pISSN 1976-913X
eISSN 2092-805X

Modified Multi-Chaotic Systems that are Based on Pixel Shuffle for Image Encryption

Om Prakash Verma*, Munazza Nizam* and Musheer Ahmad**

Abstract—Recently, a pixel-chaotic-shuffling (PCS) method has been proposed by Huang et al. for encrypting color images using multiple chaotic systems like the Henon, the Lorenz, the Chua, and the Rossler systems. All of which have great encryption performance. The authors claimed that their pixel-chaotic-shuffle (PCS) encryption method has high confidential security. However, the security analysis of the PCS method against the chosen-plaintext attack (CPA) and known-plaintext attack (KPA) performed by Solak et al. successfully breaks the PCS encryption scheme without knowing the secret key. In this paper we present an improved shuffling pattern for the plaintext image bits to make the cryptosystem proposed by Huang et al. resistant to chosen-plaintext attack and known-plaintext attack. The modifications in the existing PCS encryption method are proposed to improve its security performance against the potential attacks described above. The Number of Pixel Change Rate (NPCR), Unified Average Changed Intensity (UACI), information entropy, and correlation coefficient analysis are performed to evaluate the statistical performance of the modified PCS method. The simulation analysis reveals that the modified PCS method has better statistical features and is more resistant to attacks than Huang et al.'s PCS method.

Keywords—Chaotic Systems, Number of Pixel Change Rate, Unified Average Changed Intensity, Correlation Coefficient, Entropy

1. INTRODUCTION

With the rapid advancement in Internet and multimedia technologies, a number of researchers and scientists have made various attempts to solve the issue of data security and integrity by designing and deploying several encryption/decryption algorithms. Most important aspects of sensitive data are protection against fabrication, unauthorized access, and illegal usage. Encryption is the transformation of data such as text, images, audio, video, etc. into unintelligible forms that provides data confidentiality, integrity, authentication, and non-repudiation. The core idea behind encryption is to protect the valuable or sensitive data that can only be understood after decryption with correct secret key. A good encryption/decryption technique does not distort the original content and is resistant to conventional and other types of cryptographic attacks.

Traditional encryption techniques like Data Encryption Standard (DES), RSA, IDEA etc. are

Manuscript received November 13, 2012; accepted May 12, 2013.
Corresponding Author: Munazza Nizam

* Dept. of Information Technology, Delhi Technological University, New Delhi, India ({munazzanizam23, opverma.dce}@gmail.com)
** Dept. of Computer Engineering, JMI University, New Delhi, India (musheer.cse@gmail.com)

less efficient in encrypting the redundant and bulk-sized multimedia data like images [1-2].

To overcome this challenge, a number of image encryption techniques have been proposed in the literature on image security. Among them, chaos based techniques have gained special attention due to their intrinsic features like stochasticity, dynamic behavior, and sensitivity to initial conditions. A chaos based image encryption algorithm was first proposed by R. Matthews [3]. The highly sensitive response of chaotic system to the initial value conditions and to the variation in parameters makes the chaotic trajectory so unpredictable that a great number of researches implement chaotic sequences to perform encryption of images before transmitting them over an unsecure and open communication network.

2. Related Work

In 2005, L. Zhang et al. [4] proposed a scheme to resist differential attacks by first analyzing the performance of discrete exponential chaotic map and then permuting the pixels of the image. A video encryption method that is based on chaotic maps in Discrete Cosine Transform (DCT) domain is presented in [5].Two coupling chaotic maps were employed for scrambling the DCT coefficient of the original frame and for encrypting the DCT coefficients of the scrambled frame.

The one dimensional chaotic cryptosystem has drawbacks of small key space and weak security. Considering this point, Chong Fu et al. [6] proposed an image encryption technique based on 3D Lorenz chaotic system.

A new encryption scheme was presented in [7], which also employed 3D chaotic systems for bringing confusion and diffusion in encrypted image. It has the advantage of having large key space and low iteration time. In [8], an approach was designed to resist chosen-plaintext cryptanalysis and to protect the secrecy of digital images. The shuffling tables were generated by various logistic maps and the key space was governed by choosing the number of logistic maps. A random number, called ‘nonce’, was introduced to initialize the values of logistic maps. In [9], a scheme for encrypting color images was proposed to increase confusion and diffusion between pixels. It was based on chaotic maps and genetic operations as tools. The sequences were controlled by parameters and given initial values that were considered the key to the encryption technique. In [10], a cryptosystem is proposed that exploits the ergodic property of the simple low-dimensional and chaotic logistic map. However, there were certain drawbacks that were removed in [11] with adjustable sensitivity to initial conditions. A nonlinear chaotic algorithm (NCA) was proposed [12] to get over the drawbacks of one-dimensional linear logistic maps. The algorithm employed nonlinear functions, such as tangent function and power function with improved larger key space and high-level security. Many encryption algorithms have been proposed that use chaotic maps for encryption as their basic tool [13-18], as chaos based methods provide strong encryption strength, better statistical characteristics, and security.

