A Novel Approach for Image Encryption using Dynamic SCAN Pattern

T.Sivakumar and R.Venkatesan

Abstract — **In today's digital era, information is extensively stored and transmitted in the form of digital images, audio, and video files. Security is an important issue for storage and transmission of information due to the growth of electronic data exchange. Digital images are used in fields like telemedicine, military communication, and space research, so it is essential to protect the images from unauthorized disclosure during transmission. In this paper, a novel image encryption method is proposed using scan pattern, circular shift, and transposition methods. The pixels of the original image are permuted by the scan pattern generated by the notion of K^{th} smallest and stack dynamically. The circular shift and transposition are done by Shuffling Key (SK) generated from the original image. Value transformation is performed to confuse the relationship between the original and encrypted images using bitwise XOR operation. The proposed method satisfies the performance analysis tests such as histogram, correlation, large key space, and acceptable encryption speed. Also, the proposed method is resistant to statistical, brute force, and differential attacks.**

Index Terms— **Information Security, Image Encryption, Dynamic SCAN Patterns, Pixel Permutation**

I. INTRODUCTION

SECURITY and ethical issues in information systems are important concerns for most organizations and today's Internet communication. New media technologies such as social networking and video sharing are used by individuals and organizations to share information [1]. The phenomenal growth of Internet in the recent years has brought privacy concerns to the fore front. Data privacy, data management, data security, and protection of systems from hackers are very important for any organization [2]-[3].

Cryptography is an art of achieving security by encrypting messages to make them non-readable at the sender side and decrypting the messages at the receiver end to obtain the original information. The conventional cryptosystems such as DES – Data Encryption Standard, AES - Advanced Encryption Standard, and IDEA - International Data Encryption Algorithm have been designed to protect textual data and may not be suitable for multimedia data. The use of conventional cryptosystems to encrypt images directly is not adequate for two reasons [4]: (a) the image size is much larger than the text and it needs more time to encrypt the images and (b) the decrypted text must be equal to the original text, but this is not required for images, that is, small distortion in the decrypted image is acceptable by human perception system.

Manuscript received July 20, 2013; revised February 19, 2014.
T.Sivakumar and R.Venkatesan are with the PSG College of Technology, Coimbatore, Tamilnadu-641 004, India. (phone: 0422-257217; fax: 0422-2573833; e-mail: sk@ity.psgtech.ac.in, ramanvenkatesan@yahoo.com).

A scanning of a two dimensional array is an order in which each element of the array is accessed exactly once. This is a formal language-based two dimensional spatial accessing methodology which could generate a large number of scanning paths [5]. An image could be encrypted in three ways namely, (a) bit permutation, (b) pixel permutation, and (c) block transformation. In this paper, a novel image encryption method using pixel permutation and transposition is proposed.

The rest of the paper is organized as follows. Section two reviews existing image cryptosystems and section three presents the proposed image encryption method with an example. Section four gives the experimental results and analysis. Section five and six deals with the real time implementation choices and conclusion of the paper.

II. RELATED WORK

Han Shuihua *et al*. [4] designed an asymmetric image encryption scheme based on matrix transformation. This scheme is based on matrix transformation, which is easily implemented and highly efficient to quickly encrypt and decrypt images. Maniccam *et al*. [5] revealed an image and video encryption method based on SCAN methodology, which is a formal language-based two-dimensional spatial accessing technique. The basic idea of this image encryption method is to rearrange the pixels of the image and change the pixel values. The pixel rearrangement is done by scan keys and the pixel values are changed by a simple substitution mechanism.

The intelligible information present in an image is due to the correlations among the bits, pixels and blocks in a given arrangement. This perceivable information could be reduced by decreasing the correlation among the bits, pixels and blocks using certain combinational permutation techniques. The permutation of pixels and blocks are good at producing higher level security compared to bit permutation [6]. In Hill cipher, if the key matrix is not invertible, then encrypted text could not be decrypted. Bibhudendra *et al*. [7] presented an Advanced Hill encryption technique, which uses an involutory matrix, to encrypt images. The complexity could be reduced by avoiding the process of finding inverse of the matrix during decryption.

Chin Chen *et al*. [8] described an image encryption algorithm based on Vector Quantization to encrypt many images with only one codebook. Guodong Ye [9] proposed an image encryption using double logistic maps, in which the digital image is confused from row and column respectively. Confusion and diffusion effects are carried out solely by the substitution and diffusion stages. Chen's system is employed to diffuse the gray value distribution. Haojiang Gao *et al*. [10] presented a Nonlinear Chaotic Algorithm (NCA) using power and tangent functions instead

of linear function. This encryption algorithm is a one-time-one-password system.

Kanso *et al*. [11] suggested an algorithm based on three dimensional chaotic maps. There are three rules to determine the shuffling, mixing and scrambling processes of the pixel values in the plain-image. The image pixels are shuffled according to a search rule based on the 3D chaotic map. Khaled Loukhaoukha *et al*. [12] brought an image encryption algorithm based on Rubik's cube principle. The original image is scrambled using the principle of Rubik's cube and then to confuse the relationship between original and encrypted images where XOR operation is applied to rows and columns of the scrambled image. Liu Hongjun *et al.* [13] designed a stream cipher based on one-time keys and robust chaotic maps. The algorithm uses piecewise linear chaotic map as the generator of a pseudo-random key sequence.

Mohammad Ali *et al.* [14] introduced an image encryption algorithm based on block-based transformation and the conventional Blowfish algorithm. The algorithm resulted in lowest correlation and the highest entropy when the transformation algorithm and Blowfish are combined together. It is also found that an inverse relationship exists between the number of blocks and correlation, and a direct relationship between the number of blocks and entropy. Sanfu Wang *et al*. [15] presented an image scrambling method based on a folding transform with folding matrix which is orthogonal thereby enabling to fold images either up-down or left-right. The algorithm has effective hiding ability for image information with matrix addition and subtraction. Sathishkumar *et al.* [16] developed an image encryption technique using multiple chaotic based circular mapping. The image is encrypted using logistic map, sub key and its transformation leads to diffusion process. Image encryption based on random pixel permutation and random shuffling strategies are used for permuting pixel coordinates [17]-[18].

Tzung Her Chen *et al*. [19] proposed a Random Grids-based Visual Secret Sharing scheme with the capability of encrypting multiple secret images at once into only two circular cipher-grids. To decrypt all secrets, decoders stack the two circular cipher-grids to disclose the first secret and then gradually rotate one circular cipher-grid at a fixed degree to disclose the second image. Wenping Guo [20] presented an image scrambling encryption algorithm based on chaotic sequence. The algorithm utilizes the good features of chaotic sequence related to cryptographic properties, such as pseudo-random and sensitivity to initial conditions. Xiaomin Wang *et al*. [21] presented an image scrambling method using Poker shuffle, which is controlled dynamically by chaotic system. The pixel positions are scrambled by shuffle sequences, which are generated by a chaotic-controlled Poker shuffle process.

Thus, image encryption based on chaotic sequence, scan pattern, permutation, pseudorandom numbers, and block transformation are applied by researchers in the literature. Nevertheless, it is imperative to make use of the concepts of k^{th} smallest and stack to encrypt images. Hence, in this paper, an attempt is made to encrypt digital images using the above said concepts.

III. THE PROPOSED SYSTEM

The input image(x) of size mxn is taken and the pixel positions are rearranged as follows. The 1^{st} smallest pixel element of each row is scanned and pushed onto the stack. After scanning the 1^{st} smallest pixel in the last row, the content of the stack is popped to form the first row of cipher matrix. Then the 2^{nd} smallest pixel of each row is scanned and pushed onto the stack. After scanning the 2^{nd} smallest pixel of the last row, the content of the stack is popped to form the second row of cipher matrix. This process is continued until scanning the n^{th} smallest pixel in all the rows. While scanning the image, the column order in which the first row is accessed is stored as shuffling key. This shuffling key is used for circular shift (for both row and column), permutation (for both row and column), and bitwise XOR operation. The encrypted image is transmitted along with the SCAN key and shuffling key. The encryption process of the proposed method is shown in Figure 1.

