.* **78** (2014), 1545–1551.

[20] Z. Lin, G. Wang, X. Wang, S. Yu and J. Lü, Security performance analysis of a chaotic stream cipher, *Nonlinear Dyn.* **94** (2018), 1–15.

[21] H. Liu and X. Wang, Color image encryption using spatial bit-level permutation and high-dimension chaotic system, *Opt. Commun.* **284** (2011), 3895–3903.

[22] H. Liu and A. Kadir, Asymmetric color image encryption scheme using 2D discrete-time map, *Signal Process.* **113** (2015), 104–112.

[23] B. Muite and G. Tabia, *Chaos based cryptography*, Tartu University, Tartu, Estonia, 2016.

[24] M. A. Murillo-Escobar, C. Cruz-Hernández, F. Abundiz-Pérez, R. M. López-Gutiérrez and O. A. Del Campo, A RGB image encryption algorithm based on total plain image characteristics and chaos, *Signal Process.* **109** (2015), 119–131.

[25] A. Nag, J. P. Singh, S. Khan, S. Biswas, D. Sarkar and P. P. Sarkar, Image encryption using affine transform and XOR operation. in: *Signal Processing, Communication, Computing and Networking Technologies (ICSCCN), 2011 International Conference on* (IEEE), pp. 309–312, July 2011.

[26] C. Pak and L. Huang, A new color image encryption using combination of the 1D chaotic map, *Signal Process.* **138** (2017), 129–137.

[27] N. K. Pareek, V. Patidar and K. K. Sud, Cryptography using multiple one-dimensional chaotic maps, *Commun. Nonlinear Sci. Numer. Simul.* **10** (2005), 715–723.

[28] V. Patidar, N. K. Pareek and K. K. Sud, A new substitution–diffusion based image cipher using chaotic standard and logistic maps, *Commun. Nonlinear Sci. Numer. Simul.* **14** (2009), 3056–3075.

[29] V. Patidar, N. K. Pareek, G. Purohit and K. K. Sud, Modified substitution–diffusion image cipher using chaotic standard and logistic maps, *Commun. Nonlinear Sci. Numer. Simul.* **15** (2010), 2755–2765.

[30] N. Prasad, V. M. Ravi and L. Chandrasekhar, Image encryption with an encrypted QR, random phase encoding, and logistic map, in: *2018 IEEE International Conference on Computer Communication and Informatics* (*ICCCI*), pp. 1–4, January 2018.

[31] R. M. Rad, A. Attar and R. E. Atani, A new fast and simple image encryption algorithm using scan patterns and XOR, *IJSIP* **6** (2013), 275–290.

[32] M. I. S. Reddy and A. S. Kumar, Secured data transmission using wavelet based steganography and cryptography by using AES algorithm, *Procedia Comput. Sci.* **85** (2016), 62–69.

[33] P. R. Sankpal and P. A. Vijaya, Image encryption using chaotic maps: a survey, in: *Signal and Image Processing (ICSIP), 2014 Fifth International Conference on* (IEEE), pp. 102–107, January 2014.

[34] S. M. Seyedzadeh and S. Mirzakuchaki, A fast color image encryption algorithm based on coupled two-dimensional piecewise chaotic map, *Signal Process.* **92** (2012), 1202–1215.

[35] P. K. Singh, R. S. Singh and K. N. Rai, An image encryption algorithm based on XOR operation with approximation component in wavelet transform, in: *Computer Vision, Pattern Recognition, Image Processing and Graphics (NCVPRIPG), 2015 Fifth National Conference on* (IEEE), pp. 1–4, December 2015.

[36] L. Sui, H. Lu, Z. Wang and Q. Sun, Double-image encryption using discrete fractional random transform and logistic maps, *Opt. Laser. Eng.* **56** (2014), 1–12.

[37] S. Suri and R. Vijay, A synchronous intertwining logistic map-DNA approach for color image encryption, *J. Amb. Intel. Hum. Comp.* (2018), 1–14.

[38] C. J. Tay, C. Quan, W. Chen and Y. Fu, Color image encryption based on interference and virtual optics, *Opt. Laser Technol.* **42** (2010), 409–415.

[39] L. Teng, X. Wang and J. Meng, A chaotic color image encryption using integrated bit-level permutation, *Multimed. Tools Appl.* **77** (2018), 6883–6896.

[40] X. J. Tong, M. Zhang, Z. Wang, Y. Liu, H. Xu and J. Ma, A fast encryption algorithm of color image based on four-dimensional chaotic system, *J. Vis. Commun. Image R.* **33** (2015), 219–234.

[41] A. Tsuneda, Design of binary sequences with tunable exponential autocorrelations and run statistics based on one-dimensional chaotic maps, *IEEE Trans. Circuits Syst.-I* **52** (2005), 454–462.

[42] X. Wang, Y. Zhao, H. Zhang and K. Guo, A novel color image encryption scheme using alternate chaotic mapping structure, *Opt. Laser. Eng.* **82** (2016), 79–86.

[43] J. Wei, X. Liao, K. W. Wong and T. Zhou, Cryptanalysis of a cryptosystem using multiple one-dimensional chaotic maps, *Commun. Nonlinear Sci. Numer. Simul.* **12** (2007), 814–822.

[44] Y. Xing, Q. H. Wang, Z. L. Xiong and H. Deng, Encrypting three-dimensional information system based on integral imaging and multiple chaotic maps, *Opt. Eng.* 55 (2016), 023107.

[45] F. Yu, L. Gao, K. Gu, B. Yin, Q. Wan and Z. Zhou, A fully qualified four-wing four-dimensional autonomous chaotic system and its synchronization, *Optik* **131** (2017), 79–88.

[46] Y. Q. Zhang and X. Y. Wang, A new image encryption algorithm based on non-adjacent coupled map lattices, *Appl. Soft Comput.* **26** (2015), 10–20.

[47] Y. Zhou, L. Bao and C. P. Chen, Image encryption using a new parametric switching chaotic system, *Signal Process.* **93** (2013), 3039–3052.

[48] Y. Zhou, L. Bao and C. P. Chen, A new 1D chaotic system for image encryption, *Signal Process.* **97** (2014), 172–182.

[49] Z. L. Zhu, W. Zhang, K. W. Wong and H. Yu, A chaos-based symmetric image encryption scheme using a bit-level permutation, *Inf. Sci.* **181** (2011), 1171–1186.