2.1 Motivation

The cryptosystem proposed in [19] applied cipher block chaining (CBC) encryption and was penetrable to chosen-ciphertext attack (CCA) and chosen plaintext attack (CPA) since the keystream generated for every plaintext image was identical. This was cryptanalyzed by Rhouma [20] and in order to make the cryptosystem robust against CCA and CPA, the keystream was updated in a way so that every plaintext image has its own unique shuffling pattern for generation of ciphertext image. In 2009 C. K. Huang and H. H. Nien [21] proposed Pixel Chaotic Shuffle (PCS) encryption method, which is purely based on pixel shuffling using the chaotic sequences generated from 3D multiple chaotic systems. These sequences act as the key for the cryptosystem to perform vertical and horizontal shuffling of the plaintext image bits. However, Rhouma et al.[22] successfully broke the encryption scheme by cracking the sorting sequences that are the keys of the cryptosystem. This was due to the fact that the chaotic sequence used for shuffling the plaintext image bits neither depends on the plaintext image nor on the ciphertext image and the generated key was same for all plaintext images. By analyzing the plain/ciphered image pair, one can deduce the key sequences, and the scheme is prone to chosen-plaintext attack and known-plaintext attack. Thus, in order to make PCS cryptosystem robust against the aforesaid attacks, this paper presents modifications to make the keystream dependent on the plaintext image, which generates the chaotic sequence according to the information contained in the plaintext image. The proposed method generates different chaotic sequences for different plaintext images. The indices of the chaotic sequences are used for the encryption of RGB image by shuffling the bits of the three components column-wise and then row-wise into a pair of two bits.

2.2 Description of PCS cryptosystem

The PCS encryption scheme uses four three-dimensional chaotic systems for pixel shuffling. These chaotic systems are iterated to generate 12 random chaotic sequences. The indices of the sequences are used to map the plaintext image bits. The encryption is executed in two steps. In the first step, the bits of plaintext image are shuffled among themselves using column-wise shuffling. In the second step, the bits are rearranged within pixels of the image using row-wise shuffling. Each color component is shuffled separately using the chaotic sequences.

2.3 Weaknesses in PCS cryptosystem

The cryptanalysis performed in [22] on the cryptosystem [21] shows that the scheme is prone to two different attacks—namely, chosen-plaintext attack and known-plaintext attack. This weakness arises from the fact that same shuffling indices are used to shuffle the plaintext images without taking into account the plaintext image for sequence generation. Thus, the mapping of the pixel bits was same for all plaintext images. A slight change in the pixel values of original image produces negligible change in the respective encrypted image. This results in making the encrypted image easier to comprehend by analyzing the pairs of (plain/ciphered) images.

Another drawback of PCS encryption originates as the RGB components of the image are encrypted separately using the x, y, and z chaotic sequences of the maps, thereby decreasing the randomness of the bits in encrypted image. This facilitates the attacker to apply CPA and KPA to any of the color components of the RGB color image. The attacks are employed onto the 8-bit pixels of each of the color components of the image using the four chaotic sequences generated by the chaotic systems. Decoding the pixel values of color component individually needs lower computation and requires less time to carry out cryptanalysis of the method.

3. Proposed Modified Multi-Chaotic Systems that are Based on Pixel Shuffle Scheme

In this section we propose a modified version of the cryptosystem presented in [21] with a similar basic description and values of the parameters. The modification is performed to deal with the drawbacks mentioned in Section 2.3. The chaotic maps are generated by extracting the information from the plaintext image and are utilized to generate mapping sequences. Thus, an entirely different mapping sequence is used for shuffling each plaintext image. Furthermore, the bits of the RGB components are permuted to increase the randomness and confusion among pixels. Finally, column-wise and row-wise shuffling is performed to render the image totally unrecognizable and unpredictable. The flowchart of the proposed cryptosystem is shown in Fig. 1.

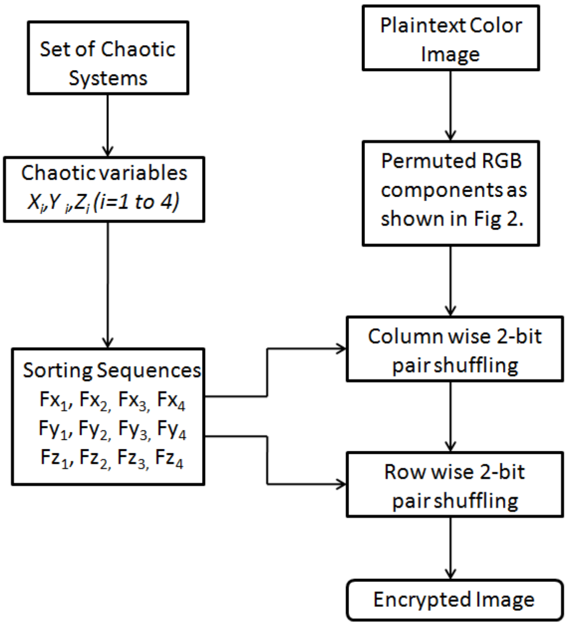

Set of Chaotic Systems
Chaotic variables X_i, Y_i, Z_i (i=1 to 4)
Sorting Sequences
Fx_1, Fx_2, Fx_3, Fx_4
Fy_1, Fy_2, Fy_3, Fy_4
Fz_1, Fz_2, Fz_3, Fz_4
Plaintext Color Image
Permuted RGB components as shown in Fig 2.
Column wise 2-bit pair shuffling
Row wise 2-bit pair shuffling
Encrypted Image

Fig. 1. Flowchart of the Modified Multi-Chaotic Image Encryption

3.1 Plaintext Preparation

The plaintext is a color image of size *m x n* where *m* and *n* are the number of rows and columns in the image respectively. Each pixel is represented as a byte. The plaintext image is first vectorized using row scan method to obtain an array of size *Nx1*, where *N* = *mn*. The array of pixels is further split into its binary equivalents represented in 8-bit format. For example, a pixel with an intensity value of *125* is represented in 8-bit binary format by *01111101*. Similarly, the pixel intensity of the color image is converted into its binary equivalent, forming an array of size *N x 8* for each of the color components.

3.2 Chaotic Map Generation

Let the size of image be *m x n* where *m* and *n* are the number of rows and columns in the image. To make the keystream dependent on the plaintext image, total number of *1s* in binarized color image are calculated.