A. Pseudocode of Encryption Function

The encryption function performs tasks such as (a) encrypt the image (b) generate SCAN key for permutation, and (c) generate Shuffling Key (SK) to perform circular shift, transposition and XOR operations. The decryption is just the inverse of the encryption process.

Input: Original Image of size rowsize x colsize
Output: Cipher image of size rowsize x colsize,
Scan key and the Shuffling key

1. Let original_image[rowsize, colsize] be the original image for encryption, where rowsize, and colsize represent the number of rows and columns, respectively.
2. Initialize i, j, k, top equal to 1;

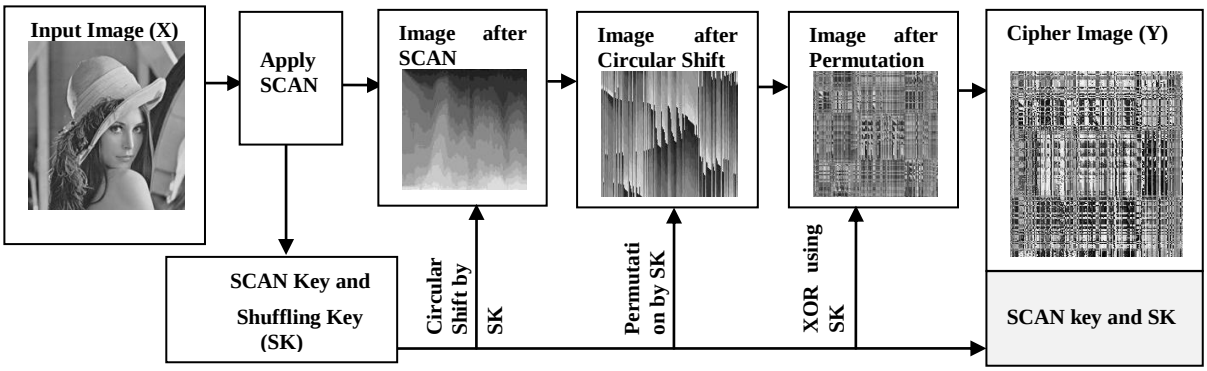


Fig. 1. Typical Encryption Process of Proposed System

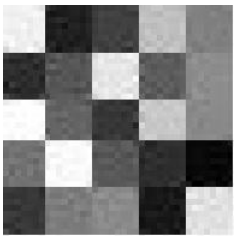
3. Repeat for i::1 to rowsize,
   (a) Repeat for j::1 to colsize,
      i. stack[top]←Min(original_image[i,:]), such that minimum pixel value for the i^{th} row can be pushed on to stack.
      ii. Increment top;
   (b) Repeat until the stack is empty
      i. i_image[:, k]←stack[top], such that an intermediate cipher image is created by arranging the minimum pixel values from each row.
      ii. scan_key[i, k]←index(stack[top]), such that the index of each minimum pixel value for the i^{th} row is popped and assigned as scan key for permutation.
      iii. Decrement top and increment k;
   (c) Increment i;
4. Assign shuffling_key←scan_key[1, i];
5. Repeat for i::1 to colsize
   R_Shift[i, :]←row_circularshift (original_image, shuffling_key[i]), such that each row is right shifted circularly.
6. Repeat for i::1 to rowsize
   Col_Shift[:,i]←col_circularshift(R_Shift, shuffling_key[i]), such that each column is shifted down circularly.
7. Repeat for i::1 to colsize // Row transposition
   Row_Trans[i, :]←row_trans (shuffling_key[i]),
8. Repeat for i::1 to rowsize // Column transposition
   Col_Trans[:, i]←col_trans(shuffling_key[i])
9. Repeat for i::1 to colsize
   Row_XOR[:,i]←bitxor(Col_Trans[i,:], shuffling_key[i]), such that each row elements are XORed with the shuffling_key.
10. Repeat for i::1 to rowsize
    Cipher_Matrix[i, :]←bitxor (Row_XOR [i,:], shuffling_key[i]), such that each column elements are XORed with the shuffling_key.
11. Return the Cipher_Matrix.

B. Illustration

In this section, the working model of the proposed system is illustrated using an image of size 5X5 pixels.

Step 1: Input the plain image (x).

Let us consider the following gray scale image shown in Figure 2(a). The intensity value of each pixel is shown in Figure 2(b).

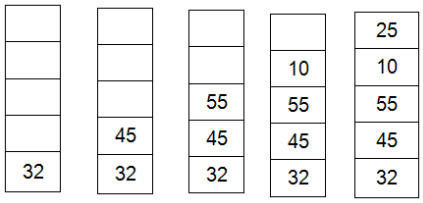


240	32	65	192	128
45	86	231	93	141
255	100	55	200	137
115	250	89	46	10
56	125	139	25	225

Fig. 2. (a) Original Image (b) Gray Value of Original Image

Step 2: Read the 1^{st}smallest pixel value from each row and push into the stack. Once all the rows are scanned for the smallest element, the elements are popped from stack to form the first row of the cipher matrix. The output of this process is shown in Figures 3(a) and (b).

The first smallest value in row one is at index two and it is stored as the first shuffling key. Figures 3(c) and (d) show the content of Shuffling Key (SK) array and SCAN key array after the first iteration.



(a) Content of Stack

25	10	55	45	32

(b) First Row of the Cipher Matrix

2				

(c) Shuffling Key

1, 2	2, 1	3, 3	4, 5	5, 4

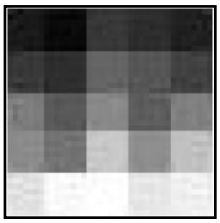
(d) SCAN Key

Fig. 3. (a) – (d) Output after First Iteration

The above step is repeated for 2^{nd}, 3^{rd}, 4^{th} and 5^{th}smallest in each row. The content of cipher matrix, the corresponding image and the value of the shuffling key array are shown in Figure 4(a), (b) and (c), respectively.

25	10	55	45	32
56	46	100	86	65
125	89	137	93	128
139	115	200	141	192
225	250	255	231	240

(a) Matrix after SCAN



(b) Image after SCAN

2	3	5	4	1

(c) Shuffling Key

Fig. 4. (a) – (c) Output after SCAN

Figure 5(a) shows the content of SCAN key array after scanning all elements of the original matrix and Figure 5(b) is the reduced SCAN key matrix after removing the column numbers. Since, the first column of the reduced scan key array is same as the shuffling key it is enough to send the reduced 5x5 matrix as SCAN key and Shuffling Key (SK) for decryption.

1, 2	2, 1	3, 3	4, 5	5, 4
1, 3	2, 2	3, 2	4, 4	5, 1
1, 5	2, 4	3, 5	4, 3	5, 2
1, 4	2, 5	3, 4	4, 1	5, 3
1, 1	2, 3	3, 1	4, 2	5, 5

Fig. 5. (a) SCAN Key

2	1	3	5	4
3	2	2	4	1
5	4	5	3	2
4	5	4	1	3
1	3	1	2	5

(b) Reduced SCAN Key

Step 3: Apply circular shift (for both row and column wise) using the shuffling key generated in step (2).The row and column wise circular shift is performed as follows: The first row is shifted right by two pixel positions using the shuffling key 2, with the first element of the row circularly following the last. Likewise, the second row is shifted right by three pixel positions using the shuffling key 3 and this is

done for the 3rd, 4th, and 5th rows using the shuffling keys 5, 4, and 1 respectively. The same process is repeated for columns using the same shuffling key. That is, the first column is shifted two pixel positions beneath, with the top element of the column circularly following the last. The final result of step (3) is shown in Figures 6(a) and (b).

115	89	25	56	231
240	200	65	93	55
45	225	137	192	46
100	32	141	255	128
125	86	250	10	139

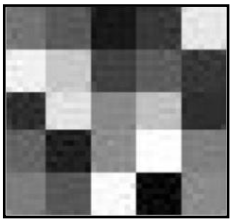
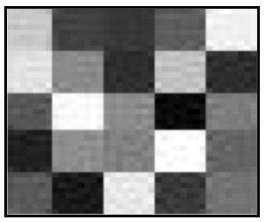


Fig. 6. (a) Cipher Matrix after Circular Shift (b) Image after Circular Shift

Step 4: Perform row and column permutation by using the shuffling key generated in step (2). The row and column wise permutations are performed by rearranging the columns first and then rows according to the order 2, 3, 5, 4 and 1. Figures 7(a) and (b) show the result after permuting rows and columns.

