International Journal of Computer Science and Engineering Communications,
Vol.5, Issue.2 (2017):Page.1511-1520
IJCSEC.COM

# Image Encryption based on Pixel Permutation and Generation of True Random Numbers

**Aarthi G.V[1] and A.Geetha[2]**

*[1]Assistant Professor, [2] Assistant Professor,*
*Department of Information Technology,*
*Sri Krishna College of Technology, Coimbatore, India*
*[1]g.v.aarthi@skct.edu.in, [2]geetha.a@skct.edu.in*

***Abstract:*** *The rapid growth of internet allows large multimedia files to be easily stored and transmitted. The multimedia data plays a major role in medical imaging, telemedicine, military and satellite communications. It is important to protect the multimedia data from unauthorized disclosure during the transmission. Traditional cryptosystems, such as AES, DES, and IDEA provides security for text encryption but are not ideal for image encryption. Image encryption methods based on bit permutation, pixel permutation, and block transformations are introduced by various researchers in the literature. An image encryption method for permuting the pixels based on Knight Travel path and True Random Numbers is proposed in this paper. The Knight travel is a pattern in which the path of a knight around a chess board without revisiting any particular node is taken and it is used as a SCAN pattern. When this pattern is applied to an image, the obtained image will be a scrambled image. The random key stream is the one which consists of random values used in image encryption for XOR operation. The audio file is considered as key which has a range of amplitude values producing the random key stream. The scrambled image is further XORed with the random key stream to get highly encrypted image.*

***Keywords*** *-image, keystream, knight, pixels, permutation, random numbers generator,*

## 1. Introduction

The growth of internet provides a new way for spreading the digital information especially in form of image, audio and video. With the fast evolution of digital data exchange and increased usage of multimedia images, it is essential to protect the confidential image data from unauthorized access. Cryptography is popularly the art and science of secret writing and this enables us to store and transmit sensitive information across insecure networks. Image encryption is widely used in internet, medical imaging, telemedicine and military communications [5]. Various techniques such as permutation of pixels and blocks, transformation, chaotic systems can be used for image encryption.

### 1.1 About SCAN Methodology

The scan is a formal language-based two dimensional spatial accessing methodology which can represent and generate a large number of wide varieties of scanning paths. The SCAN language uses four basic scan patterns such as continuous raster (C), continuous diagonal (D), continuous orthogonal (O), and spiral (S), and this is shown in Figure 1.

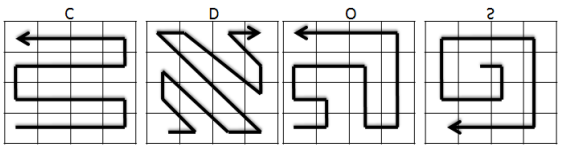



**Fig.1. Basic Scan Patterns**

### 1.2 About Knight Travel Path

The chess piece which does not move in a straight line is the Knight; the legal move for a knight is two spaces in one direction, then one in a perpendicular direction[2]. The Knight Travel path is a journey in which each cell of the chessboard is visited only once and should not be revisited. If the knight ends on a square that is one move of a knight from beginning following the same path, the tour is closed, else it is open. A simple graph can be used to determine the path of a knight. Figure shows the possible and legal movements of a knight on the chessboard.

