Journal of Multimedia and Information System Vol. 2, No 2, June 2015(pp. 193-206): ISSN 2383-7632(Online)
http://dx.doi.org/10.9717/JMIS.2015.2.2.193

Image Encryption Based on Quadruple Encryption using Henon and Circle Chaotic Maps

Author: Gururaj Hanchinamani [1,*], Linganagouda Kulkarni[2]

Abstract

In this paper a new approach for image encryption based on quadruple encryption with dual chaotic maps is proposed. The encryption process is performed with quadruple encryption by invoking the encrypt and decrypt routines with different keys in the sequence $EDEE$. The decryption process is performed in the reverse direction $DDED$. The key generation for the quadruple encryption is achieved with a 1D Circle map. The chaotic values for the encrypt and decrypt routines are generated by using a 2D Henon map. The Encrypt routine E is composed of three stages i.e. permutation, pixel value rotation and diffusion. The permutation is achieved by: row and column scrambling with chaotic values, exchanging the lower and the upper principal and secondary diagonal elements based on the chaotic values. The second stage circularly rotates all the pixel values based on the chaotic values. The last stage performs the diffusion in two directions (forward and backward) with two previously diffused pixels and two chaotic values. The security and performance of the proposed scheme are assessed thoroughly by using the key space, statistical, differential, entropy and performance analysis. The proposed scheme is computationally fast with security intact.

Key Words: Quadruple encryption, Circle map, Henon map, Differential attacks

I. INTRODUCTION

With the rapid development in the internet and multimedia communication, a great deal of concerns have been raised in the security of multimedia information transmitted or stored over the open networks. Ensuring security to the multimedia information is a vital issue. A way out is to encrypt the information so that only authorized persons with a valid key can use the information. A number of encryption schemes have been proposed and widely used such as the DES, AES, LUCIFER, FEAL, IDEA etc. However, these algorithms are suitable for text information but not for the images due to the special features of the images such as the high correlation, high redundancy and bulk volume of data [1-10].

Chaotic map based encryption schemes typically possesses high speed with low cost, which makes them better candidates than the traditional cryptosystems for the multimedia encryption [1-5]. Moreover, the encryption algorithms based on the chaotic systems offer the advantages to be very sensitive to the initial conditions, ergodicity, randomness and simplicity. Hence, there is a growing research interest from the cryptographers on the chaotic map based cryptosystems.

Recently, a number of chaotic based image encryption schemes have been proposed; however, the majority of them have their own strengths and weaknesses in terms of the security and performance [1-5]. This paper proposes an efficient image encryption scheme based on quadruple encryption with Henon and Circle maps. The proposed scheme is resistant to a variety of attacks such as the brute force, statistical, differential and entropy and is computationally fast over the other schemes in the literature.

The novelty of this paper is listed below.

i) The quadruple encryption is explored for the images with four different keys.

ii) The Circle and Henon map are explored together for the image encryption.

iii) The encryption operations are carried with new methods.

iv) The proposed image encryption/decryption scheme is computationally fast.

Manuscript received May 31, 2015 ; Revised June 10 ; Accepted June 29, 2015. (ID No. JMIS-2005-11)

Corresponding Author(*): Gururaj Hanchinamani , College of Engineering and Technology, Hubli-580031, Dharwad, Karnataka, India, 091 9341961485, gs_hanchinamani@bvb.edu.

[1]BVB College of Engineering & Technology, Hubli, India , gs_hanchinamani@bvb.edu

[2]BVB College of Engineering & Technology, Hubli, India , Region, Country, linganagouda@yahoo.co.uk

The rest of the paper is organized as follows. In section 2, the literature survey is presented. The 1D circle map and 2D Henon map are discussed in section 3. In section 4, the proposed encryption scheme is described in detail. Experimental results and the security analysis are presented in section 5 to show the efficacy and validity of the algorithm. Finally, the conclusions are drawn in the last section.

II. LITERATURE SURVEY

Chaotic based image cryptosystems typically consist of iteration of two processes: permutation and diffusion [1-10]. The permutation stage is employed to decorrelate the adjacent pixels and the diffusion stage is employed to spread the effect of individual pixels across the entire image [1-5]. However, as many rounds of permutation and diffusion or iterations should be taken, the overall encryption speed is slow. A short summary of the recently proposed chaotic based encryption schemes is given hereafter.

The authors of [1], proposed an image encryption scheme based on a generalized Arnold cat map. The chaotic sequences are generated by employing two generalized Arnold cat maps. The typical permutation diffusion architecture is employed. The scheme has three parts: circular permutation, positive diffusion and opposite diffusion. The authors of [2], employed multiple chaotic maps to increase the key space. The chaotic maps used are: Logistic map, Cubic map, Sin map and Tent map. The scheme consists of three modules: Key expansion, encryption and decryption. The encryption scheme consists of eight w-bit registers $R_i(i = 1,2, \dots ,8)$, which contain the initial input plain image as well as the output cipher image at the end of the encryption process. The encryption is performed by using a *xor* and the pixel value circular rotations. In [3], an image encryption scheme based on the transformed Logistic maps is presented. The encryption operations employed are: permutation of all pixels based on keys, nonlinear diffusion and *xor*. The authors of [4], presented an image encryption scheme based on coupling of chaotic maps and an *xor* operator. The encryption process is straightly realized on the bits of the plain image. The image is shuffled by using a chaotic function based on linear congruence. In [5], a combinational domain image encryption scheme is presented that encrypts important data in spatial domain and the irrelevant data in wavelet domain. The proposed scheme considers the varying data significance and supports dissimilar security level to regions of the different significance. The scheme makes use of Prewitt edge detector, discrete wavelet transform, Arnold cat map and Logistic map. In [6], an image encryption scheme based on an intertwining chaotic map is proposed. The scheme uses multiple chaotic maps to increase the key space. The authors of [7], proposed an image encryption scheme based on chaotic Chebyshev generator. The scheme employs multiple permutations and a diffusion step. The permutation is based on row and column scrambling. The diffusion is based on addition and *xor* operations.

The authors of [8], proposed an image encryption scheme based on self adaptive wave transmission. The self adaptive encryption is implemented by using one half of the image to encrypt the other half of the image reciprocally. In [9], a symmetric encryption scheme based on cyclic elliptic curve and chaotic system is presented. The round keys are generated based on piecewise nonlinear chaotic maps. The authors of [10] proposed a gray level encryption scheme to eliminate the image outlines and to disrupt the distributional characteristics of gray level. The scheme makes use of Chua chaotic system. In [11], an image encryption scheme based on a large pseudorandom permutation is proposed which is combinatorially generated from small permutation matrices based on the chaotic maps. The authors of [12], introduced a hierarchy of 2D piecewise nonlinear chaotic maps with an invariant measure. In [13], chaos based image encryption is merged with pixel bit. The scheme uses a single chaotic system, which is applied directly to the position scrambling encryption; moreover, it also performs the gray encryption at the same time. In [14], a common framework of guidelines for developing the image cryptosystems is provided, and it addresses three issues: implementation, key management and security analysis. The authors of [15], proposed an image cipher based on substitution diffusion architecture by using Standard and Logistic chaotic maps. In [16], an image encryption scheme based on two Logistic maps and an external 80-bit key is proposed. The initial conditions for the Logistic maps are derived by using external keys. The encryption is performed by using eight different operations.