200	65	55	93	240
225	137	46	192	45
86	250	139	10	125
32	141	128	255	100
89	25	231	56	115

Fig. 7. (a) Cipher Matrix after Permutation

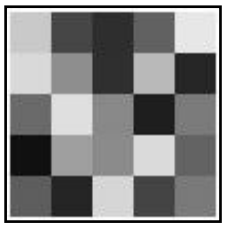


(b) Image after Permutation

Step 5: Perform row wise bitwise XOR operation.
Figures 8(a) and (b) show the output after performing bitwise XOR operation for all rows and columns using the shuffling key. Result shows that the encrypted matrix is completely different from the original pixel matrix shown in Figure 2(b).

200	64	48	91	243
224	137	40	199	47
81	252	139	11	121
38	138	128	255	97
90	27	227	61	115

Fig. 8. (a) Cipher Matrix

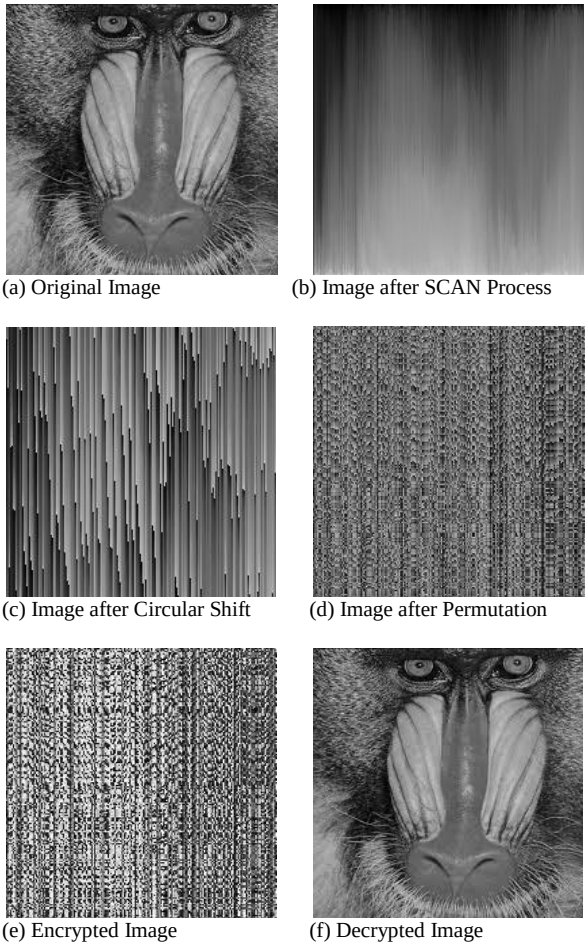


(b) Cipher Image

IV. EXPERIMENTAL RESULTS AND ANALYSIS

The proposed method is implemented in Matlab R2010a using a Personal Computer - Pentium-IV, 2.80 GHz, 1GB and the operating system used is Windows XP. The system is tested using standard gray scale images with 256x256 pixels.

Figures 9(a) - (f) illustrate the step by step result of proposed encryption on Baboon image. The original image is shown in Figure 9(a) and the resultant image after SCAN is shown in Figure 9(b). Figure 9(c) shows the output after applying circular shift for all rows and columns. Figure 9(d) shows the output after the SCAN process, circular shift, and permutation (SCSP). Figure 9(e) shows the encrypted image after SCAN, circular shift, permutation, and XOR operation (SCSPX). Figure 9(f) shows the decrypted image and it is exactly same as the original image; and hence the proposed method is lossless.



(a) Original Image (b) Image after SCAN Process
(c) Image after Circular Shift (d) Image after Permutation
(e) Encrypted Image (f) Decrypted Image

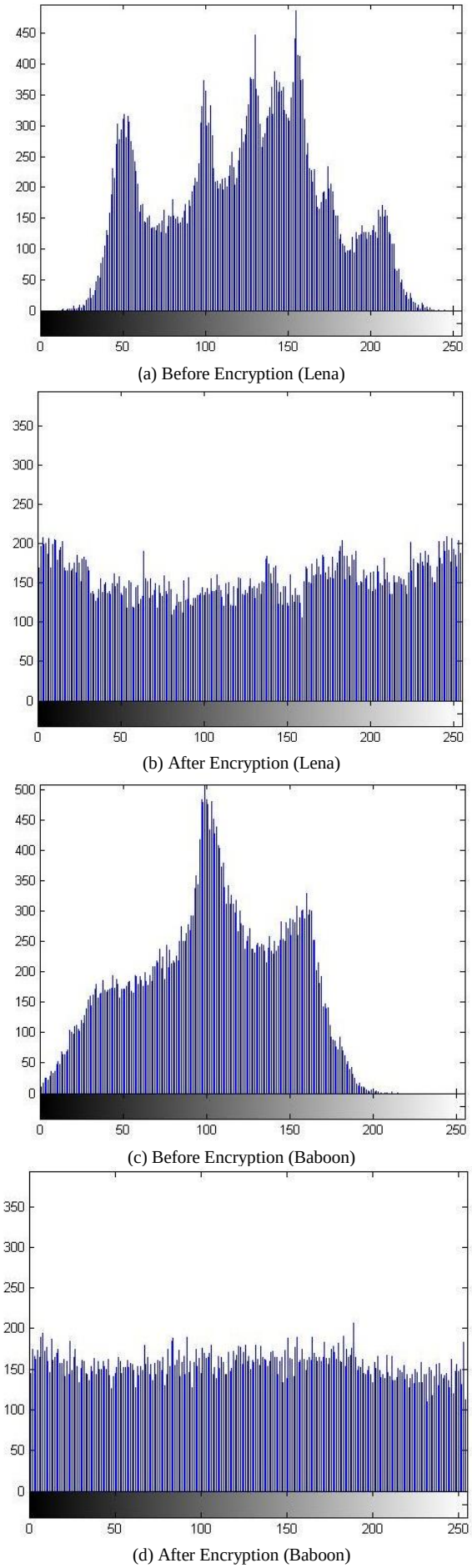
Fig. 9. (a) – (f) Step by Step Result of Proposed Method

The result shows that the encrypted image is different from the plain image and the correlation between the plain and encrypted images is close to zero. The correlation value between the original and encrypted images is 0.0029 for SCSP and -0.0139 for SCSPX.

A. Analysis of Histogram

Histogram is a graphical representation to show the visual impression of the distribution of data. An image histogram is a graphical representation of the pixel distribution in an image. It plots the number of pixels for each pixel value [22].

Figure 10 shows the histogram of the Lena and Map images before and after encryption. The histogram of the encrypted image shows that the gray scale values are uniformly distributed in the encrypted image. Hence the proposed method is safe from histogram analysis attacks and satisfies the diffusion property.



(a) Before Encryption (Lena)

(b) After Encryption (Lena)

(c) Before Encryption (Baboon)

(d) After Encryption (Baboon)

Fig. 10. (a) – (d) Histogram of Lena and Baboon Images

B. Analysis of Correlation

Correlation is a statistical analysis technique that could show whether and how strongly pairs of variables are related [22]. Correlation determines the relationship between the original image and cipher image. It is a measure that computes degree of similarity between the images and it is a useful measure to judge the encryption quality of cryptosystems. The image cryptosystem is said to be good, if encryption algorithm hides all attributes of the plaintext image, encrypted image is totally random and highly uncorrelated. This is calculated by the equation (1).

$$\gamma_{xy} = \frac{Cov(x,y)}{\sqrt{D(x)}\sqrt{D(y)}}, \qquad (1)$$

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

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

$$\mathrm{Cov}(x,y) = \frac{1}{N}\sum_{i=1}^{N}\big(xi - E(x)\big)\big(yi - E(y)\big)$$

Where, x_i and y_i are the gray scale values of corresponding pixels from the original and encrypted images, N is the number of pixel pairs (x_i, y_i), E(x) and D(x) are the mean and standard deviation of the gray level values. Figure 11 shows the correlation value between (a) original Vs. encrypted images using SCAN pattern; denoted as SCAN, (b) original Vs. encrypted images using SCAN and circular shift; indicated as SCS, (c) original Vs. encrypted images using SCAN, circular shift, and permutation; represented as SCSP, and (d) original Vs. encrypted images using SCSP and bitwise XOR operation; denoted as SCSPX.