Let ω be total number of *1s* in the image. It plays the key role in generating different chaotic sequences for different plaintext images. The chaotic maps are evaluated iteratively and their iteration is controlled by:

$$\varphi = \omega + mn \tag{1}$$

Where φ is the total iterations of the chaotic maps. Two images differing from each other by just one pixel will also have entirely different sorting sequences. The chaotic sequence used for shuffling of plaintext image bits are *x(k), y(k),* and *z(k)* where *k* varies from (ω + 1) to φ.

The 3D chaotic maps with coordinates *x, y,* and *z* used in the proposed cryptosystem are as follows [21]:

(1) Hénon map(discrete time):

$$x(k+1) = a - y^2(k) - bz(k) \tag{2a}$$

$$y(k+1) = x(k) \tag{2b}$$

$$z(k+1) = y(k) \tag{2c}$$

where a = 1.76, b = 0.1.

(2) Lorenz (butterfly attractor):

$$\dot{x} = -\sigma x + \sigma y \tag{3a}$$

$$\dot{y} = -xz + \gamma y - y \tag{3b}$$

$$\dot{z} = xy - bz \tag{3c}$$

where σ = 16, γ = 40, b =4.

(3) Chua (double scroll attractor):

$$\dot{x} = \alpha(y - x - h(x)) \tag{4a}$$

$$\dot{y} = x - y + z \tag{4b}$$

$$\dot{z} = -\beta y - \gamma z \tag{4c}$$

$$h(x) = m_1 x + 0.5(m_0 - m_1)(|x + 1||x - 1|)$$

where α = 10, β = 14.78, γ = 0.0385, m_0 = -1.27 and m_1 = -0.68.

(4) Rössler (spiral attractor):

$$\dot{x} = -(y + z) \quad (5a)$$

$$\dot{y} = x + ay \quad (5b)$$

$$\dot{z} = b + z(x - c) \quad (5c)$$

where a = 0.2, b = 0.2 and c = 5.7.

3.3 Permutation of the Plaintext Image Bits

The data in the plaintext image have strong correlation among adjacent pixels. Therefore, the pixels of image are initially permuted amongst themselves in order to decrease the correlation and increase the confusion, as shown in Fig. 2.This increases the dependency of the color components on each other and increases the computation while decoding the information by any unauthorized individual.Ψrgb ($_{mn\ x\ 1}$) is the array of size mn x 1, where each row consists of permuted 24 bits of RGB component.

Thus, the 8-bit RGB pixel that was shuffled individually [21] by 4 chaotic sequences X_1 to X_4, Y_1 to Y_4, and Z_1 to Z_4, respectively, are replaced by 24-bit shuffling of the RGB pixels using 12 chaotic sequences.

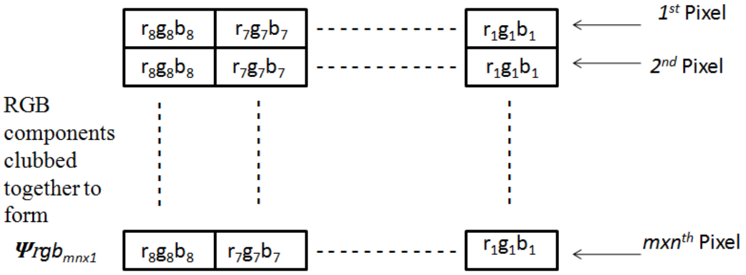


Fig. 2. Permutation of the RGB color components

3.4 Shuffling of the Plaintext Image Bits

Step 1. The chaotic sequencesX_1 *to* X_4, Y_1 *to* Y_4,andZ_1 *to* Z_4, are generated as discussed in Section 3.2.

Step 2. Prepare the chaotic sequences$X_{1(\mu,1)}$ *to* $X_{4(\mu,1)}$,$Y_{1(\mu,1)}$ *to* $Y_{4(\mu,1)}$, and $Z_{1(\mu,1)}$ *to* $Z_{4(\mu,1)}$,which is generated from chaotic variable sets, and make the indexing sequences F_{x1} *to* F_{x4}, F_{y1} *to* F_{y4}, and F_{z1} *to* F_{z4} that are :

$$F_{x1} = sort\ (X_{1(\mu,1)})$$
$$F_{x2} = sort\ (X_{2(\mu,1)})$$
$$F_{x3} = sort\ (X_{3(\mu,1)})$$
$$F_{x4} = sort\ (X_{4(\mu,1)})$$

for $\mu = 1,2,3,\ldots m \times n$ where *sort(·)* is the sequencing index function and *m x n* is the dimension of the original plaintext image. Similarly, we will calculate the sequences F_{x1} *to* F_{x4}, F_{y1} *to* F_{y4}, and F_{z1} *to* F_{z4}, respectively, as in [21, Section 2.2]

Step 3. Combine the original binarized R-level, G-level, and B-level matrixes to form $\Psi rgb_{(mn \times 1)}$, as discussed in Section 3.3. Each row consists of 24-bits of the RGB image.