Though, there exist several image encryption schemes in the literature, each of them has their own strengths and limitations more or less in terms of the security level and the computational performance. Majority of the schemes produce satisfactory security results but lack in the computational speed. This paper proposes a novel chaotic image encryption scheme based on the quadruple encryption by using Henon and Circle maps. The proposed scheme is computationally fast and also provides satisfactory security level.

III. CHAOTIC MAPS

Chaotic maps are nonlinear maps that are sensitive to initial conditions and parameters, non-periodic, non-convergent, and topologically mixing [1-5]. The proposed scheme uses two chaotic maps: 1D circle map and 2D Henon map are discussed hereafter.

The 1D circle map is a discrete-time dynamical system, and is defined as,

$$Z_{i+1} = \sqrt{p_1 + p_2 \times Z_i + \sin 2\pi p_3 Z_i} \qquad (1)$$

where Z_i is the current chaotic value, Z_{i+1} is the next chaotic value and p_1, p_2, p_3 are the control parameters. The key set of the Circle map is $\{Z_0, p_1, p_2, p_3\}$. In the proposed scheme the Circle map is used to generate the keys for the Henon map.

The 2D Henon map is a discrete-time dynamical system, and is defined as,

$$X_{i+1} = 1 - a \times X_i^2 + Y_i \qquad (2)$$

$$Y_{i+1} = b \times X_i \qquad (3)$$

where X_i,Y_i are the current chaotic values and X_{i+1}, Y_{i+1} are the next chaotic values and a, b are the control parameters. The key set for the Henon map is $\{X_0, Y_0, a, b\}$. In the proposed scheme, the Henon chaotic values are used during the permutation, pixel value rotation and the diffusion stages of the encrypt and decrypt functions.

The propositions of the chaotic maps are given in Eq.(4-6). The chaotic outputs of the above two chaotic maps are analyzed by computing the mean and the self-correlations according to the propositions given in Eq.(4-6). It is observed that the mean values are close to 0.5 and the self correlations within the sequence and across the two sequences are very close to 0.

Proposition 1. The mean value of the chaotic sequence is defined as,

$$x_{mean} = \lim_{N\to\infty} \frac{1}{N} \sum_{k=0}^{N-1} x_k = 0.5 \qquad (4)$$

Proposition 3. Correlation function between two chaotic sequences is given as,

$$S2(\beta) = \lim_{N\to\infty} \frac{1}{N} \sum_{k=0}^{N-1} (x_k - x_{mean})(y_k - y_{mean}) = 0 \qquad (6)$$

IV. PROPOSED ENCRYPTION SCHEME

The proposed encryption scheme is based on multiple encryption, specifically quadruple encryption. The reason being; it is stronger than the triple encryption against the man-in-the-middle attack [17]. Multiple encryption is one combination technique which uses an algorithm to encrypt the same plaintext multiple times with multiple keys. The proposed quadruple encryption operates on a plain image and then on modified a plain image four times with four different keys. The sender, first encrypts, with the first key, then decrypts with the second key, then encrypts with the third key and finally encrypts with the fourth key according to Eq.(7). This is called as encrypt-decrypt-encrypt-encrypt (EDEE) mode. The receiver decrypts with the fourth key, then decrypts with the third key, then encrypts with the second key and finally decrypts with the first key according to Eq.(8).

$$C = E_{K_4}\left(E_{K_3}\left(D_{K_2}\left(E_{K_1}(P)\right)\right)\right) \qquad (7)$$

$$P = D_{K_1}\left(E_{K_2}\left(D_{K_3}\left(D_{K_4}(C)\right)\right)\right) \qquad (8)$$

where P is a plain image, C is a cipher image, E is an encrypt function, D is a decrypt function and K_1, K_2, K_3, K_4 are the keys at respective invocations.

4.1. Key generation and scheming

The keys K_1, K_2, K_3, K_4 are used to generate the Henon chaotic values. The keys can be described as,

$$\begin{aligned} K_1 &= \{ X_1, Y_1, a_1, b_1 \} \\ K_2 &= \{ X_2, Y_2, a_2, b_2 \} \\ K_3 &= \{ X_3, Y_3, a_3, b_3 \} \\ K_4 &= \{ X_4, Y_4, a_4, b_4 \} \end{aligned} \qquad (9)$$

where X_i, Y_i are the initial values and a_i, b_i are the parameters of the Henon map. So, the key comprises of totally 16 values $\{ X_i, Y_i,\ a_i, b_i, i = 1\ to\ 4\}$.. It is hard to remember the above key set. Hence, the following key scheming approach is used to limit the key length. Generate eight chaotic values $(c_1\ to\ c_8)$ by using Circle map according to Eq.(1), then these chaotic values are assigned to $\{ X_1, Y_1, X_2, Y_2, X_3, Y_3, X_4, Y_4\}$. The a and b parameters are kept the same for all the four invocations of the encrypt/decrypt functions.

So the modified keys can be described as,

$$\begin{aligned} K_1 &= \{X_1 = c_1, Y_1 = c_2\ , a, b\} \\ K_2 &= \{X_2 = c_3, Y_2 = c_4\ , a, b\} \\ K_3 &= \{X_3 = c_5, Y_3 = c_6\ , a, b\} \\ K_4 &= \{X_4 = c_7, Y_4 = c_8\ , a, b\} \qquad (10) \end{aligned}$$

So the key set for the quadruple encryption/decryption process is,

$$Key = \{Z_0, p_1, p_2, p_3, a, b\} \qquad (11)$$

4.2. Encrypt function

The encrypt function consists of three stages: permutation, pixel value rotation and diffusion and are discussed hereafter.

4.2.1. Permutation

The purpose of the permutation is to decorrelate the adjacent pixels in the plain image. Let *I* be a gray original image of size $M \times N$, it is a matrix containing M rows and N columns, and the gray values ranges from 0 to 255. The following are the operations performed in this stage.

1 Initially $M + N$ Henon chaotic values $\{(X_1, \dots, X_M), (Y_1, \dots, Y_N)\}$ are generated by using Eq.(2,3), after doing iterations in chaos maps. Let $PM = \{X_1, \dots, X_M\}$ and $PN = \{Y_1, \dots, Y_N\}$. Then PM and PN are sorted, and the positions of the sorted chaotic values in the original chaotic sequence are found and stored in PM' and PN'. The next step is to shuffle the row position of all the values from the first column to last the column according to $PM'_1, \dots, PM'_M$. Similarly shuffle the column position of all the values from the first row to the last row according to $PN'_1, \dots, PN'_N$.

2 Generate $M \times N$ Henon chaotic values $\{(X_1, \dots, X_{M\times N}),\ (Y_1, \dots, Y_{M\times N})\}$ by using Eq.(2,3) after doing iterations in chaos maps. Then, exchange the lower principal diagonal element with the corresponding upper principal diagonal element based on the chaotic condition $X_i > Y_i$.