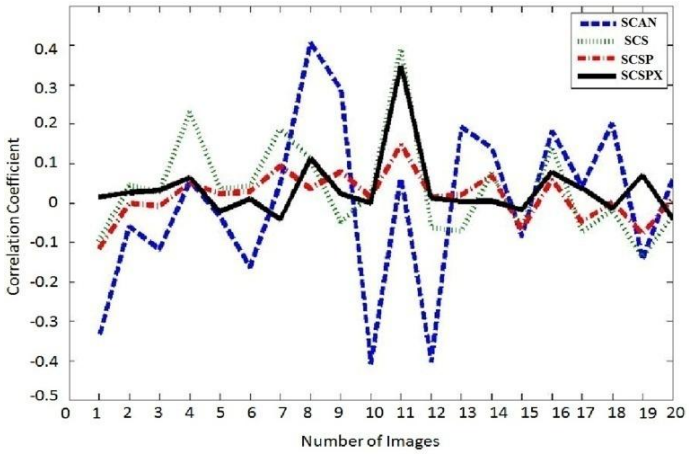
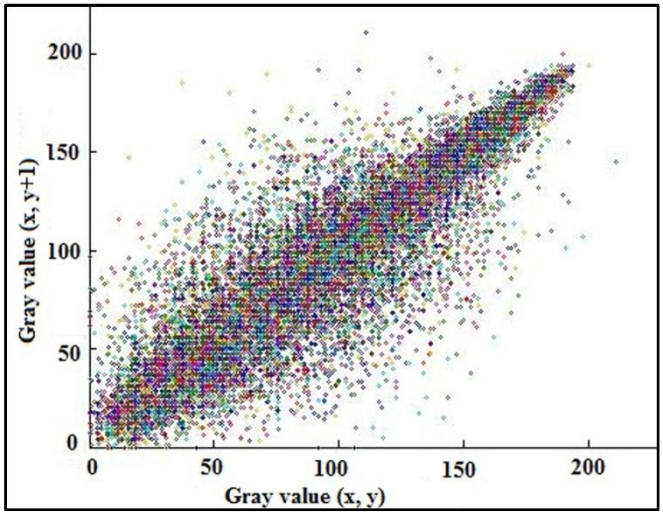
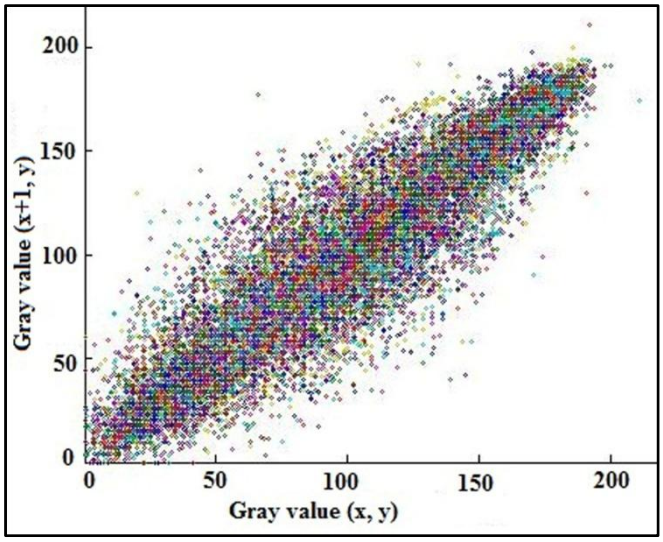


Fig. 11. Correlation between Original and Encrypted Images

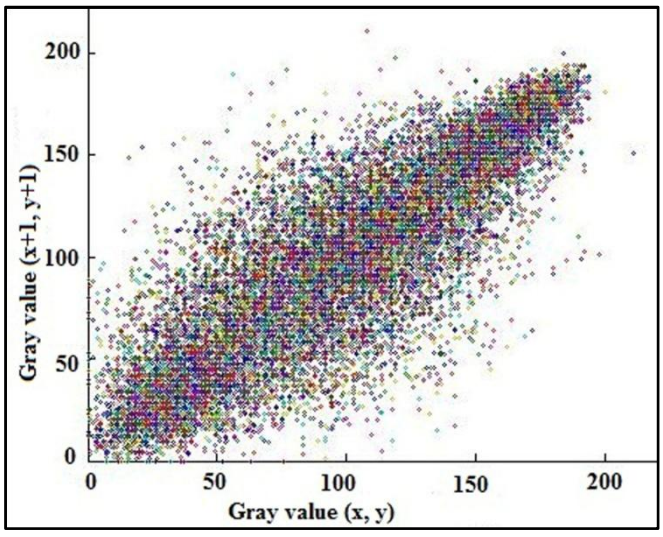
The correlation value shows that the statistical relationship between the original and encrypted images varies between 0.4 to -0.4. The correlation value is close to zero after the operations SCAN, circular shift, and permutation. Figure 12 (a) – (f) shows correlation distributions of the horizontal, vertical, and diagonal adjacent pixels of the original baboon image and the corresponding encrypted image.



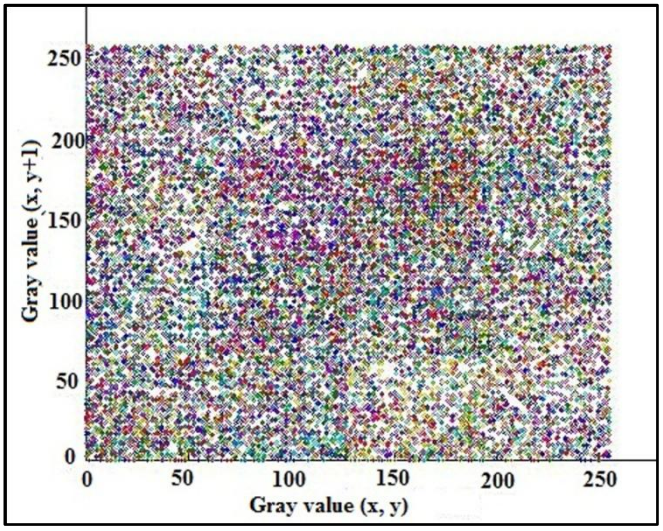
(a) Horizontal – Original Image



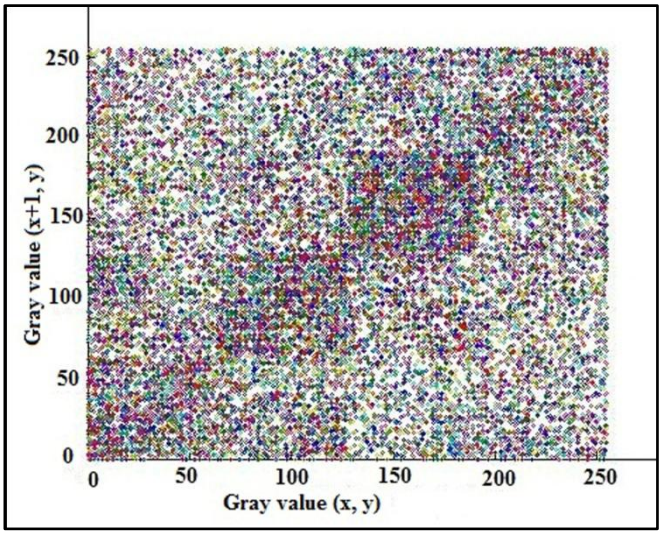
(b) Vertical – Original Image



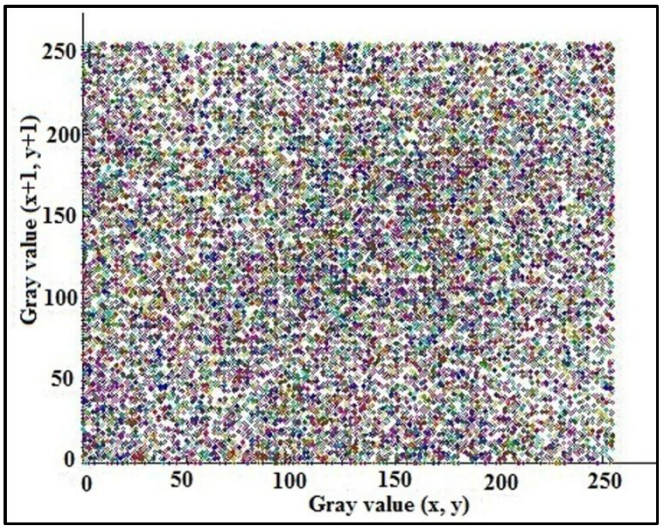
(c) Diagonal – Original Image



(d) Horizontal - Encrypted Image



(e) Vertical – Encrypted Image



(f) Diagonal - Encrypted Image

Fig. 12. Adjacent Pixels Correlation Distribution

Table II shows the corresponding correlation value and the correlation value is reduced considerably for the encrypted image. Hence, the proposed system satisfies confusion property significantly.