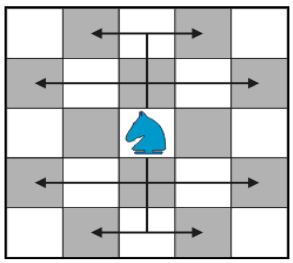


**Fig.2. Possible movements of a Knight**

### 1.3 Adaption of Knight Travel Path for Pixel Scanning

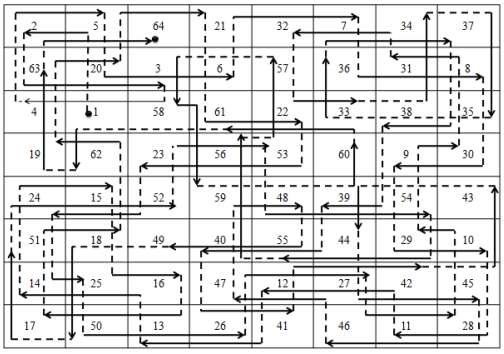
The knight travel path represents a sequence of moves of the knight on a chessboard such every square is visited only once. The Knight Travel path is used to scan the pixels of an image in the appropriate pattern which permute the pixels. The knight travel scans the pixels block by block to attain the scrambled image. Figure 3(a) represents the initial input matrix of an image. Figure 3(b) shows the sample scan path for the starting coordinate (3, 2). Figure 3(c) represents the coordinate corresponding to 3(b). Figure 3(d) shows the output matrix after applying the scan path on 3(a) and the result shows that the image is scrambled.

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|
| 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
| 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 |
| 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 |
| 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 |
| 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 |
| 49 | 50 | 51 | 52 | 53 | 54 | 55 | 56 |
| 57 | 58 | 59 | 60 | 61 | 62 | 63 | 64 |

(a) Original Matrix

| 3,2 | 1,1 | 2,3 | 3,1 | 1,2 | 2,4 | 1,6 | 2,8 |
|---|---|---|---|---|---|---|---|
| 4,7 | 6,8 | 8,7 | 7,5 | 8,3 | 7,1 | 5,2 | 7,3 |
| 8,1 | 6,2 | 4,1 | 2,2 | 1,4 | 3,5 | 4,3 | 5,1 |
| 7,2 | 8,4 | 7,6 | 8,8 | 6,7 | 4,8 | 2,7 | 1,5 |
| 3,6 | 1,7 | 3,8 | 2,6 | 1,8 | 3,7 | 5,6 | 6,4 |
| 8,5 | 7,7 | 5,8 | 6,6 | 7,8 | 8,6 | 7,4 | 5,5 |
| 6,3 | 8,2 | 6,1 | 5,3 | 4,5 | 5,7 | 6,5 | 4,4 |
| 2,5 | 3,3 | 5,4 | 4,6 | 3,4 | 4,2 | 2,1 | 1,3 |

(c) Scan Coordinates



(b) Scan Path

| 2 | 5 | 64 | 21 | 32 | 7 | 34 | 37 |
|---|---|---|---|---|---|---|---|
| 63 | 20 | 3 | 6 | 57 | 36 | 31 | 8 |
| 4 | 1 | 58 | 61 | 22 | 33 | 38 | 35 |
| 19 | 62 | 23 | 56 | 53 | 60 | 9 | 30 |
| 24 | 15 | 52 | 59 | 48 | 39 | 54 | 43 |
| 51 | 18 | 49 | 40 | 55 | 44 | 29 | 10 |
| 14 | 25 | 16 | 47 | 12 | 27 | 42 | 45 |
| 17 | 50 | 13 | 26 | 41 | 46 | 11 | 28 |

(d) Output Matrix

**Fig.3.Pixel Permutation based on Knight Travel Path**

### 1.4 Role of Random Numbers in Cryptography

Random numbers are useful for various purposes, such as generating data encryption keys, simulating and modeling complex phenomena and for selecting random samples from larger data sets. Generation of random numbers consists of two main approaches using computers. They are the Pseudo-Random Number Generators (PRNGs) and True Random Number Generators (TRNGs)[12]. Many computers are built with inputs that digitize some real world analog sources, such as sound from microphone, audio, etc. If the system has enough gain to detect anything, such input can provide reasonably high quality random bits [11]. Table 1 gives the comparison between the TRNG and PRNG, and in case of cryptography, TRNGs are considered as most suitable[12].

**Table.1. Comparison of PRNG and TRNG**

| Characteristic | PRNG | TRNG |
|---|---|---|
| Efficiency | Excellent | Poor |
| Deterministic | Deterministic | Nondeterministic |
| Periodicity | Periodic | Aperiodic |

### 1.5 Proposed Random Number Generator

True Random Number Generators extract randomness from physical phenomena and introduce it into a computer [6]. In the proposed method, audio file is considered as one such true random source to generate range of random values from the amplitude. After the removal of redundant amplitude values, the set of true random numbers are created by two methods. This random key stream is further used to perform XOR operation.