3 Exchange the lower secondary diagonal element with the corresponding upper secondary diagonal element based on the chaotic condition $X_i < Y_i$.

This stage shuffles all pixels and decorrelates the neighboring pixels.

4.2.2. Pixel value rotation

Initially, the real chaotic values are transformed into the integer form with the following transform.

$$X'_i = (\ X_i * \ 10^8\ )\ mod\ m \qquad (12)$$

where X_i is the real chaotic value, X'_i is the transformed integer value, and m is 256 for 8-bit gray image.

The circular rotations are applied on pixel by pixel basis as follows.

Circularly rotate right 3 times, if $0 \leq X'_i \leq 63$
Circularly rotate left 2 times, if $64 \leq X'_i \leq 127$
Circularly rotate right 5 times, if $128 \leq X'_i \leq 191$
Circularly rotate left 6 times, if $192 \leq X'_i \leq 255$ (13)

where $\{X'_i, i = 1\ to\ M \times N\}$ are the chaotic values in the integer form.

4.2.3. Deffusion

The diffusion function is used to ensure the plain image sensitivity i.e., a 1-bit change in any one pixel of the plain image should spread out to almost all pixels in the whole image.

The 2D permuted image is transformed into 1D array $P_{1\times MN}$ by scanning the image from left to right and from top to bottom. The diffusion process is performed in two directions (forward and backward) with two previously diffused pixels and two chaotic values of the Henon map $\{X, Y\}$. The computed and encrypted pixel values depend on the previously encrypted pixels and chaotic sequences, hence the algorithm shows the resistance against the differential attacks.

The forward diffusion is performed by using the following equation,

$$E_i = \left(((((P_i + E_{i-2}\ )mod\ 256\ ) + E_{i-1})mod\ 256 \oplus X_i\right) + Y_i)mod\ 256, i = 1,2, \dots, MN \qquad (14)$$

where $+$ indicates the modulo addition, $\oplus$ is the bitwise XOR, E_i is the current pixel, E_{i-1} and E_{i-2} are the previously encrypted pixels, P_i is the permuted pixels, X_i and Y_i are the 2D Henon chaotic values and E_{-1} and E_0 can be considered as the constants.

The backward diffusion is performed by using the following equation to make the influence of every pixel equal.

$$F_i = \left(((((E_i + F_{i+2}\ )\ mod\ 256\ ) + F_{i+1})mod\ 256 \oplus X_i\right) + Y_i)\ mod\ 256, i = MN, \dots, 1 \qquad (15)$$

where F_i is the current pixel, F_{i+1} and F_{i+2} are the previously encrypted pixels, E_i is the forward diffused image pixel, X_i and Y_i are the 2D Henon chaotic values and E_{MN+1} and E_{MN+2} can be considered as constants. Finally, the encrypted image is obtained after the diffusions using Eq.(14,15) in two directions.

4.3 Decrypt function

Decryption involves the reconstruction of the gray levels of the original image from the encrypted image. It is an inverse process of the proposed encrypts function. Initially, the diffusion process is carried out, first backward then forward. Then, the pixel values are circularly rotated in reverse rotate direction. Lastly, the permutation steps are carried out in reverse order.

The reverse forward diffusion is calculated by Eq.(16), and the reverse backward diffusion with Eq.(17).

$$P_i = \Bigg(\bigg(\Big(\big((E_i - Y_i) mod\ 256 \oplus X_i\big) - P_{i+1}\Big) mod\ 256\bigg) - P_{i+2}\Bigg) mod\ 256, i = MN, \dots, 1 \quad (16)$$

$$E_i = \Bigg(\bigg(\Big(\big((F_i - Y_i) mod\ 256 \oplus X_i\big) - E_{i-1}\Big) mod\ 256\bigg) - E_{i-2}\Bigg) mod\ 256, i = 1, \dots, MN \quad (17)$$

4.4 Encryption algorithm

The encryption algorithm makes use of the following three routines: Main routine for the encryption process, Encrypt routine and Decrypt routine.

4.4.1 Main routine for the encryption process

1. Read the original image and store the pixel values in the matrix $I_{M \times N}$.
2. Initialize the Circle map and generate eight chaotic values (c_1 to c_8) by using Eq.(1).
3. Initialize four Keys K_1, K_2, K_3, K_4 by using Eq.(10).
4. Invoke the Encrypt function with K_1 for an original image.
5. Invoke the Decrypt function with K_2 for a modified image.
6. Invoke the Encrypt function with K_3 for a modified image.
7. Invoke the Encrypt function with K_4 for a modified image.

4.4.2 Encrypt function

The encrypt function is composed of the following fifteen steps.

1. Get the image.
2. Generate M chaotic values of the X_i sequence $(X_1, \dots, X_M)$, and N chaotic values of the Y_i sequence $(Y_1, \dots, Y_N)$ using Eq.(2,3).
3. Copy X_i chaotic values to PM and Y_i chaotic values to PN.
4. Sort PM and PN, find the position of the sorted chaotic values in the original chaotic sequence and store in PM' and PN'.
5. Shuffle all the rows using PM'.
6. Shuffle all the columns using PN'.
7. Generate $M \times N$ chaotic values $\{(X_1, \dots, X_{M \times N}), (Y_1, \dots, Y_{M \times N})\}$ using Eq.(2,3).
8. Exchange the lower principal diagonal element with the corresponding upper principal diagonal element if $X_i > Y_i$, Repeat this for all the lower diagonals.
9. Exchange the lower secondary diagonal element with the corresponding upper secondary diagonal element if $X_i < Y_i$, Repeat this for all the lower diagonals.
10. Transform the real chaotic values into integers using Eq.(12).
11. Perform left or right circular rotations based on the chaotic values for all the pixels according to Eq.(13)
12. Transform the 2D processed image into 1D array i.e. dimension transform from $M \times N\ to\ 1 \times MN$.
13. Perform the forward diffusion using Eq.(14).
14. Perform the backward diffusion using Eq.(15).
15. Transform the 1D encrypted array into 2D array i.e. dimension transform from $1 \times MN\ to\ M \times N$.

4.4.3 Decrypt function

The decrypt function is composed of the following fifteen steps.