TABLE II
ADJACENT PIXEL CORRELATIONS

Image	Original			Encrypted		
	Hori.	Vert.	Diag.	Hori.	Vert.	Diag.
Lena	0.985	0.973	0.962	0.342	0.352	0.298
Baboon	0.975	0.983	0.972	0.293	0.395	0.207
Deer	0.969	0.958	0.948	0.249	0.386	0.174
Pepper	0.980	0.979	0.971	0.296	0.324	0.266

The results were taken after a single iteration of the proposed scheme. The correlation value after performing the proposed scheme on Baboon, Cameraman, and Planet images for the number of iterations varying in terms of five is shown in Figure 13.

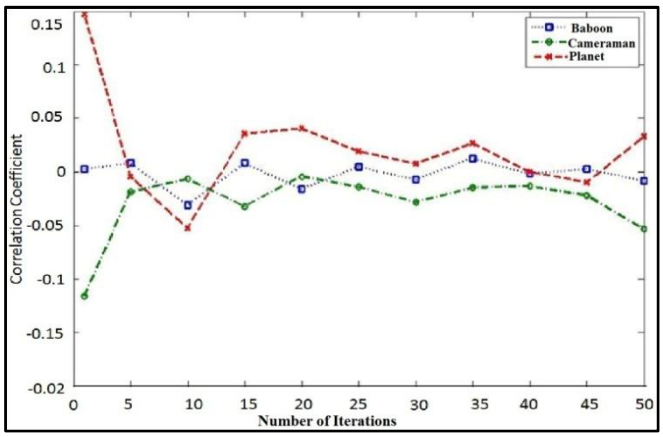
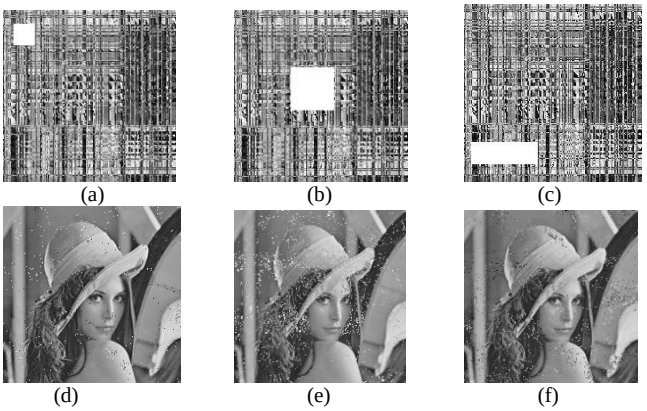


Fig. 13. Correlation Vs Number of Iterations

The correlation value is rapidly decreased for five iterations and for subsequent iteration, there is no drastic change. Thus, it is enough to repeat the SCSP process for five to ten iterations to reduce the processing overhead.

C. Cut Test Analysis

The prohibited users may cut or destroy the information condition so that they could not be decrypted. The cut test analysis is performed using Lena image. Figures 14(a) - (c) show the encrypted Lena images after removing 25x25, 50x50, and 75x25 pixel regions, respectively. Figures 14(d) - (f) show the corresponding decrypted images.



(a) (b) (c)

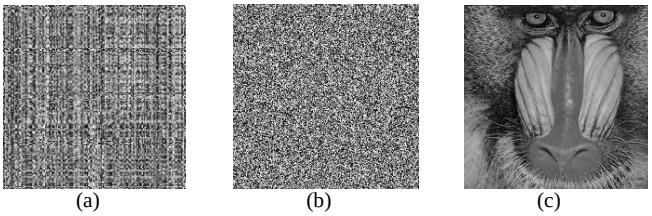
(d) (e) (f)

Fig. 14. Cut Test Analysis using Lena Image: (a) Encrypted Image (25x25), (b) Encrypted Image (50x50), (c) Encrypted Image (75x25), (d) Decrypted Image (25x25), (e) Decrypted Image (50x50), (f) Decrypted Image (75x25).

The decrypted images are negligibly distorted and could be recognized by human perception system as a Lena image. The proposed method is harmless from cropping attack and loss of data during transmission. It is also seen that the removed region is spread all over the image, and hence the proposed scheme has good dispersion rate.

D. Sensitivity of Shuffling Key

For successful decryption, the Shuffling Key (SK) is an important component. The proposed method is tested for the effect of small changes in the SK and the effect of change in SK value on Baboon image is shown in Figure 15.



(a) (b) (c)

Fig. 15. Testing the Sensitivity of Shuffling Key (a) Encrypted Baboon Image, (b) Decrypted Image using Random SK, (c) Decrypted Image using Original SK

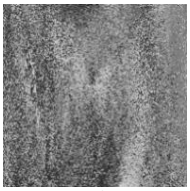
The result shows that the decrypted image is completely different from the original Baboon image, and hence the proposed method is very much sensitive with respect to the shuffling key value. i.e., a small change in SK will degrade and affect the quality of decrypted image.

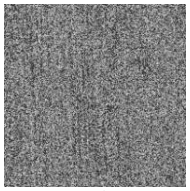
E. Key Space Analysis

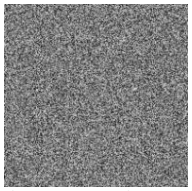
The elapsed time taken for encryption, including SCAN key generation, is approximately 12 sec and decryption is 0.62 sec. If the scan key in unknown, then there are n! combinations to place the pixels of the first row, where n is the number of columns. An image of size mxn pixels will have m*n! combinations to place all pixel elements of the image. Hence, the key space of an image of size 256x256 pixels is 10^{509}. If one decryption takes 0.5 sec, then an attacker need, approximately, $3.49e^{501}$ years for exhaustive key search attack. The time complexity of encryption() and decryption() functions are O(mxn).

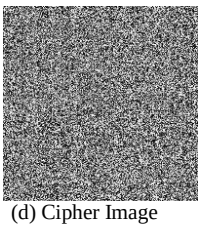
V. REAL TIME IMPLEMENTATION STRATEGIES

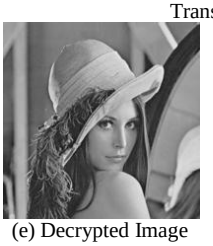
The SCAN and Shuffling Keys are generated from the plain image, and the proposed cryptosystem is a one-time-one-password scheme. A key image could be used to generate the scan pattern and shuffling key to encrypt and decrypt other images in real time implementation. The key image is shared securely before communication and this will eliminate overhead of transmitting the scan and shuffling keys for each image. If any suspicion or security breach is found in the network, then the key image could be altered. Figures 16(a) - (e) show the results of step by step encryption of Lena image using Baboon as a key image. Figure 16(f) shows the histogram of the encrypted Lena image and it is very much flat.