#### 1.5.1 Method 1

The first method generates the random key stream by taking the fractional part of the amplitude value of the audio file and normalizing them. In case of negative value, the value is made positive.

Example: 0.2542 to 254
-0.2537 to 0.2537 to 254
0.3422 to 342 mod 255 to 87

#### 1.5.2 Method 2

The second method generates the random key stream by converting the fractional part into binary stream and by consideration of the first eight bits to decimal.

Example: 0.2546 to 100111110010 to 10011111 to 159

If necessary, more than 8 bits can also be taken. By considering the 16 bits, two decimal values can be attained, which will operate on two pixel values.

## 2. Related Work

In block permutation, the image can be decomposed into blocks, and blocks are permuted. For better encryption the block size should be higher. If the blocks are very small, then the objects and its edges don't appear clearly. In block permutation the blocks are permuted horizontally in the image. The permutation of blocks along vertical side is also similar to horizontal side block permutation. At the receiver the original image can be obtained by the inverse permutation of the blocks [1]. The pattern exists such that each node is visited only once and should not be revisited [2]. Encryption of an image can be done at different stage or in multiple stages and in multiple ways. If the encryption process is only in single stage then security is less as compare to multistage encryption using text as a key word with the help of ASCII code to produce carrier image [3].A hybrid technique for image encryption which employs the concept of carrier image and SCAN patterns generated by SCAN methodology is used. The SCAN is a formal language-based two-dimensional spatial-accessing methodology which can efficiently specify and generate a wide range of scanning paths.

The carrier image is the one which consists of confidential information anditis created with the help of alphanumeric keyword. Each alphanumeric key will be having a unique 8 bit value generated by 4 out of 8-code from the lookup table. The carrier image is created and scan methodology is applied to either original image or carrier image, after which the XOR operation is done to obtain highly distorted encrypted image [5]. The image encryption method is based on permutation of the pixels of the image using scan keys and replacement of the pixel valuesis presented in the paper [7]. The permutation is done by scan patterns generated by the SCAN methodology and the pixel values are replaced using a simple substitution rule to produce an iterated product cipher which adds confusion and diffusion properties to the encryption method. Varsha Bhatt et al.[8]Presented a block based transformation architecture to increase security level of the encrypted images. Encryption algorithm is divided into three phase.

## 3. The Proposed Method

The proposed method aims at design and development of a new image encryption based on pixel permutation using knight travel path and XOR operation using true random numbers generated from the audio file. Initially the original image is taken as input and divided into blocks of size NxN. The pixels of each block are permuted based on the knight travel path with starting position of the knight as key1 to produce the scrambled image. The random key stream is produced with the support of audio file (key2) which is further shuffled by scan pattern key3 (k3) to produce a highly shuffled random key stream. The random key stream is XORed with the scrambled image to produce pre-cipher image. The obtained image is further scrambled using key 4 (k4) to attain the final cipher image. The overall working model of the proposed method is shown in Figure 4.In the flow diagram, key2 represents the audio file chosen by the communicating parties, which act as a symmetric key.Both key 3 and key 4 can be any one of the basic scan pattern such as continuous raster, diagonal, orthogonal or spiral, and they are optional. In this paper, the diagonal scan pattern is used. The proposed method is tested with JPEG images of size 256 x 256 pixels and blocks of size (N) 8, 16, and 32.

### 3.1 Generation of True Random Numbers

The true random numbers are generated from the audio files by the two methods. The first method generates the random key stream by taking the value after the decimal part of the amplitude value and normalizing them by removing the redundant values. The second method generates the random key stream by converting the fractional part of the amplitude value into binary stream and by consideration of the first eight or sixteen bits as decimal value.