Step 4. Apply the shuffle function *sq(·)* on the pixels of *Ψrgb* for column indexing and shuffling, as shown in Fig. 3. The function *sq(·)*shuffles and indexes the bits of each pixel by the indexing sequences. Thus, we have the encrypted column shuffled RGB-level matrix as:

$$\psi ergb = [\psi ergb_{\mu 1}, \psi ergb_{\mu 2}, \ldots \psi ergb_{\mu 24}]$$

Where,

$$\psi ergb_{\mu i} = \left\{ \begin{array}{l} sq(\psi rgb_{\mu i}, Fx_1), i = 1,2 \\ sq(\psi rgb_{\mu i}, Fx_2), i = 3,4 \\ sq(\psi rgb_{\mu i}, Fx_3), i = 5,6 \\ sq(\psi rgb_{\mu i}, Fx_4), i = 7,8 \\ sq(\psi rgb_{\mu i}, Fy_1), i = 9,10 \\ sq(\psi rgb_{\mu i}, Fy_2), i = 11,12 \\ sq(\psi rgb_{\mu i}, Fy_3), i = 13,14 \\ sq(\psi rgb_{\mu i}, Fy_4), i = 15,16 \\ sq(\psi rgb_{\mu i}, Fz_1), i = 17,18 \\ sq(\psi rgb_{\mu i}, Fz_2), i = 19,20 \\ sq(\psi rgb_{\mu i}, Fz_3), i = 21,22 \\ sq(\psi rgb_{\mu i}, Fz_4), i = 23,24 \end{array} \right\}$$

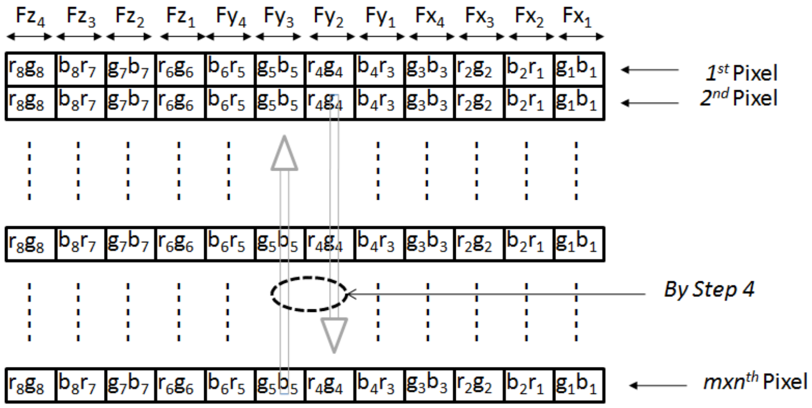

Fz4 Fz3 Fz2 Fz1 Fy4 Fy3 Fy2 Fy1 Fx4 Fx3 Fx2 Fx1
r8g8 b8r7 g7b7 r6g6 b6r5 g5b5 r4g4 b4r3 g3b3 r2g2 b2r1 g1b1
1st Pixel
2nd Pixel
By Step 4
mxnth Pixel

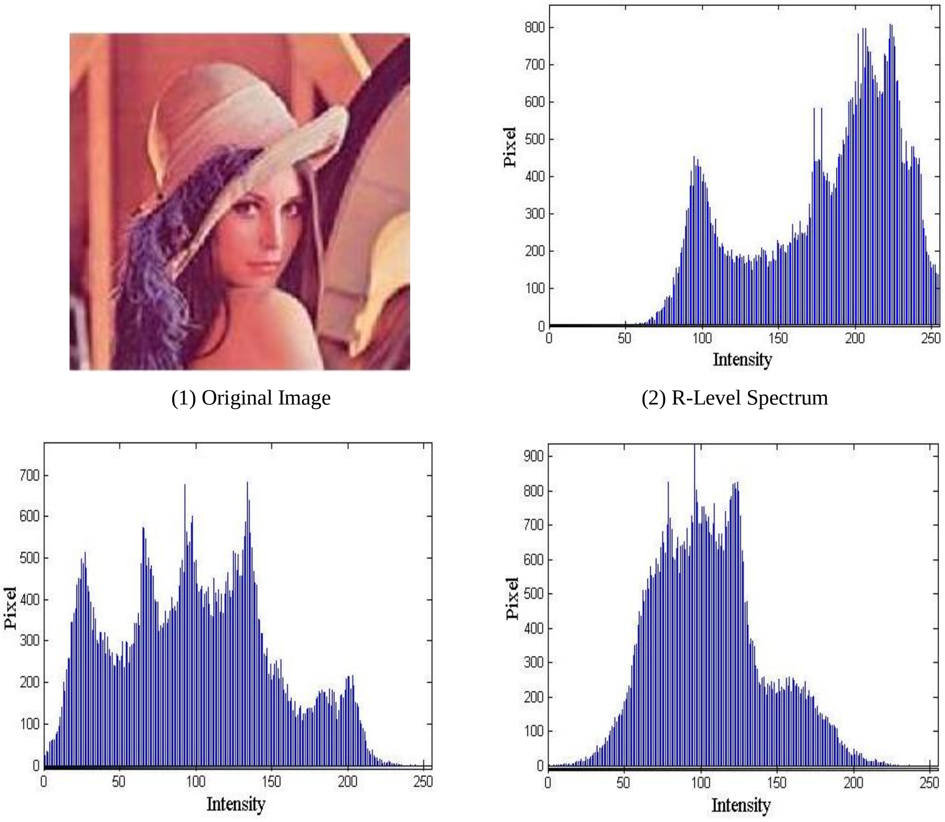
Fig. 3. Column-wise shuffling of the plaintext image bits

and Ψrgb_{u1} is the ith bit of the μth pixel of the original binary RGB-level matrix.

Step 5. Now, perform row-wise shuffling on the bits within each RGB block in pairs of 2-bits by using 12 sorted indices: F_{x1} *to* F_{x4}, F_{y1} *to* F_{y4}, and F_{z1} *to* F_{z4}.

4. Results

In this paper we have taken color images of Lena, Peppers, and Baboon in the size 256 x 256 as our test images to prove the security and robustness of the modified PCS method. The RGB components of the original image, *Lena*, and those obtained from [21] are shown in Fig. 4 and Fig. 5 respectively. The histogram levels show the distribution of pixel intensities in the image. Fig. 6 demonstrates the RGB levels in the encrypted image obtained by using proposed algorithm.