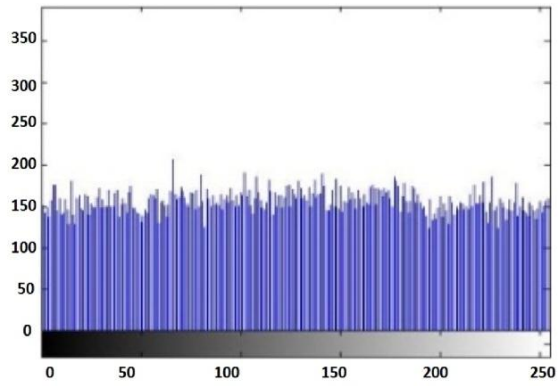

(a) Image after SCAN


(b) Image after Circular Shift

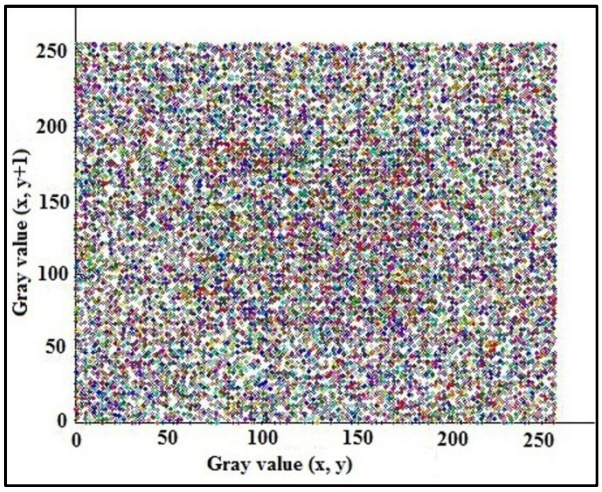

(c) Image after Transposition


(d) Cipher Image

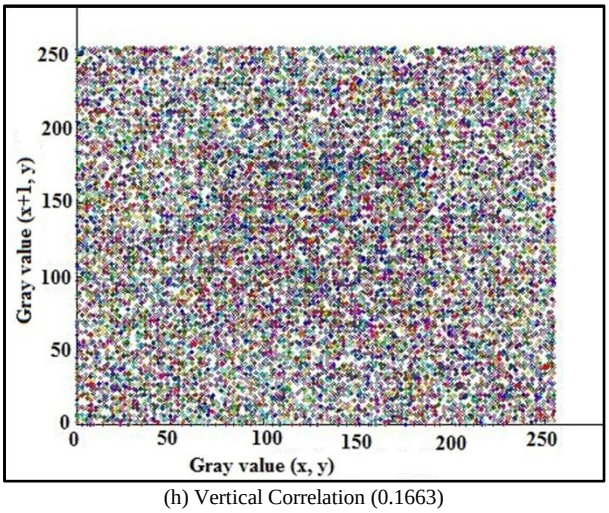

(e) Decrypted Image



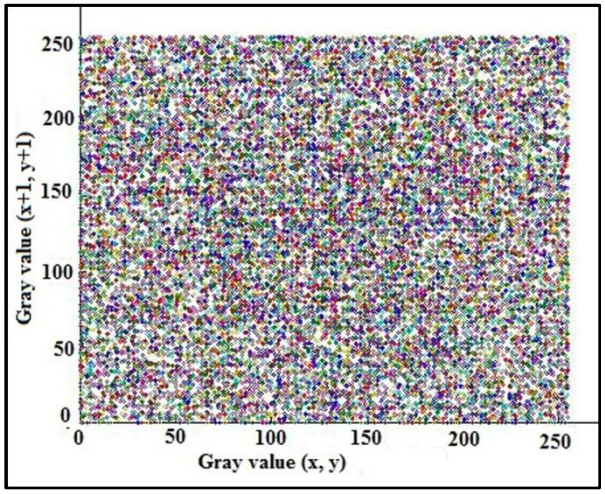
(f) Histogram of Encrypted Image



(g) Horizontal Correlation (0.1701)



(h) Vertical Correlation (0.1663)



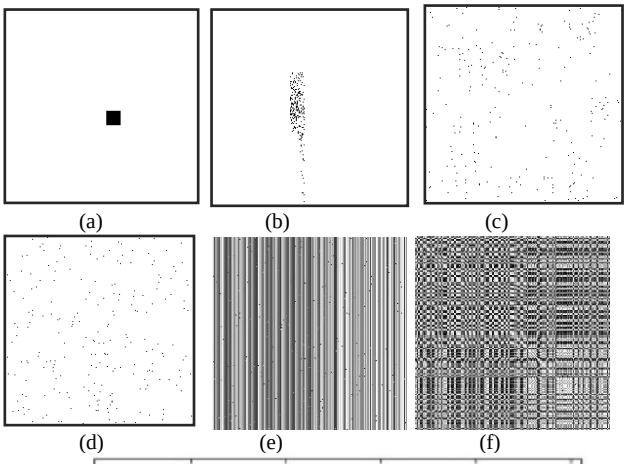
(i) Diagonal Correlation (0.1047)

Fig. 16. (a) – (i) Results of Encrypting Lena Image using Baboon as Key Image

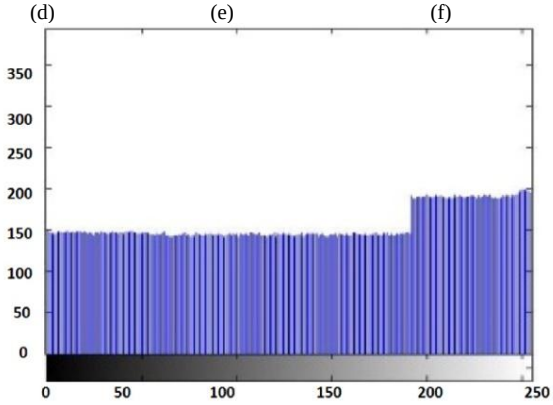
Figure 16(g) - (i) shows the correlation distribution between adjacent pixels of encrypted Lena image. The result shows that the proposed method satisfies the diffusion property and adjacent pixels in encrypted image are weakly correlated.

A. Results of Dispersion Test

Figures 17(a) - (f) illustrate the results of dispersion test using a white image with 15x15 black region. The results show that the 15x15 region is distributed across the entire image, and hence the proposed scheme has acceptable pixel permutation.


(a) (b) (c)

(d) (e) (f)



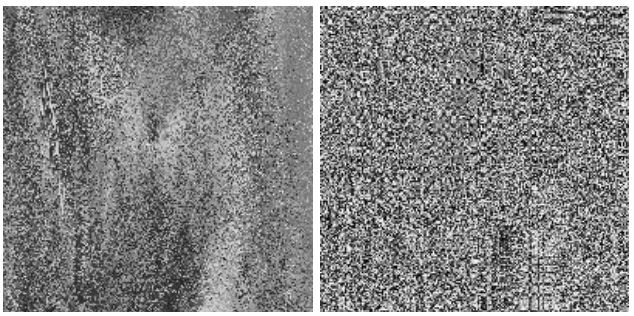
(g) Histogram of Encrypted Image

Fig. 17. Results of Dispersion Test: (a) Original Image, (b) Image after SCAN, (c) Image after Circular Shift, (d) Image after Transposition, (e) Image after Row XOR, (f) Cipher Image

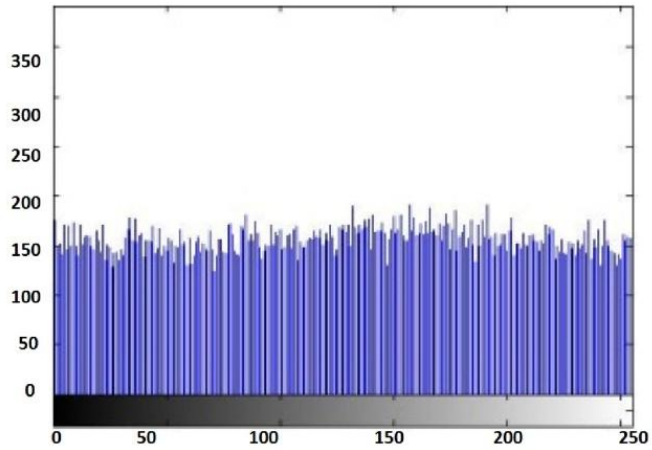
The proposed method could be implemented using any one of the following cases in real time applications. Case (1): applying SCAN pattern and XOR operation, case (2): applying SCAN pattern, row and column transposition, and XOR operation, and case (3): applying SCAN pattern, row and column circular shift, and XOR operation.

B. Results of Case 1

The result of case (1) using Baboon as key image and Lena as plain image is shown in Figure 18. Result shows that the encrypted image is different from the original image and thus, the proposed scan pattern provides acceptable pixel permutation. The expected time taken for encryption and decryption is 0.0382 and 0.0330 seconds, excluding SCAN key generation, respectively. The correlation between the original and encrypted images is 0.0042.