1. Get the image.
2. Generate $M \times N$ chaotic values $\{(X_1, \dots, X_{M \times N}), (Y_1, \dots, Y_{M \times N})\}$ using Eq.(2,3).
3. Transform the real chaotic values into integers using Eq.(12).
4. Transform the 2D image into 1D array, i.e. dimension transform $M \times N\ into\ 1 \times MN$.
5. Perform the reverse backward diffusion using Eq.(17).
6. Perform the reverse forward diffusion using Eq.(16).
7. Transform the 1D array into 2D array, i.e. dimension transform $1 \times MN\ into\ M \times N$.
8. Change the direction of rotation and perform left or right circular rotations based on the chaotic values for all pixels according to Eq.(13).
9. Exchange the lower secondary diagonal element with the corresponding upper secondary diagonal element if $X_i < Y_i$, Repeat this for all the lower diagonals.
10. Exchange the lower principal diagonal element with the corresponding upper principal diagonal element if $X_i > Y_i$, Repeat this for all the lower diagonals.
11. Generate M chaotic values of the X_i sequence $(X_1, \dots, X_M)$, and N chaotic values of the

```
    $Y_i$ sequence $(Y_1, \dots, Y_N)$ using Eq.(2,3).
12  Copy $X_i$ chaotic values to $PM$ and $Y_i$ chaotic values to $PN$.
13  Sort $PM$ and $PN$, find the position of the sorted chaotic values in the original chaotic sequence and store in $PM'$ and $PN'$.
14  Shuffle all the columns using $PN'$.
15  Shuffle all the rows using $PM'$.
```

4.5 Decryption algorithm

The decryption algorithm makes use of three routines: Main routine for the decryption process, Encrypt routine and Decrypt routine.

4.5.1 Main routine for the decryption process

```
1  Read the encrypted image and store the pixel values in the matrix $EN_{M\times N}$.
2  Initialize the Circle map and generate eight chaotic values ( $c_1$ to $c_8$ ) using Eq.(1).
3  Initialize four Keys $K_1, K_2, K_3, K_4$ using Eq.(10).
4  Invoke the Decrypt function with $K_4$ for an encrypted image.
5  Invoke the Decrypt function with $K_3$ for a modified image.
6  Invoke the Encrypt function with $K_2$ for a modified image.
7  Invoke the Decrypt function with $K_1$ for a modified image.
```

4.5.2 Encrypt and decrypt functions

The encrypt and decrypt functions for the decryption process are the same as given in sections 4.4.2 and 4.4.3.

IV. EXPERIMENTS AND SECURITY ANALYSIS

An image encryption scheme is expected to resist the various attacks such as the brute-force attacks, statistical attacks, differential attacks and entropy based attacks [1-10]. This section analyzes the properties of the proposed encryption scheme to demonstrate its effectiveness in resisting these attacks.

5.1 Experimental setup

The proposed algorithm is implemented by using C programming language under the Linux platform by using a personal computer with an intel (R) Core(TM) i3-2120 CPU at 3.30 GHz with 2.91 GB of RAM. According to Eq.(11), the initial parameters of the quadruple encryption process is randomly set to $\{Z_0 = 0.8,\ p_1 = 0.5,\ p_2 = 0.6\ , p_3 = 0.7, a = 1.4, b = 0.3\}$. The test images are chosen from the *USC-SIPI* image database (sipi.usc.edu/database/) and are gray-scale images(256×256).

5.2 Visual assessment

The proposed encryption scheme has been tested with a number of test images of differing contents. Fig.1 shows the visual assessment of the original, encrypted and decrypted images of five different images. The encrypted images are non-recognizable in appearance, unintelligible, incomprehensible, disordered, random and noise-like images without any outflow of the original information. The decrypted images are exactly alike to the original images.

5.3 Key-space analysis

Brute-force attacks are based on the exhaustive search with each and every possible key. Hence, the encryption schemes are expected to provide large key-space. The proposed encryption scheme makes use of two chaotic maps. The key set of the Circle map is $\{Z_0, p_1,\ p_2\ ,\ p_3\}$ and the Henon map is $\{X_0, Y_0, a, b\}$. The proposed scheme uses all the four parameters of the Circle map and the last two values of Henon map as its key set, so the key set is $\{Z_0, p_1,\ p_2\ , p_3,\ a, b\}$. With 64 bits for each parameter and there are six real parameters, thus the key-length is 384 bits and the key-space is 2^{384}. Hence, the proposed algorithm has adequate key-space and is defiant to the brute-force attacks. Table 1 shows the key space size of the proposed scheme and the other schemes in the literature. It can be seen that the proposed scheme has competitive and adequate key space of 2^{384}.

Table 1. Key-space of the proposed scheme and some of the other methods in the literature

Method	Key space
Proposed algorithm	2^{384}
Ref.[1]	Not specified
Ref.[2]	2^{349}
Ref.[3]	2^{400}
Ref.[4]	2^{128}
Ref.[5]	2^{135}
Ref.[6]	2^{216}
Ref.[7]	10^{56}
Ref.[8]	2^{128}

5.4 Statistical tests

Statistical attacks are based on the assumptions about the distribution of the pixel intensity values and the correlations of the neighboring pixels. Statistical analysis can be performed by plotting the histograms and by computing the correlations as discussed below.