(1) Original Image

(2) R-Level Spectrum

(3) G-Level Spectrum

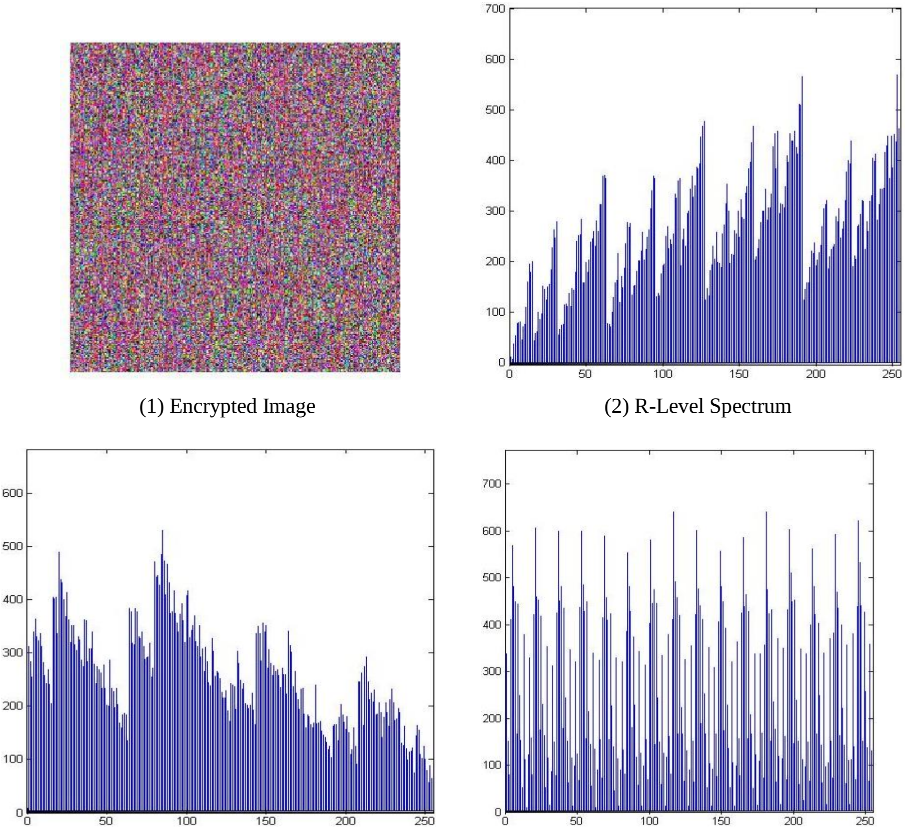
(4) B-Level Spectrum

Fig. 4. Lena and its RGB-Level Spectrums

4.1 Histogram Analysis

An image-histogram illustrates how pixels in an image are distributed by plotting the number of pixels of each color intensity level. The histogram obtained from the PCS technique shown in Fig. 5 has more number of peaks as compared to the proposed scheme. The image with flat histogram level is analogous to a noised image and the ciphered image is indistinguishable when compared with the original image.

The histograms shown in Fig. 6 resembles that of a noisy image and do not reveal any information regarding pixel values of the plaintext image. We can observe that we obtain more flat and fairly uniform histogram using the proposed encryption scheme when compared with PCS cryptosystem.



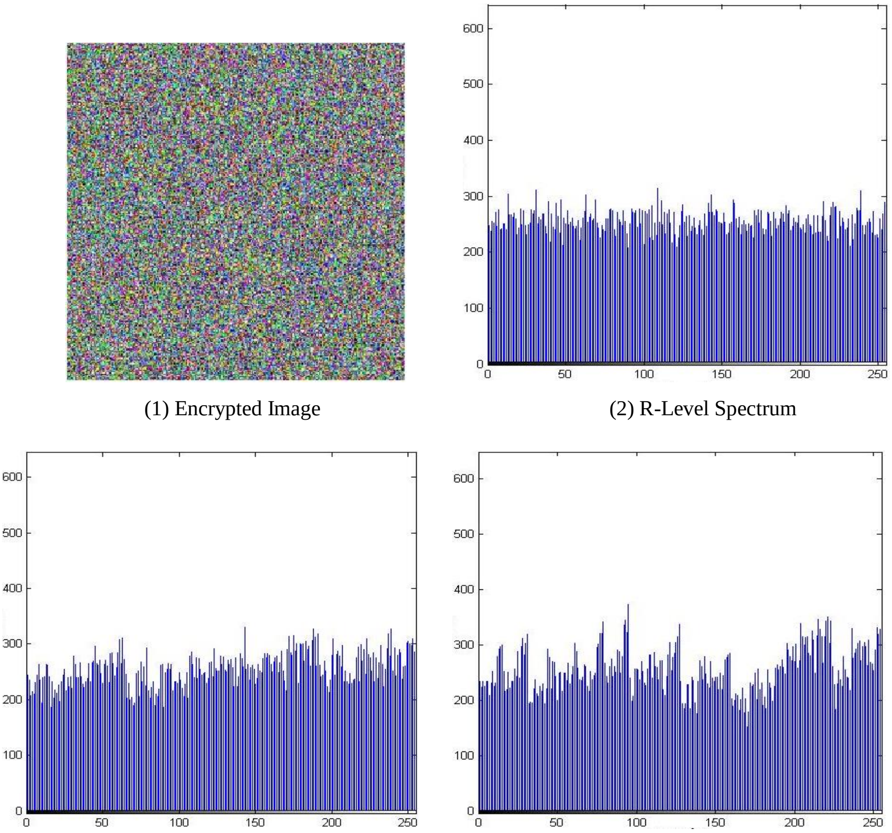
(1) Encrypted Image

(2) R-Level Spectrum

(3) G-Level Spectrum

(4) B-Level Spectrum

Fig. 5. The encrypted image ofLena using the PCS encryption scheme and its RGB-level spectrums