(a) Image after SCAN (b) Encrypted Image

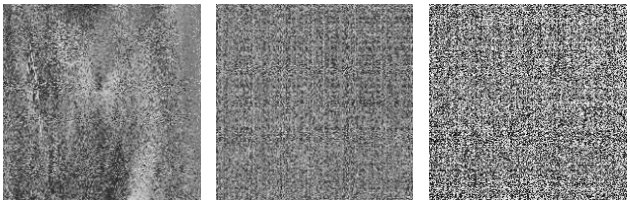


(c) Histogram of Encrypted Image

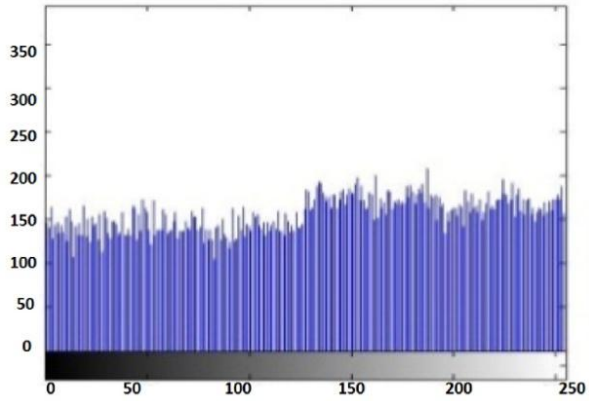
Fig. 18. (a) – (c) Results of Case (1)

C. Results of Case 2

The result of case (2) using Baboon as key image to encrypt Lena image is shown in Figure 19. Results show that the encrypted image in each step is entirely different from the original Lena image and hence, the combination of SCAN, transposition, and XOR offer reasonable hiding. The encryption and decryption time is 0.0140 and 0.0438 seconds, except SCAN key generation, respectively. The correlation between the original and encrypted images is -0.0013.



(a) Scanned Image (b) Image after Transposition (c) Encrypted Image

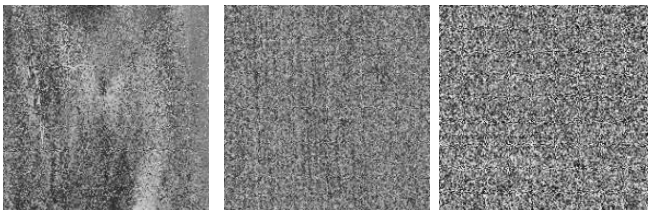


(d) Histogram of Encrypted Image

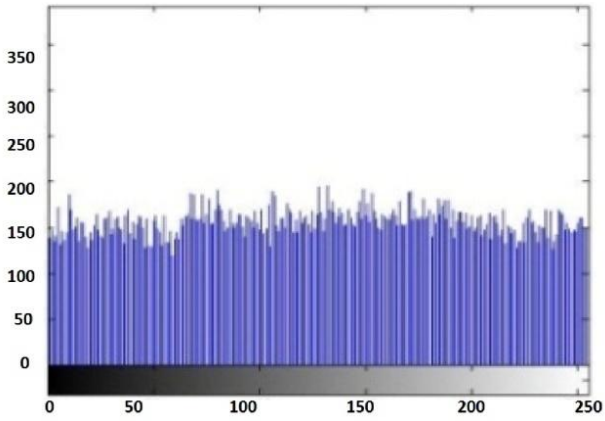
Fig. 19. Results of Case (2)

D. Results of Case 3

The results of case (3) using Baboon as key to encrypt Lena image is shown in Figure 20. The combination of proposed SCAN and circular shift could provide acceptable encryption result. The estimated time taken for encryption and decryption is 0.4532 and 0.5365 seconds, excluding SCAN key generation, respectively. The correlation between the original and encrypted images is -0.0089.



(a) Image after SCAN (b) Image Circular Shift (c) Encrypted Image



(d) Histogram of Encrypted Image

Fig. 20. Results of Case (3)

E. Adjacent Pixel Correlation

The horizontal, vertical, and diagonal correlation values of the encrypted images are given in Table III. The correlation value is reduced considerably after encryption and therefore, the proposed system satisfies confusion property significantly.

TABLE III

ADJACENT PIXEL CORRELATION (PROPOSED)

Case	Directions	Lena	Baboon	Cameraman
1	Horizontal	0.1610	0.1316	0.1394
	Vertical	0.1999	0.1618	0.1552
	Diagonal	0.0548	0.0211	0.0486
2	Horizontal	0.1915	0.1522	0.1923
	Vertical	0.1739	0.1445	0.2718
	Diagonal	0.0452	0.1100	0.1159
3	Horizontal	0.1686	0.1203	0.2449
	Vertical	0.1645	0.1627	0.2653
	Diagonal	0.0051	0.0269	0.1258

The horizontal, vertical, and diagonal correlations between adjacent pixels for a few existing image encryption algorithms are given in Table IV. The result shows that the proposed method correlation value is approximately close to the correlation value of existing algorithms mentioned in Table IV.

TABLE IV

ADJACENT PIXEL CORRELATION (EXISTING IMAGE ENCRYPTION METHODS)

Encryption Methods	Horizontal	Vertical	Diagonal
Ref [10]	-0.0159	-0.0654	-0.0323
Ref [12]	0.0068	0.0091	0.0063
Ref [14] Case 1	0.034	0.038	0.013
Ref [14] Case 2	0.0245	0.0067	0
Ref [14] Case 3	0.0129	0.0034	0.0014
Ref [23]	0.0263	0.0163	0.0114
Ref [24]	0.01776	0.04912	0.00348

F. Randomness Test

To resist the differential attacks, the cipher image should be sensitive to small changes in the plain image, and the cipher image must have random pixel values. The Number of Pixel Change Rate (NPCR) and the Unified Average Change in Intensity (UACI) are two parameters used to evaluate the sensitivity and randomness of an image. The algorithm is better, when the value of NPCR is close to 100 and UACI is just about 30. The NPCR and UACI are calculated by the equations (2) and (3), respectively.

$$\text{NPCR} = \frac{\sum_{i,j} D(i,j)}{W*H} * 100\% \quad (2)$$

$$\text{UACI} = \frac{1}{W*H}\left[\sum_{i,j} \frac{C1(i,j)-C2(i,j)}{255}\right] * 100\% \quad (3)$$

Where,

- C1 and C2 are the cipher images whose plain images have a slight difference.
- C1(i, j) denotes the pixel position in C1, and C2(i, j) denotes the pixel position in C2.
- D(i, j) is determined by C1(i, j) and C2(i, j), namely if C1(i, j) = C2(i, j) then D(i, j)=1; otherwise, D(i, j)=0.
- W and H are width and height of the image.

The resulting NPCR and UACI values are tabulated in Table V to Table VIII for both existing and proposed methods.

TABLE V

NPCR VALUES OF PROPOSED METHOD (in %)

Case	Lena	Baboon	Camera man	Coin
1	99.499	99.550	99.544	99.546
2	99.473	99.587	99.594	99.576
3	99.459	99.550	99.596	99.586

TABLE VI

NPCR VALUES OF EXISTING METHODS (in %)

Existing Methods	NPCR Value
Ref. [12]	99.5850
Ref. [23]	99.6185
Ref. [24]	99.85
Ref. [25]	99.72
Ref. [26]	> 99.42
Ref. [27]	> 99.5

The experimental NPCR value of proposed method is similar to the existing methods suggested in [12], [26], [27], and approximately close to the methods proposed in [23], [24] and [25].

TABLE VII

UACI VALUES OF PROPOSED METHOD (in %)

Cipher Images	UACI Value
C_1 Vs C_2	30.5472
C_1 Vs C_3	30.5365
C_1 Vs C_4	30.0617
C_2 Vs C_3	30.7639
C_2 Vs C_4	30.1261
C_3 Vs C_4	33.1652

Where, C_1, C_2, C_3, and C_4 are encrypted Lena images using baboon, peppers, cameraman, and coin as key images, respectively.