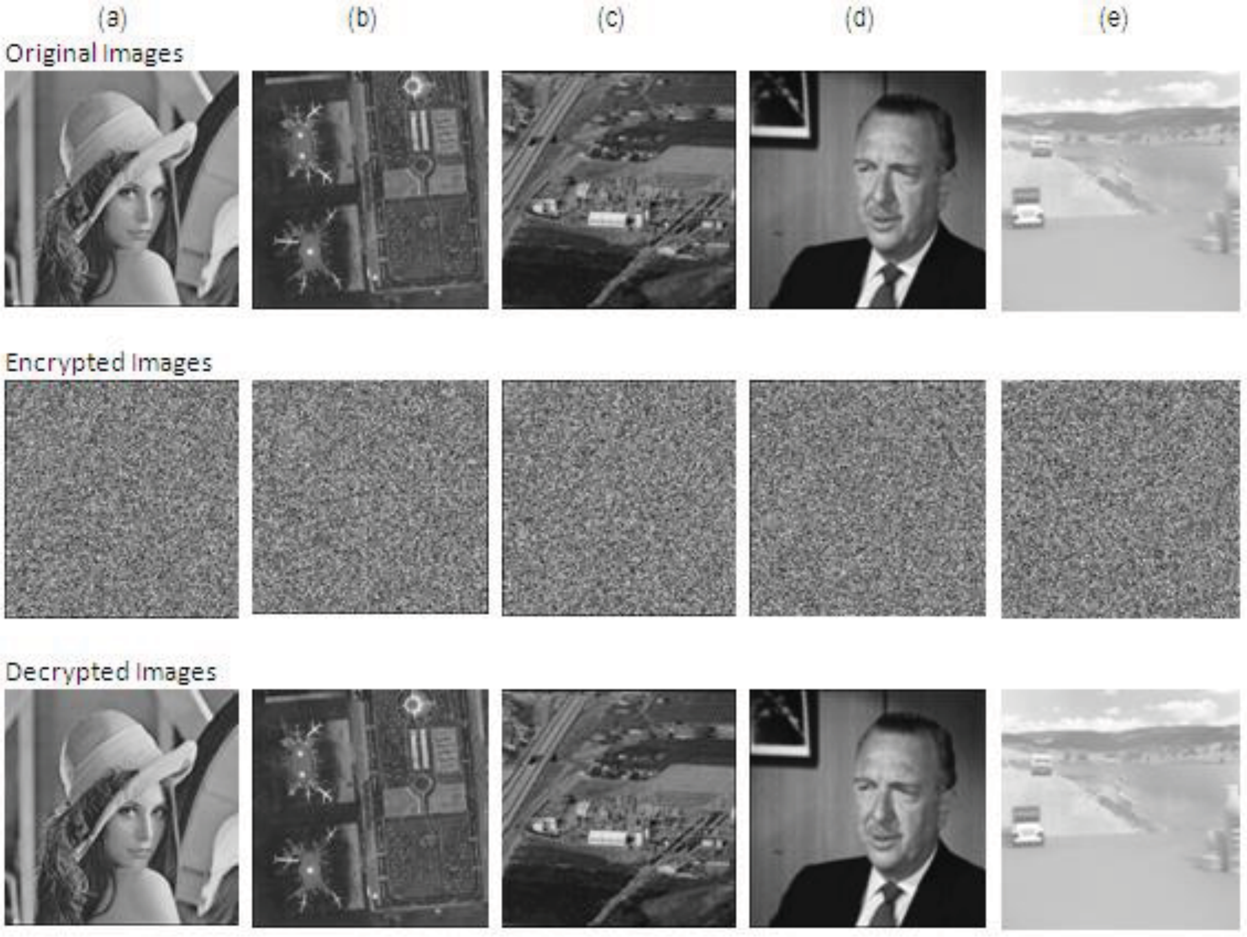

Fig. 1. Original images, encrypted images and decrypted images with the proposed algorithm (a) Lena (b) Airport (c) Chemical Plant (d) Walter Cronkite (e) Toy vehicle

5.4.1 Histogram analysis

An image histogram plots the frequency of each gray level. An encrypted image is anticipated to have no statistical similarity with the original image to prevent the outflow of the information. The histogram of the original image normally consists of huge spikes with some profiles. These spikes correspond to the gray values that occur more often in the image. The histogram of the encrypted image is expected to be adequately uniform to resist the statistical attacks. The histograms of a number of plain images are plotted and analyzed. The histograms for Airport and Walter Cronkite images are shown in Fig.2. The histograms of the encrypted images are uniformly spread and are entirely different from that of the original image, and bear no statistical resemblance to the original images. Hence, the proposed scheme is resistant to the histogram based statistical attacks. Majority of the schemes in the literature produce uniformly distributed histograms.

5.4.2 Correlation analysis

In most cases, for any original plain image having specific visual content, each pixel is highly correlated with its neighboring pixels in all the three directions: horizontal, vertical and diagonal. A good quality encryption scheme is anticipated to produce the encrypted images with no such correlations in the adjacent pixels. The correlation coefficient of the adjacent pixels is calculated according to Eq.(18-21).

$$E(x) = \frac{1}{N}\sum_{i=1}^{N} x_i \tag{18}$$

$$D(x) = \frac{1}{N}\sum_{i=1}^{N}(x_i - E(x))^2 \tag{19}$$

$$cov(x,y) = \frac{1}{N}\sum_{i=1}^{N}\left(x_i - E(x)\right)\left(y_i - E(y)\right) \tag{20}$$

$$r_{xy} = \frac{cov(x,y)}{\sqrt{D(x)}\sqrt{D(y)}} \tag{21}$$

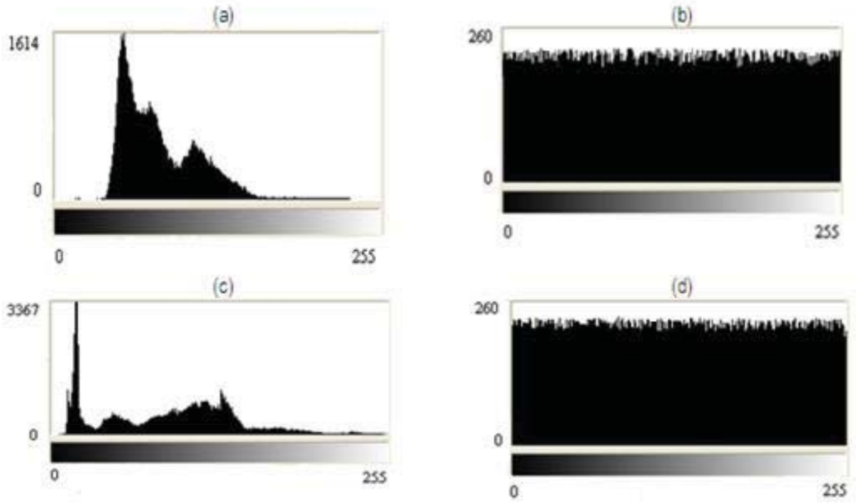

Fig. 2. Histograms of the original and encrypted images (a) Histogram of the Airport image (b) Histogram of encrypted image of the Airport image (c) Histogram of the Walter Cronkite image (d) Histogram of the encrypted image of Walter Cronkite

where x and y are the adjacent pixels of the original or encrypted images, $E(x)$ is the mean value, $D(x)$ is the deviation with regard to the mean, $cov(x,y)$ is the covariance between the adjacent pixels, and r_{xy} is the correlation coefficient. To assess the correlation in the original and encrypted images, 5096 pairs of the adjacent pixels are randomly selected in horizontal, vertical and diagonal directions, and their correlation coefficients are computed by using Eq.(21). The Table 2 lists the computed correlation coefficients of the original and encrypted images for different images. From Table 2, it is observed that the two adjacent pixels in the original image are greatly correlated to each other, whereas the correlation coefficients for the encrypted images are extremely close to zero. Hence, the proposed scheme is resistant to the correlation based statistical attacks. The comparison of the correlation results with the other scheme is listed in Table 3. It can be seen that correlation results of the proposed scheme is extremely close to zero and comparable to the other schemes in the literature.

5.5 Sensitivity tests

The differential attacks are based on how to differ in input can influence the resultant difference at the output. An image encryption scheme is anticipated to be sensitive to both the keys and the plain image and these are discussed below.

5.5.1 Key sensitivity analysis

Key sensitivity implies that a minute change in the secret key should generate entirely different encrypted image. The key sensitivity test is conducted with the following approach.

1 The original image is encrypted by using a test key K_1 to produce cipher image C_1.

2 The original image is encrypted again with a very small change in the test key K_1, i.e. K_2 to produce cipher image C_2.

3 The two cipher images C_1 and C_2 with little different keys are compared with pixel by pixel to observe the number of differing pixels.

The $NPCR$ and $UACI$ measures are used to evaluate the key sensitivity and are discussed hereafter.

$NPCR$ (Number of Pixels Change Rate) is used to compute the percentage of the total number of different pixels in two images and is calculated by using Eq.(22).

$$NPCR = \frac{\sum_{i,j} D(i,j)}{M \times N} \times 100\% \qquad (22)$$

$$D(i,j) = \begin{cases} 1, & if \ C_1(i,j) \neq C_2(i,j) \\ 0 & , Otherwise \end{cases} \qquad (23)$$

where C_1 and C_2 are the two encrypted images with slightly different keys K_1 and K_2. $C_1(i,j)$ and $C_2(i,j)$ are the pixel values of C_1 and C_2 at position (i,j). D is a bipolar array with the same size as C_1 and C_2 and its contents are either 0 or 1 based on Eq.(23).