(1) Encrypted Image

(2) R-Level Spectrum

(3) G-Level Spectrum

(4) B-Level Spectrum

Fig. 6. The encrypted image of Lena using the proposed encryption scheme and its RGB-level spectrums

4.2 Entropy Analysis

The entropy of an image is a basic criterion used to depict the randomness of data and the distribution of the information. A greater value of information entropy shows a more uniform distribution of the gray value of the image. The entropy *H* of a symbol source *S* image can be computed by [23].

$$H(S) = \sum_{i=0}^{255} p(s_i)\log\left(\frac{1}{p(s_i)}\right) \tag{6}$$

Where $p(s_i)$ represents the probability of symbol s_i and the entropy is expressed in bits. If the source *S* emits 2^8 symbols with equal probability (i.e., $S = \{S_0, S_1, ...S_{255}\}$) then the result of entropy is $H(S) = 8$, which corresponds to a true random source and represents the ideal value of entropy for message source.

Table 1. Information entropy for the original image and ciphered images

	Original Image	PCS Technique	Our Method
Lena	7.7847	7.6722	7.9838
Peppers	7.7242	7.5510	7.9628
Baboon	7.6310	7.5272	7.9798

If the entropy value tends to 8, then the predictability of the method decreases, which strengthens the image security. Table 1 shows that the entropy value of the proposed system is closer to ideal value of 8 than those computed from PCS encryption. Therefore, the leakage of information through entropy is lesser in the proposed encryption system when compared to PCS technique.

4.3 NPCR and UACI Analysis

The NPCR and UACI are two most significant quantities that are used to evaluate the strength of image encryption algorithms/ciphers to resist different attacks. A sufficiently high NPCR/UACI score is usually considered to be a strong resistance against attacks. NPCR is the measure of absolute number of pixels change rate and UACI determines the averaged difference between two paired ciphertext images when the changes in plaintext images are subtle. In this section, NPCR and UACI values are computed as described below.

4.3.1.NPCR and UACI between the ciphertext images before and after one pixel change

Let the plaintext image be P_1 and the corresponding ciphertext image be C_1. Now, after altering a single pixel value in P_1we get changed plaintext image as,P_2, and its ciphered image as,C_2.

The NPCR and UACI value of the two ciphertext images,$C_1$and C_2, can be mathematically represented by equations (7) and (8), respectively, where T denotes the largest supported pixel value compatible with ciphertext image format,|·| denotes absolute value function, and other symbols have their usual meanings[24].

$$\text{NPCR:}\quad N(C_1,C_2)=\sum_{ij}\frac{D(i,j)}{m\times n}\times 100\% \tag{7}$$

$$\text{UACI:}\,U(C_1,C_2)=\frac{1}{m\times n}\sum_{ij}\frac{|C_1(i,j)-C_2(i,j)|}{T}\times 100\% \tag{8}$$

Where,

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

For example, the test image *Lena* shows that P_1 is encrypted and ciphered image C_1 has been obtained. To calculate NPCR/UACI, a pixel from the plaintext image P_1 is randomly chosen *(P_1(i,j), i= 40, j = 56)*whose R color component is set to *0*, while the value of the other two components remain the same. Let this new image be named P_2 and its corresponding ciphered

Table 2. NPCR and UACI between $C_1$and C_2 for *Lena*

Pixel value changed at position P_1(i,j) here i = 40, j = 56		PCS Encryption Scheme		Proposed Encryption Scheme	
		NPCR%	UACI%	NPCR%	UACI%
P_1(i, j, 1) = 0	R	0.0156	0.0	99.51	33.44
	G	0.0	0.0	99.54	33.85
	B	0.0	0.0	99.57	33.89
P_1(i, j, 2) = 0	R	0.0	0.0	99.51	33.44
	G	0.0	0.0	99.54	33.85
	B	0.0	0.0	99.57	33.89
P_1(i, j, 3) = 0	R	0.0	0.0	99.52	33.10
	G	0.0	0.0	99.44	33.14
	B	0.0	0.0	99.38	33.47

image be C_2.The NPCR and UACI values for $C_1$and C_2 are calculated for the proposed scheme and the original scheme and are listed in Table 2. Similarly, green and blue components are set to zero for the same pixel position *(P_1(i,j), i = 40, j = 56)*and values of NPCR and UACI are evaluated. The results in Table 2 show that a small change in the plain image is reflected by a large difference in the ciphered image.

We can analyze from Table 2 that our scheme is significantly more sensitive towards initial conditions than PCS. The drastic variation in the values of NPCR and UACI arises from the fact that we employ dynamic chaotic maps that vary with varying plaintext images. On the other hand, a constant chaotic sequence for every plaintext image is being used in PCS and it shuffles the pixels to entirely same indices for all plaintext images. In PCS encryption, the two ciphered images obtained by changing one pixel in plaintext image differ from each other at utmost four pixel values, which correspond to the mapping done by the 4 chaotic sequences for the changed pixel value. The proposed technique changes the chaotic map completely and the two ciphered images differ entirely from each other, thus giving a higher score for NPCR and UACI.

4.3.2 NPCR and UACI between the original image and the encrypted image

Let the plaintext image be denoted as *P* and the respective ciphered image obtained by applying proposed encryption algorithm is *C*. The formulae to calculate NPCR and UACI for the two images are represented by following equations:

$$\text{NPCR}: \ N(P,C)=\sum\nolimits_{ij}\frac{D(i,j)}{m\times n}\times 100\% \tag{9}$$

$$\text{UACI}: \ U(P,C)=\frac{1}{m\times n}\sum\nolimits_{ij}\frac{|P(i,j)-C(i,j)|}{T}\times 100\% \tag{10}$$

Where,

$$D(i,j)=\begin{cases}0, if P(i,j)=C(i,j)\\ 1, if P(i,j)\neq C(i,j)\end{cases}$$

Table 3. NPCR and UACI between P and C for *Lena*

		PCS Encryption Scheme		Proposed Encryption Scheme	
		NPCR%	UACI%	NPCR%	UACI%
Lena	R	99.48	24.55	99.62	33.15
	G	99.55	27.51	99.60	30.67
	B	99.67	27.58	99.57	28.27

These values are calculated for the test image *Lena* and tabulated in Table 3. The NPCR and UACI scores determine the randomness between the plaintext image and its ciphertext image. We get a better UACI value for our system and NPCR score is also significantly good.