#### 3.1.1 Creation of True Random Numbers: Method 1 (TRN1)

Function Name: TRN1()
Input: Audio file
Output: Random key stream
Step 1: Load the audio file k2.
Step 2: Let keyStream[] be an array to store the generated random numbers.
Step 3:Fetch amplitude values from the audio file.
Step 4:Remove the redundancy from the fetched amplitude values.
Step 5:Extractand convert the fractional partas integer value.
Step 6:If the integer value is greater than 255 perform mod 256.
Step 7: Store the integer in the keyStream array.
Step 8: Return the array keyStream.

#### 3.1.2 Creation of True Random Numbers: Method 2 (TRN2)

Function Name: TRN2()
Input: Audio file

Output: Random key stream
Step 1:Load the audio file k2.
Step 2: Let keyStream[] be an array to store the generated random numbers.
Step 3: Fetch the amplitude values.
Step 4: Remove the redundant amplitude values.
Step 5: Convert the fractional part into binary stream.
Step 6: Convert MSB 8 or 16 bits into decimal value, and store in the array keyStream.
Step 7: Return the array keyStream.

### 3.2 Proposed Encryption Algorithm

Function Name: Encryption()
Input: Plain Image, Scan Path, K1, K2, K3, and K4.
Output: Cipher Image
Step 1: Let I[r c] be the original image and r and c be the number of rows and columns.
Step 2: Input the block size N and divide image I into blocks of size N*N.
Step 3: Get the starting co-ordinate of the knight, and select the corresponding scan path (k1).
Step 4: Pixel permutation is performed on each block based on k1 to attain scrambled image.
Step 5: Read the audio file in key 2 (k2).
Step 6: Generate the random key stream by calling TRN1(k2) (or) TRN2(k2)
Step 7: Obtain the distorted random key stream with basic scan pattern (key 3).
Step 8: Generate the pre-cipher image by XORing the scrambled image obtained in step (4) and the distorted random key stream obtained in step (7).
Step 9: Generate the cipher image by applying any basic scan pattern (key 4).
Step 10: Return cipher image.

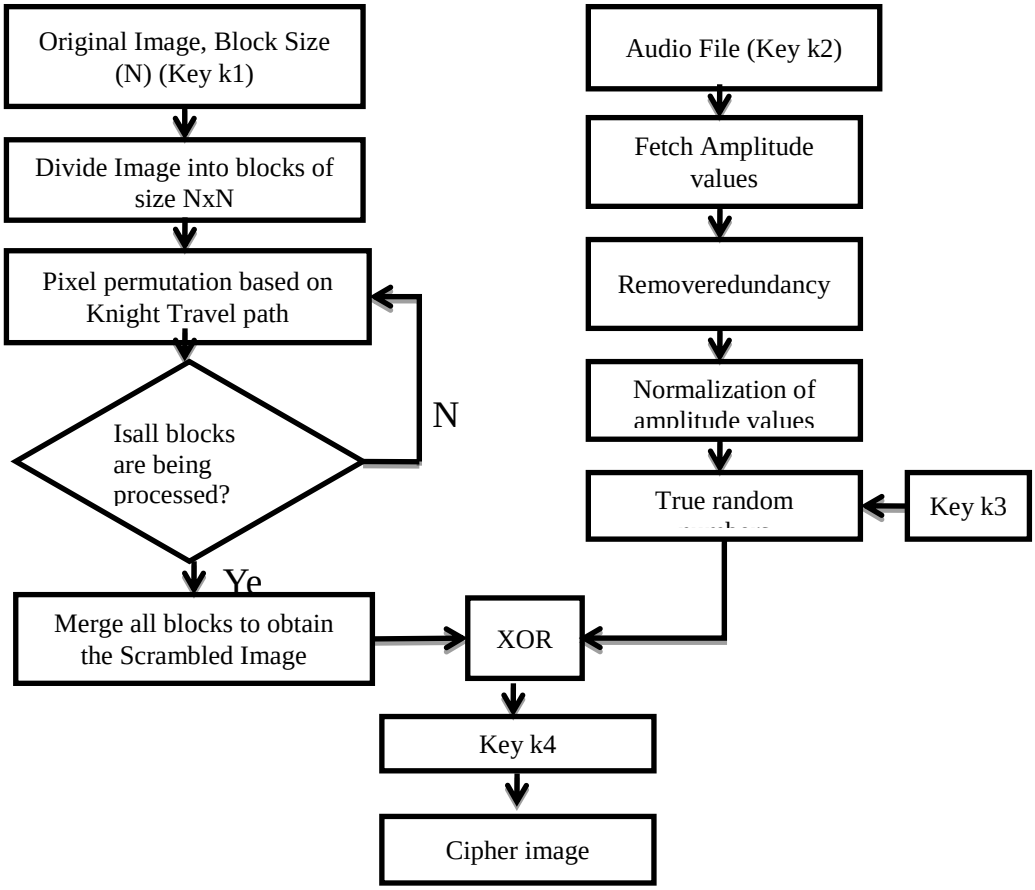
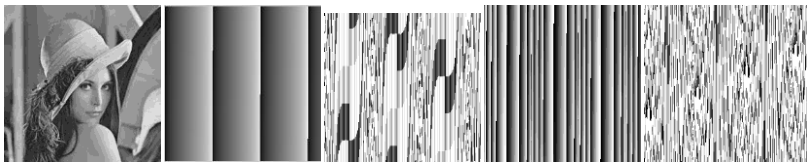