$UACI$ (Unified Average Changing Intensity) is used to compute the percentage of the average changing intensity difference between the two encrypted images and is computed as,

$$UACI = \frac{1}{M \times N}\left[\sum_{ij} \frac{C_1(i,j) - C_2(i,j)}{255}\right] \times 100\% \qquad (24)$$

Table 2. Correlation coefficients of the adjacent pixels in different directions for the original and encrypted images

Image	Correlation coefficients for encrypted images			Correlation coefficients for original images		
	Horizontal	Vertical	Diagonal	Horizontal	Vertical	Diagonal
Lena	0.005794	0.005644	0.003480	0.968683	0.943269	0.933408
Airport	0.003446	-0.007657	0.002155	0.850811	0.800954	0.729673
Chemical plant	0.002155	-0.002842	0.000282	0.916546	0.947592	0.896024
Walter Cronkite	-0.000058	0.002272	0.001407	0.987105	0.987038	0.978499
Toy Vehicle	0.000192	-0.013196	-0.002466	0.943520	0.982781	0.933587

Table 3. Comparison of the correlation coefficients of the proposed scheme with the other methods

Method	Correlation coefficients		
	Horizontal	Vertical	Diagonal
Proposed algorithm	0.005794	0.005644	0.003480
Ref.[1]	0.077000	-0.072360	-0.061530
Ref.[2]	0.003200	0.002424	0.001520
Ref.[3]	0.006300	0.005900	0.007300
Ref.[4]	-0.001700	0.005900	-0.001900
Ref.[5]	-0.044500	0.004100	Not specified
Ref.[6]	0.004800	0.002700	0.003600
Ref.[7]	-0.097360	-0.070680	0.048440
Ref.[8]	0.011830	0.001600	0.014800

The key sensitivity is assessed by testing one parameter at a time with a very small change in the key. The proposed scheme has six parameters $\{Z_0, p_1, p_2, p_3, a, b\}$. Table 4 shows the $NPCR$ and $UACI$ values for six different parameters. From Table 4, it is observed that the $NPCR$ and $UACI$ values are close to their ideal values of 99.6% and 33.4%. Hence the proposed approach has high key sensitivity. The key sensitivity results in the literature are satisfactory and are demonstrated in the visual form hence are not shown here.

Table 4. Key sensitivity results with different parameters of the chaotic map

Parameter changed	$NPCR$ (%)	$UACI$ (%)
Z_0	99.598694	33.533218
p_1	99.598694	33.533218
p_2	99.598694	33.533218
p_3	99.598694	33.533218
a	99.630737	33.416212
b	99.617004	33.350189

Furthermore, the key sensitivity is also assessed pictorially with the following approach. The original key is altered with a minute change and a different key is generated. The keys can be described as, original key $\boldsymbol{Key_1}$= (0.8, 0.5, 0.6, 0.7, 1.4, 0.3), and the slightly modified key $\boldsymbol{Key_2}$= (**0.80000000001**, 0.5, 0.6, 0.7, 1.4, 0.3). The two encrypted images (C_1 and C_2) with slightly different keys (Key_1 and Key_2) are shown in Fig.3b-c respectively. Although both look alike, they are totally different from each other. This can be confirmed by finding the difference image between C_1 and C_2. Fig.3d shows the difference image between C_1 and C_2. From Fig.3d it is observed that the majority of the pixels in the difference image are nonzero.

In addition, the key sensitivity test is also analyzed for the decryption process. The decryption is performed with the correct key and the slightly different keys. The Fig.4a shows the decrypted image with correct key $\boldsymbol{Key_1}$= (0.8, 0.5, 0.6, 0.7, 1.4, 0.3) and Fig.4b-c are the decrypted images with slightly altered keys $\boldsymbol{Key_2}$= (**0.80000000001**, 0.5, 0.6, 0.7, 1.4, 0.3) and $\boldsymbol{Key_3}$= (0.8, **0.50000000001**, 0.6, 0.7, 1.4, 0.3). Hence, the correct decryption cannot be attained if there is a little change in the key.

5.5.2 Plain-image sensitivity analysis

Plain-image sensitivity means, a little change in the original plain image should cause a considerable change in the encrypted image. The plain-image sensitivity assessment is conducted with the following steps.

1 Encrypt the original plain-image to produce a cipher image C_1.

2 Change one bit of the original plain-image at any random location, and encrypt again to produce a cipher image C_2.

3 The two cipher images C_1 and C_2 are compared with pixel by pixel to observe the number of differing pixels.

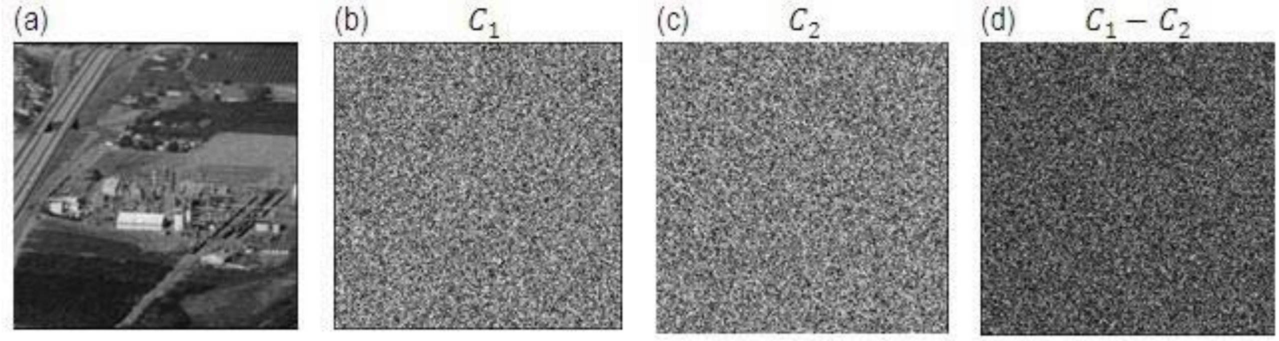

Fig. 3. Key sensitivity analysis for the encryption process for Chemical plant image (a) original image (b) encrypted image with the correct key $\boldsymbol{Key_1}$ (c) encrypted image with slightly different key $\boldsymbol{Key_2}$ (d) difference image between $\boldsymbol{C_1}$ and $\boldsymbol{C_2}$