4.4 Correlation Coefficient Analysis

To withstand statistical attack, the correlation between the adjacent pixels of the cipher image should be as low as possible. For evaluating correlation between pixels in the cipher image we randomly selected 5,000 pairs of adjacent pixels from an image (in horizontal, vertical, and diagonal direction). For precision these values were averaged for 100 iterations. Then, the correlation coefficient of each pair was calculated by [23].

$$\rho = \frac{N\sum_{i=1}^{N}(x_i \times y_i) - \sum_{i=1}^{N} x_i \times \sum_{i=1}^{N} y_i}{\sqrt{(N\sum_{i=1}^{N} x_i^2 - (\sum_{i=1}^{N} x_i)^2) \times (N\sum_{i=1}^{N} y_i^2 - (\sum_{i=1}^{N} y_i)^2)}} \qquad (11)$$

where x and y are gray values of two adjacent pixels in an image and N is the total number of pairs of horizontally, vertically, or diagonally adjacent pixels. The values of correlation coefficients for the proposed algorithm and the PCS scheme are given in Table 4. According to the results, our proposed algorithm has a lesser value for correlation coefficient when compared to PCS scheme. Thus, our algorithm outperforms PCS encryption technique.

Table 4. Correlation coefficients for the test image *Lena*

	Original Image	PCS technique	Our Method
Horizontal	0.90795	0.08622	0.07457
Vertical	0.95298	0.19197	0.12074
Diagonal	0.85709	0.08429	0.08087

4.5 Advantages of the Proposed Encryption Scheme

In the proposed version, the generation of shuffling sequences is made dependent on the pending image information in such a way that a tiny difference in the plaintext image results in distinct shuffling sequences, which in turn produces a totally different encrypted image. Moreover, the components of the pending image are processed collectively and dependently. These improvements make the attacks executed in [22] infeasible and impossible. So, the proposed updated version can resist the chosen-plaintext and known-plaintext attacks effectively.

5. Conclusion

In this paper a new way of image encryption that is robust against CPA and KPA has been proposed. The drawback of PCS technique is overcome by dynamically updating the chaotic map when the plaintext image changes. This ensures the robustness of the proposed algorithm against aforesaid attacks. It is achieved by giving weight to the total number of binary *1s* in the plaintext image and using it in determining the key of the encryption scheme. To make the cipher image more robust against any attack, the color image is permuted before performing the actual column-wise and row-wise shuffling. The experimental values of NPCR and UACI show that the proposed encryption scheme is resistant to differential attacks. Thus, a cryptanalysis of plain/cipher image pair will not reveal the keystream. It is also shown that the ciphered image is very sensitive to a slight change in the bit values of the original image. If one pixel of the plaintext image is changed, then the corresponding cipher image obtained is altered completely in an unpredictable or pseudorandom manner. Several tests have been performed to evaluate the security of the proposed system. Namely, statistical analysis, which includes Histogram analysis; correlation analysis; NPCR and UACI analysis; and information entropy analysis. The experimental values that govern the proposed algorithm yield a very good performance over the existing PCS algorithm. Hence, this paper presents a cryptosystem that is highly secure against attacks and is useful for secure image encryption and transmission applications.

References

[1] G. Chen, Y. Mao, C. Chui, “A symmetric image encryption scheme based on 3D chaotic cat maps”, *Chaos, Solitons & Fractal* , Vol.21, Issue 3, 2004, pp.749–61.

[2] F. Chiaraluce, L. Ciccarelli, “A new chaotic algorithm for video encryption,”*IEEE Transactions on Consumer Electronics,* Vol.48, Issue 4, 2002, pp.838–44.

[3] R. Matthews, “On the derivation of a chaotic encryption algorithm”, *Cryptologia,* Vol.13, No. 1, Jan 1989, pp.29-41.

[4] L. Zhang, X. Liao, X. Wang, “An image encryption approach based on chaotic maps,” *Chaos Solitons Fractals,* Vol.24, Issue 3, 2005, pp.759 -765.

[5] S. Yang, S. Sun, “A video encryption method based on chaotic maps in DCT domain”, *Progress in Natural Science;* Vol.18, 2008, pp.1299–1304.

[6] C. Fu, Z. Zhang, Y. Cao, “*An improved image encryption algorithm based on chaotic maps,”Procedings of the 3rd Int. Conf. on Natural Computation, (ICNC 2007):* Haikou, Vol.3, pp.189-193.

[7] K. Sakthidasan, B. V. K. Santhosh, “A New Chaotic Algorithm for Image Encryption and Decryption of Digital Color Images”, *International Journal of Information and Education Technology,* Vol.1, No. 2, 2011, pp.137-141.

[8] W. Li, N. Yu, *“A Robust chaos based Image encryption scheme”, IEEE International Conference on Multimedia and Expo:* New York, 2009, pp.1034-1037.

[9] M. El-Sayed, A. El-Alfy; Khaled, Al-Utaibi, *“An Encryption Scheme for Color Images Based on-Chaotic Maps and Genetic Operators”, The Seventh International Conference on Networking and Services,* Venice/Mestre, Italy, 2011, pp.92-97.