**Fig.4. Flow Diagram of Proposed Method**

## 4. Experimental Results

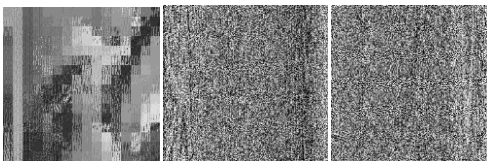
The proposed method is implemented using Matlab r2011a with Intel(R) Corei3 processor, RAM of 4GB, hard disk of 320GB, clock speed of 2.40GHz and Windows 7(64-bit) operating systems.The proposed methodis tested with many audio files, and the results are presented for two audio files namely bark.wav (audio1) and hero.wav(audio2). Figure 6shows the original Lena image and the random key streams derived from audio1 and audio2using method1 and method2.



(a) (b) (c) (d) (e)

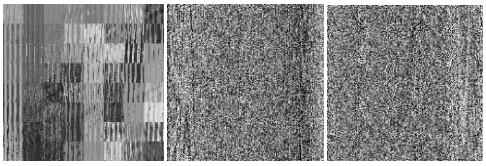
**Fig.6. (a) Original Image (b) Random key Stream from Audio1using Method1 (c) Random key Stream from Audio1using Method2 (d) Random key Stream from Audio2 using Method1 (e) Random key Stream from Audio2 using Method2**

Figures 7 to 8 show the results of permutation of pixels(P), audio2 for random key stream generation, and by making use of the keys k1,k2,k3, and k4 with block size N=16, and N=32. Where, P represents permutation, k1 represents the starting co-ordinate of the knight, k2 is the audio file shared by the communicating parties, k3 and k4 are the scan patterns.



**(a)** (b) (c)

**Fig.7. (a) Pixel Permuted Image (b) Cipher Image using Method1 (c) Cipher Image using Method2**



(a) (b) (c)