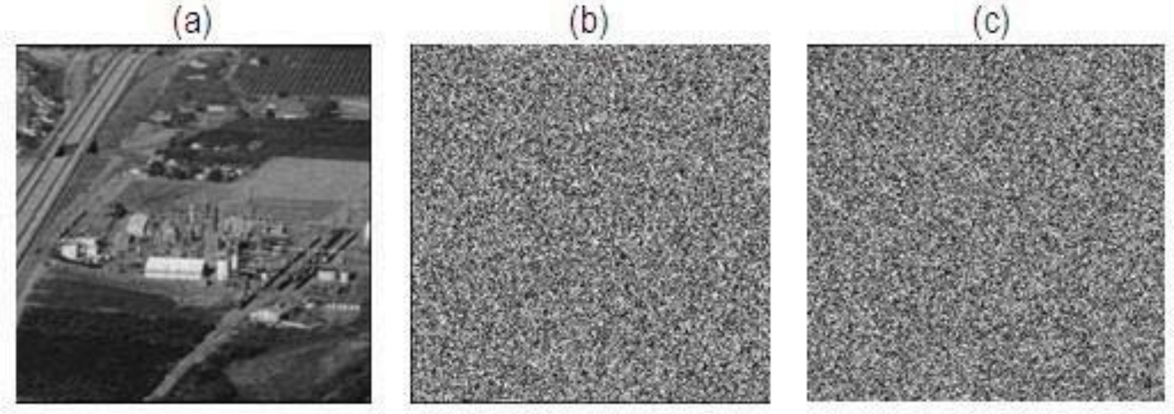

Fig. 4. Key sensitivity analysis for the decryption process for Chemical plant image. (a) decryption with the correct key (b-c) decryption with slightly changed keys

Two measures $NPCR$ and $UACI$ are used to assess the plain-image sensitivity as given in Eq.(22-24). The $NPCR$ and $UACI$ values are calculated for different randomly chosen positions by altering one bit at a time and the results are listed in Table 5. From Table 5, it is observed that the $NPCR$ and $UACI$ values are close to their ideal values of 99.6% and 33.4% irrespective of the pixel position. Thus the proposed approach has high sensitivity to plain-images and is defiant to the differential attacks. Table 6 lists the plain image sensitivity results of the proposed scheme and the other schemes in the literature. The plain image sensitivity results of the proposed scheme are comparable to other schemes and are close to the ideal values.

Table 5. Plain-image sensitivity test at different positions of the image

Position	$NPCR$ (%)	$UACI$ (%)
(0,0)	99.600220	33.382053
(40,70)	99.630737	33.416492
(80,35)	99.601746	33.419903
(128,128)	99.610901	33.490643
(190,220)	99.606323	33.452763
(255,255)	99.607849	33.395039
Average values	99.609629	33.426148

Table 6. Comparison of the plain-image sensitivity results with other schemes in the literature

Method	Plain image sensitivity	
	NPCR %	*UACI %*
Proposed algorithm	99.609629	33.426148
Ref.[1]	99.576000	33.441000
Ref.[2]	99.610000	33.450000
Ref.[3]	99.609200	33.489100
Ref.[4]	99.610000	33.450000
Ref.[5]	60.00 – 94.00	15.00 – 38.00
Ref.[6]	99.562100	33.432400
Ref.[7]	99.233000	33.479000
Ref.[8]	50.200000	25.200000

5.6 Information entropy analysis

The information entropy is a measure of the amount of unpredictability in information content, and is defined as,

$$H(K) = \sum_{i=0}^{r-1} P(K_i) \, log_2 \frac{1}{P(K_i)} \qquad (25)$$

where K_i represents the pixel values, $P(K_i)$ is the probability of the symbol K_i and r is the number of symbols and is 256 for gray level image. Suppose the gray level image has 2^8 gray values with equal probabilities, $K = (K_0, K_1, K_2, \dots, K_{255})$, according to Eq.(25), we obtain its entropy value $H(K) = 8$. Generally, due to high correlation, the original plain images are seldom random, hence the entropy value is usually smaller than the ideal value of 8. The entropy reaches the maximum ideal value of 8 when all the pixels are randomly scattered. Table 7 shows the entropy values for the original plain images and the encrypted images. From the results, it is observed that the entropies of the encrypted images are extremely close to the ideal value of 8. The information outflow in the proposed encryption scheme is insignificant and is secure against the entropy based attacks. The comparison of the entropy outcome with other schemes is listed in Table 8. The entropy outcome of the proposed scheme is close to the ideal value of 8 and is comparable to other schemes in the literature.

Table 7. Entropy values for the original and encrypted images for different images

Image	Entropy	
	Original image	Encrypted image
Lena	7.426985	7.997118
Airport	6.690835	7.997248
Chemical plant	7.169468	7.996899
Walter Cronkite	7.206486	7.997322
Toy Vehicle	6.141587	7.997055

Table 8. Comparison of the entropy values of the proposed scheme with the other methods

Table 8. Comparison of the entropy values of the proposed scheme with the other methods

Method	Entropy
Proposed algorithm	7.997118
Ref.[2]	7.996991
Ref.[3]	7.993000
Ref.[6]	7.994000
Ref.[1,4,5,7,8]	Not specified

5.7 Other tests

5.7.1 Peak signal to noise ratio (PSNR) analysis

By considering the original plain image as a signal and the encrypted image as noise, the objective evaluation of the encryption scheme can be performed by computing the $PSNR$ [5]. The $PSNR$ is computed as follows,

$$PSNR = 20 \times log_{10}\left(\frac{255}{\sqrt{MSE}}\right) dB \qquad (26)$$

where MSE is the mean square error and is calculated as follows,

$$MSE = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}(\,|I(i,j) - I'(i,j)|\,)^2 \qquad (27)$$

where $I(i,j)$ is the pixel intensity value of the original plain image and $I'(i,j)$ is the pixel intensity value of the encrypted image at location (i,j). The $PSNR$ values for different test images are computed and are shown in Table 9. From Table 9, it is observed that the $PSNR$ values are small, which indicate the complexity in getting the original plain image from the encrypted image for invaders. The comparison of the $PSNR$ outcome with other schemes is listed in Table 10. The entropy outcome of the proposed scheme is lower and is comparable to other schemes in the literature.

Table 9. The $\boldsymbol{PSNR}$ values for different test images

Image	$PSNR$(dB)
Lena	9.244208
Airport	8.871231
Chemical plant	8.729025
Walter Cronkite	8.092808
Toy Vehicle	7.805952

Table 10. Comparison of the $PSNR$ values of the proposed scheme with the other methods

Table 10. Comparison of the PSNR values of the proposed scheme with the other methods

Method	$PSNR$(dB)
Proposed algorithm	7.805952 – 9.244208
Ref.[5]	8.2462 – 9.7322
Ref.[1,2,3,4,6,7,8]	Not specified

5.7.2 Gray value degree（GVD）analysis

The gray difference of a pixel with its four neighborhood pixels in an image is defined as [13],

$$G = \frac{\sum\,[I(m,n) - I(m',n')]^2}{4}\,,\ \ here\ (m',n') = \begin{cases}(m-1,n)\\(m+1,n)\\(m,n-1)\\(m,n+1)\end{cases} \qquad (28)$$

where $I(m,n)$ represents the pixel intensity value at location(m,n), and $I(m',n')$ denotes the pixel intensity values of the four neighborhood pixels. The average neighborhood gray difference for the whole image can be computed as,

$$W(\,G(m,n)\,) = \frac{\sum_{m=2}^{M-1}\sum_{n=2}^{N-1}G(m,n)}{(M-2)\times(N-2)} \qquad (29)$$

where M and N are the height and width of the image. By using Eq.(28,29) the gray value degree is defined as,

$$GVD = \frac{W'\big(G(m,n)\big) - W\big(G(m,n)\big)}{W'\big(G(m,n)\big) + W\big(G(m,n)\big)} \qquad (30)$$

where W' and W denote the average neighborhood gray difference of the original plain image and the encrypted image. Table 11 lists the computed gray value degree values for different images by the proposed approach. From Table 11, it is observed that the gray value degrees computed by the proposed method are close to the ideal value of 1. Table 12 shows the comparison of the GVD value with other methods.