TABLE VIII

UACI VALUES OF EXISTING METHODS (in %)

Existing Methods	UACI Value
Ref. [12]	28.6210
Ref. [23]	32.069
Ref. [24]	33.58
Ref. [25]	32.821
Ref. [26]	> 24.94
Ref. [27]	> 33.4

The measured UACI value of the proposed method is approximately close to the existing methods [25], [23], [24], and [27], and better than the methods suggested in [26], and [12].

VI. CONCLUSION

In this work, a novel encryption method is applied on digital images and found to give significantly improved performance. A comparable performance improvement is obtained in histogram, correlation, key space, and encryption speed. The histograms of the encrypted images are flat and so it is difficult to identify images by analyzing the histograms. The cross correlation value indicates that there is a weak relationship between the original and encrypted images. The correlation between adjacent pixels in the encrypted image is approximately close to zero. The cut test and dispersion test results show that the proposed method provides acceptable pixel permutation. The proposed method has approximately 10^{509} key space for an image of size 256 x 256 pixels to resist the brute force attack. The real time implementation choices confirm that the proposed method consumes less encryption time and could be robust for real time applications. The values of NPCR and UACI are greater than 99.4% and 30%, respectively, and thus the proposed method is resistant to differential attack.

REFERENCES

[1] Alexander J. Barelka, Anand Jeyaraj, and Ryan G. Walinski, "Content Acceptance Model and New Media Technologies" *Journal of Computer Information Systems*, vol. 53, no. 3, pp. 56-74, 2012.

[2] Sang M. Lee, Seong No Yoon, and Jongheon Kim, "The Role of Pluralistic Ignorance in Internet Abuse", *Journal of Computer Information Systems*, vol. 48, no. 3, pp. 38-43, 2008.

[3] Terry Smith, Alex Koohang, and Robert Behling, "Understanding and Prioritizing Technology Management Challenges", *Journal of Computer Information Systems*, vol. 51, no. 1, pp. 91-98, 2010.

[4] Han Shuihua, Yang Shuangyuan, "An Asymmetric Image Encryption Based on Matrix Transformation, *ECTI Transactions on Computer and Information Technology*, vol. 1, no. 2, pp.126-133, 2005.

[5] S.S Maniccam, N.G Bourbakis, "Image and Video Encryption using Scan Patterns", *Journal of Pattern Recognition Society*, vol. 37, no. 4, pp.725–737, 2004.

[6] A. Mitra, Y.V Subba Rao, and S.R.M Prasanna, "A New Image Encryption Approach using Combinational Permutation Techniques", *International Journal of Electrical and Computer Engineering*, vol. 1, no. 2, pp.127–131, 2006.

[7] Bibhudendra Acharya, Soraj Kumar Panigrahy, Sarat Kumar Patra, and Ganapati Panda, "Image Encryption using Advanced Hill Cipher Algorithm", *International Journal of Recent Trends in Engineering*, vol. 1, no. 1, pp. 663-667, 2009.

[8] Chin-Chen Chang, Min-Shian Hwang, and Tung-Shou Chen, "A New Encryption Algorithm for Image Cryptosystems", *Journal of Systems and Software*, pp.83-91, 2001.

[9] Guodong Ye, "An Efficient Image Encryption Scheme Based on Logistic Maps", *International Journal of Pure and Applied Mathematics*, vol. 55, no. 1, pp.37-47, 2009.

[10] Haojiang Gao, Yisheng Zhang, Shuyun Liang, and Dequn Li, "A New Chaotic Algorithm for Image Encryption, *Elsevier Science Direct*, vol. 29, no. 2, pp.393-399, 2006.

[11] A. Kanso, M. Ghebleh, "A Novel Image Encryption Algorithm based on a 3D Chaotic Map, *Elsevier*, vol. 17, no. 7, pp.2943-2959, 2012.

[12] Khaled Loukhaoukha, Jean-Yves Chouinard, and Abdellah Berdai, "A Secure Image Encryption Algorithm Based on Rubik's Cube Principle", *Journal of Electrical and Computer Engineering*, vol. 20, no. 12, pp.1-13, 2011.

[13] Liu Hongjun, and Wang Xingyuan, "Color Image Encryption based on One-Time Keys and Robust Chaotic Maps", *Journal of Computers and Mathematics with Applications*, pp. 3320-3327, 2010.

[14] Mohammad Ali Bani Younes, Aman Jantan, "Image Encryption using Block-Based Transformation Algorithm", IAENG International *Journal of Computer Science*, vol. 35, no. 1, pp.15-23, 2008.

[15] Sanfu Wang, Yuying Zheng, and Zhongshe Gao, "A New Image Scrambling Method Through Folding Transform", *IEEE International Conference on Computer Application and System Modeling*, Taiyuan, pp.v2-395-399, 22-24 Oct. 2010.

[16] G.A Sathishkumar, K. Bhoopathy, and R. Sriraam, "Image Encryption Based on Diffusion and Multiple Chaotic Maps", *International Journal of Network Security & its Applications*, vol. 3, no. 2, pp.181-194, 2011.

[17] Sesha Pallaviindrakanti, P.S Avadhani, "Permutation based Image Encryption Technique", *International Journal of Computer Applications*, vol. 28, no. 8, pp.45-47, 2011.

[18] Shao Liping, Qin Zheng, Qin Jun, and Li Huan, "Image Scrambling Algorithm based on Random Shuffling Strategy", *IEEE International Conference on Industrial Electronics and Applications*, Singapore, pp.2278-2283, 3-5 June 2008.

[19] Tzung-Her Chen, Kuang-Che Li, "Multi-Image Encryption by Circular Random Grids", *International Journal of Information Sciences*, vol. 189, pp.255-265, 2012.

[20] Wenping Guo, "A New Digital Image Scrambling Encryption Algorithm based on Chaotic Sequence", *IEEE International Conference on Computer Research and Development*, Shanghai, pp.v1-399-401, 11-13 March 2011.

[21] Xiaomin Wang, Jiashu Zhang, "An Image Scrambling Encryption using Chaos-Controlled Poker Shuffle Operation, *IEEE International Symposium on Biometrics and Security Technologies*, Islamabad, pp.1-6, 23-24 April 2008.

[22] Douglas R Stinson, "*Cryptography: Theory and Practice*", Chapman & Hall, New York, 2002.

[23] H.T Panduranga, S.K Naveen Kumar, "Hybrid Approach for Image Encryption using SCAN Patterns and Carrier Images", International Journal on Computer Science and Engineering, vol. 02, no. 02, pp. 297-300, 2010.

[24] P. Vidhya Saraswathi, M. Venkatesulu, "A Block Cipher Algorithm for Multimedia Content Protection with Random Substitution using Binary Tree Traversal", *Journal of Computer Science*, vol.8, no. 9, pp. 1541-1546, 2012.

[25] A. Kumar, M. K. Ghose, "Extended Substitution-diffusion based Image Cipher using Chaotic Standard Map," *Communications in Nonlinear Science and Numerical Simulation*, vol. 16, no. 1, pp. 372-382, 2011.

[26] C. K. Huang, H. H. Nien, "Multi Chaotic Systems based Pixel Shuffle for Image Encryption," *Optics Communications-Elsevier*, vol. 282, no. 11, pp. 2123-2127, 2009.

[27] X. Liao, S. Lai, and Q. Zhou, "A Novel Image Encryption Algorithm based on Self-adaptive Wave Transmission", *Signal Processing*, vol. 90, no. 9, pp. 2714-2722, 2010.

T.Sivakumar received his B.Sc degree in Mathematics from Manonmaniam Sundaranar University in 1998, and M.C.A degree from Bharathidasan University in 2002. He received his second master degree M.E in Computer Science and Engineering from Anna University in 2009. He is currently working as an Assistant Professor (Sr.Gr) in the Department of Information Technology, PSG College of Technology, Coimbatore-641004, India.

R.Venkatesan was born in Tamilnadu, India, in 1958. He received his B.E (Hons) degree from Madras University in 1980. He completed his Masters degree in Industrial Engineering from Madras University in 1982. He obtained his second Masters degree MS in Computer and Information Science from University of Michigan, USA in 1999. He was awarded with PhD from Anna University, Chennai in 2007. He is currently Professor and Head in the Department of Computer Science and Engineering at PSG College of Technology, Coimbatore-641 004, India.