**Fig.8. (a) Pixel Permuted Image (b) Cipher Image using Method1 (c) Cipher Image using Method2**

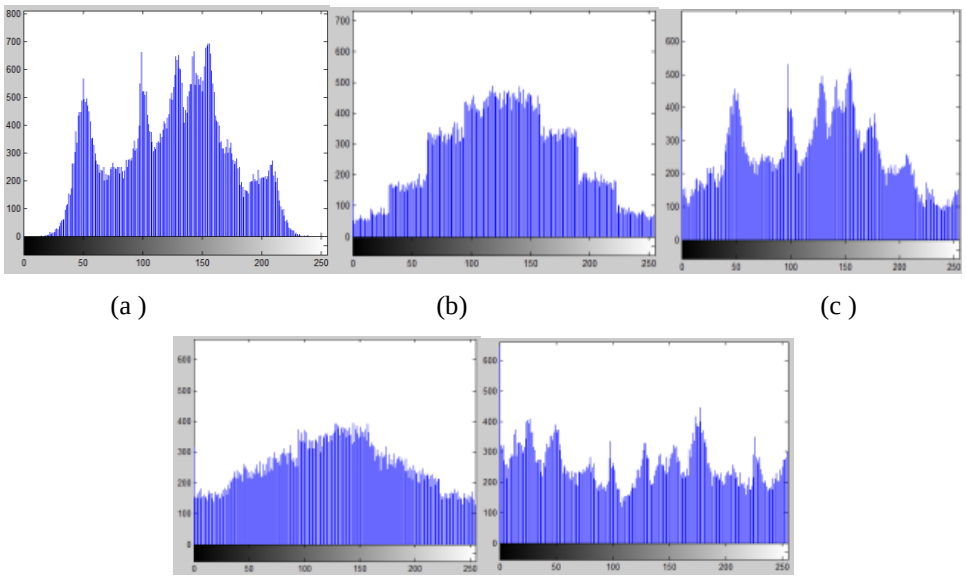
The results produced on considering various audio files using method1 and method2 with block size N= 16, and 32 show variations on attaining the encrypted image such that the property of diffusion is satisfied. As the block size increases better pixel permutation is attained.

## 5. Performance Analysis

In order to validate the strength of proposed approach, the evaluation parameters like histogram, correlation, NPCR, and UACI testing are performed. The results were analyzed and compared with existing image encryption methods.

### 5.1 Analysis of Histogram

An image histogram is a graphical representation of the number of pixels in an image as a function of their intensity. Image histogram is an important tool for inspecting images. The histogram of the cipher image must be flat and significantly different from the respective histograms of the plain image.



(a ) (b) (c )

(d) (e)

**Fig.9. (a) Histogram for Original Image of Lena Image, Histogramof Encrypted Image using: (b) Audio1 and method1 (c) Audio1 and method2(d) Audio2 and method1 (e) Audio2 and method2**

## 5.2 Adjacent pixel Correlation

Correlation determines the relationship between the original image and cipher image. It is a measure that computes degree of similarity between the images. The correlation coefficient must be very low or close to zero. If correlation coefficient is one, the two images are identical andare in perfect correlation. In case of perfect correlation, encryption process completely fails because the encrypted image is same as the plaintext image. When correlation coefficient is -1 then encrypted image is negative of plaintext image.The result is optimal when the values of the plain image is close to 0.9 and the values of cipher image is close to zero.

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

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

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

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

Where, $\gamma_{xy}$- correlation co-efficient, Cov(x,y) is the covariance of x and y, and x, y are the gray-scale values of pixels in plain and cipher images.
Table2and Table 3gives the adjacent pixel correlation values of the proposed method for audio file bark.wav as k2 using method 1 and method 2 for random key generation.

The result of correlation value using bark.wav as key k2using method1andmethod2 is found optimal, and hence the correlation between adjacent pixels is reduced. Table4and Table5gives the adjacent pixel correlation values of the proposed method for hero.wav as key k2 using method1 and method2 for random key generation.

**Table.2. Adjacent Pixel Correlation Values for Audio1 - Method1**

| Block Size(N), Starting Co-ordinate | Image | Correlation Values | | |
|---|---|---|---|---|
| | | Horizontal | Vertical | Diagonal |
| N=8 (3,2) | Cameraman Original | 0.9875 | 0.9949 | 0.9845 |
| | Cameraman Encrypted | 0.0208 | 0.0119 | 0.0173 |
| | Lena Original | 0.9846 | 0.9756 | 0.9690 |
| | Lena Encrypted | 0.0200 | 0.0076 | 0.0084 |
| N=16 (12,7) | Cameraman Encrypted | 0.0190 | 0.0015 | 0.0114 |
| | Lena Encrypted | 0.0066 | 0.0134 | 0.0022 |
| N=32 (20,5) | Cameraman Encrypted | 0.0290 | 0.0101 | 0.0273 |
| | Lena Encrypted | 0.0305 | 0.0120 | 0.0098 |