Table 11. Gray value degree values for different test images**.**

Image	GVD value
Lena	0.962085
Airport	0.936894
Chemical plant	0.961973
Walter Cronkite	0.986525
Toy Vehicle	0.994267

Table 12. The $\boldsymbol{GVD}$ values of the proposed scheme and other approaches

Method	GVD value
Proposed algorithm	0.962112
Ref.[13]	0.954000
Arnold's	0.890000
Ref.[1,2,3,4,5,6,7,8]	Not specified

5.8 Computational speed analysis

The running time of an encryption scheme can be determined by several factors such as the programming language, operating system, system configuration etc. Table 13 shows the computational time requirement of the proposed scheme and other schemes in the literature. The proposed scheme requires only 0.000009 seconds, and is fast compared to [3,6] with similar computer speed. The proposed scheme is also faster compared to [2,4] with similar programming language. The comparison of the time complexities is shown in Table 14. It can be seen that the time complexity of the proposed scheme is lower and is fast compared to other schemes.

VI. CONCLUSIONS

In this paper, a new approach for the image encryption based on Quadruple encryption with dual chaotic maps is proposed. The proposed approach has a key space of 2^{384} which is large enough to prevent the brute-force attacks. Moreover, the correlations are close to zero and the histogram is almost uniformly spread, so the statistical attacks are resisted. Furthermore, the $NPCR$ and $UACI$ values are close to their ideal values of 99.6% and 33.4% for both the key sensitivity and the plain image sensitivity tests, hence the scheme is defiant to the differential attacks. The average entropy attained is 7.997128, which is close to the ideal value of 8, and hence the information outflow is insignificant. The GVD is near to 1 and the $PSNR$ is lower. The proposed scheme is implemented under the Linux platform by using C language, and the speed attained is 0.000009 seconds for a 256×256 image. Accordingly, the image encryption scheme proposed in this paper will render excellent image encryption effects and high speed. The proposed scheme can also be applied for color images.

REFERENCES

[1] Guodong Ye and Kwok-Wo Wong, "An efficient chaotic image encryption algorithm based on a generalized Arnold map," *Nonlinear Dynamics*, vol. 69, no. 4, pp. 2079-2087, Sep. 2012

[2] Ahmed A. Abd El-Latif, Li Li, Tiejun Zhang, Ning Wang, Xianhua Song and Xiamu Niu, "Digital image encryption scheme based on multiple chaotic systems," *Sensing and Imaging,* vol. 13, no. 2, pp. 67-88, Jun. 2012

[3] Shatheesh Sam, P. Devaraj and Raghuval S. Bhuvaneswaran, "A novel image cipher based on a mixed transformed logistic maps," *Multimedia Tools and Applications*, vol. 56, no. 2, pp. 315-330, Jan. 2012

[4] M. Francois, T.Grosges, D.Barchiesi and R.Erra, "A new image encryption scheme based on a chaotic function," *Signal Processing: Image Communications,* vol. 27, no. 3, pp. 249-259, Mar. 2012

[5] Nidhi Taneja, Balasubramanian Raman and Indira Gupta, "Combinational domain encryption for still visual data," *Multimedia Tools and Applications*, vol. 159, no. 3, pp. 775-793, Aug. 2012

[6] Shantheesh Sam, P. Devaraj and R.S. Bhuvaneswaran, "An intertwining chaotic maps based image encryption scheme," *Nonlinear Dynamics*, vol. 69, no. 4, pp. 1995-2007, Sep. 2012

[7] Xiaoling Huang, "Image encryption algorithm using chaotic chebyshev generator," *Nonlinear Dynamics*, vol. 67, no. 4, pp. 2411-2417, Mar. 2011

[8] Xiaofeng Liao, Shiyue Lai and Qing Zhou, "A novel image encryption algorithm based on self-adaptive wave transmission," *Signal processing*, vol. 90, no. 9, pp. 2714-2722, Sep. 2010

[9] Ahmed A. Abd El-Latif, Li Li and Xiamu Niu, "A new image encryption scheme based on cyclic elliptic curve and chaotic system," *Multimedia Tools and Applications*, vol. 70, no. 3, pp. 1559-1584, Jun. 2014

[10] C.K.Huang, C.W.Liao, S.L.Hsu and Y.C.Jeng,

"Implementation of gray image encryption with pixel shuffling and gray-level encryption by single chaotic system," *Telecommunication Systems*, vol. 52, no. 2, pp. 563-571, Feb. 2013

[11] Ji won Yoon and Hyoungshick Kim, "An image encryption scheme with a pseudorandom permutation based on chaotic maps," Communications in Nonlinear Science and Numerical Simulations, vol. 15, no. 12, pp. 3998-4006, Dec. 2010

[12] A. Akhshani, S. Behnia, A. Akhavan, H. Abu Hassan and Z. Hassan, "A novel scheme for image encryption based on 2D piecewise chaotic maps," Optics Communications, vol. 283, no.17, pp. 3259-3266, Sep. 2010

[13] Guodong Ye, "Image scrambling encryption algorithm of pixel bit based on chao map," Pattern Recognition Letters, vol. 31, no. 5, pp. 347-354, Apr. 2010

[14] Gonzalo Alvarez and Shujun Li, "Some basic cryptographic requirements for chaos-based cryptosystems," International Journal of Bifurcation and Chaos, vol. 16, no. 8, pp. 2129-2151, Aug. 2006

[15] V. Patidar, N.K. Pareek and K.K. Sud, "A new substitution diffusion based image cipher using chaotic standard and logistic maps," Communications in Nonlinear Science and Numerical Simulations. vol. 14, no. 7, pp. 3056-3075, Jul. 2009

[16] N.K. Pareek, V. Patidar and K.K. Sud, "Image encryption using chaotic logistic map," Image Vision and computing, vol. 24, no. 9, pp. 926-934, Sep. 2006

[17] Bruce schneier, *Applied cryptography*. Singapore: John Wiley & sons, 2001

AUTHORS

Gururaj Hanchinamani received ME degree in computer science and engineering from Walchand college of engineering Sangli, India. He is pursuing PhD degree at Visvesvaraiah technological university Belgaum. His research interests are information security and computer architectures. He is currently working as associate professor at computer science department, BVB college of engineering and technology, Hubli, Karnataka, India.

Linganagouda Kulkarni received PhD degree in pattern recognition from Mysore university, India. His research interests are image processing, computer networks and information security. He is currently working as professor at computer science department, BVB college of engineering and technology, Hubli, Karnataka, India

This is blank Page