[10] M. S. Baptista, “Cryptography with chaos”, *Phys Lett A,* Vol.240, Issue 1-2 , 1998, pp.50–54.

[11] P. L. de Oliveira Luiz, and S. Marcelo, “Cryptography with chaotic mixing” *Chaos, Solitons and Fractals* , Vol.35, Issue 3, 2008, pp.466–471.

[12] H. Gao, Y. Zhang, S. Liang, D. Li, “A new chaotic algorithm for image encryption”, *Chaos, Solitons and Fractals,* Vol.29, Issue 2, 2006, pp.393–399.

[13] Y. Zhai, S. Lin, Q. Zhang, *“Improving Image Encryption Using Multi-chaotic Map”,* Workshop on

Power Electronics and Intelligent Transportation System (PEITS): Guangzhou, 2008, pp.143-148.

[14] G. Xin, L. Fen-lin, L. Bin, W. Wei, C. Juan, *"An Image Encryption Algorithm Based on Spatiotemporal Chaos in DCT Domain", The 2nd International Conference on Information Management and Engineering(ICIME):* Chengdu,2010, pp.267-270.

[15] H. H. Nien, C. K. Huang, S. K. Changchien, H. W. Shieh, C. T. Chen, Y. Y. Tuan, "Digital color image encoding and decoding using a novel chaotic random generator," *Chaos, Solitons and Fractals,* Vol.32, Issue 3, 2007, pp.1070-1080.

[16] K. Wang, W. Pei, L. Zou, A. Song, Z. He, "On the security of 3D Cat map based symmetric image encryption scheme,"*Phys. Lett. A,* Vol.343, Issue 6, 2005, pp.432-439.

[17] Z. H. Guan, F. Huang, W. Guan, "Chaos-based image encryption algorithm," *Phys. Lett. A,* Vol.346, Issue 1-3, 2005, pp.153-157.

[18] T. G. Gao, Z.Q. Chen, "A new image encryption algorithm based on hyper-chaos," *Phys. Lett. A,* Vol.372, Issue 4, 2005, pp.394-400.

[19] S. Lian, "Efficient image or video encryption based on spatiotemporal chaos system", *Chaos Solitons Fractals,* Vol.40, Issue 5, 2009, pp.2509-2519.

[20] R. Rhouma, S. Belghith, "Cryptanalysis of a spatiotemporal chaotic image/video cryptosystem", *Phys. Lett. A,* Vol.372, Issue 36, 2008, pp.5790-5794.

[21] C. K. Huang, H. H. Nien, "Multi chaotic systems based pixel shuffle for image encryption,*" Optics Communications,* Vol.282, Issue 11, 2009, pp.2123–2127.

[22] E. Solak, R. /Rhouma, S. Belghith, "Cryptanalysis of a multi-chaotic systems based image cryptosystem", *Optics Communications, Vol.*283, Issue 2, 2010, pp.232–236.

[23] M. Ahmad, O. Farooq, *"A Multi-Level Blocks Scrambling Based Chaotic Image Cipher", Communications in Computer and Information Science,*Noida -India, Vol.94, Part 1, 2010, pp.171-182.

[24] Y. Wu, J. P. Noonan, S. Agaian, "NPCR and UACI Randomness Tests for Image Encryption",*Cyber Journals: Multidisciplinary Journals in Science and Technology, Journal of Selected Areas in Telecommunications (JSAT),* April Edition, 2011, pp.31-38.

Om Prakash Verma

Om Prakash Verma received his B.E. degree in Electronics and Communication Engineering from Malaviya National Institute of Technology, Jaipur, India, M. Tech. degree in Communication and Radar Engineering from Indian Institute of Technology (IIT), Delhi, India and Ph.D. degree from Delhi University. From 1992 to 1998 he was Assistant Professor in Department of Electronics & Communication Engineering, at Malaviya National Institute of Technology, Jaipur, India. He joined Department of Electronics & Communication Engineering, Delhi Technological University (Formerly Delhi College of Engineering), Delhi, India, as Associate Professor in 1998. Currently, he is Professor and Head of Department of Information Technology and Dean continuing education at Delhi Technological University Delhi. He has authored a book on Digital Signal Processing in 2003. His research interest includes Computer Vision and Image Processing, Application of Soft Computing techniques in Image Processing, Artificial Intelligent, Optimization techniques, Digital Signal Processing etc. He has published more than 30 research papers in International Journal and conference proceedings. He has guided 30 M. Tech. Students and presently 5 Ph.D. scholars are working under his supervision. He is Principal investigator of "Information Security Education Awareness" Project. This project is sponsored by Department of Information Technology, Ministry of MHRD, Govt. of India.

Munazza Nizam

Munazza Nizam received her B.Tech degree in Computer Engineering from Jamia Millia Islamia University, Delhi, India in 2010 and M.Tech degree in Information Systems from Delhi Technological University (Formerly Delhi College of Engineering), Delhi, India in 2012. She joined Deloitte Consulting India Pvt. Ltd in 2012 as Business Technology Analyst to develop ERP solutions. Currently she is working as Assistant Professor in Dept. of Computer Science and Engineering in Galgotias University, India. Her area of research includes Image Processing.

Musheer Ahmad

Musheer Ahmad received his B.Tech and M.Tech degrees from Department of Computer Engineering, ZH College of Engineering and Technology, AMU, India in 2004 and 2008, respectively. He joined the Department of Computer Engineering, AMU, India as a Faculty in 2006.Currently, he is with Department of Computer Engineering, Jamia Millia Islamia, New Delhi, India, as an Assistant Professor. He has published about 20 research papers in refereed academic journals and international conference proceedings. His areas of research interest include multimedia security, chaos-based cryptography, image processing and soft computing techniques.