**Table.3. Adjacent pixel Correlation Values for Audio1 – Method2**

| Block Size(N), Starting Coordinate | Image | Correlation Values | | |
|---|---|---|---|---|
| | | Horizontal | Vertical | Diagonal |
| N=8 (3,2) | Cameraman Encrypted | 0.0866 | 0.0181 | 0.0039 |
| | Lena Encrypted | 0.0611 | 0.0081 | 0.0430 |
| N=16 (12,7) | Cameraman Encrypted | 0.0154 | 0.0823 | 0.0078 |
| | Lena Encrypted | 0.0527 | 0.0050 | 0.0074 |
| N=32 (20,5) | Cameraman Encrypted | 0.1461 | 0.0067 | 0.0051 |
| | Lena Encrypted | 0.0369 | 0.0076 | 0.0023 |

**Table.4. Adjacent pixel Correlation Values for Audio2 – Method1**

| Block Size(N), Starting Coordinates | Image | Correlation Values | | |
|---|---|---|---|---|
| | | Horizontal | Vertical | Diagonal |
| N=8 (3,2) | Cameraman Encrypted | 0.0596 | 0.0104 | 0.0070 |
| | Lena Encrypted | 0.0811 | 0.0034 | 0.0178 |
| N=16 (12,7) | Cameraman Encrypted | 0.0452 | 0.0147 | 0.0040 |
| | Lena Encrypted | 0.0467 | 0.0048 | 0.0118 |
| N=32 (20,5) | Cameraman Encrypted | 0.0480 | 0.0068 | 0.0012 |
| | Lena Encrypted | 0.0303 | 0.0134 | 0.0021 |

**Table .5. Adjacent Pixel Correlation Values for Audio2 - Method 2**

| Block Size(N),Starting Coordinates | Image | Correlation Values | | |
|---|---|---|---|---|
| | | Horizontal | Vertical | Diagonal |
| N=8 (3,2) | Cameraman Encrypted | 0.0426 | 0.0180 | 0.0259 |
| | Lena Encrypted | 0.0821 | 0.0268 | 0.0447 |
| N=16 (12,7) | Cameraman Encrypted | 0.0273 | 0.0168 | 0.0319 |
| | Lena Encrypted | 0.0688 | 0.0190 | 0.0230 |
| N=32 (20,5) | Cameraman Encrypted | 0.0188 | 0.0127 | 0.0318 |
| | Lena Encrypted | 0.0165 | 0.0114 | 0.0318 |

Table 6 gives the correlation values of adjacent pixels for few existing methods. It is found that the correlation value of proposed method is similar to the methods [5], [9], and [10].

**Table.6.Adjacent Pixel Correlation Values ofExistingMethods**

| Existing Methods | Horizontal | Vertical | Diagonal |
|---|---|---|---|
| [5] | 0.0263 | 0.0163 | 0.0114 |
| [9] | 0.0068 | 0.0091 | 0.0063 |
| [10] | 0.0781 | 0.0785 | 0.0683 |

### 5.3 NPCR

The Number of Pixel Change Rate (NPCR) is defined as variance rate of pixels between the original image and cipher image. For the plaintext image $I_o(i,j)$ and encrypted image $I_{ENC}(i,j)$, the equation (2) gives the mathematical expression for NPCR.

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

Where, W is the width of the image and H is height of image. If $I_o(i,j) = I_{ENC}(i,j)$, then D(i,j) =0, otherwise D(i,j) = 1. The optimal result is obtained when value is beyond 97% [4].
Table 7 gives the NPCR value of the proposed method for hero.wav using method1 and method 2 for random key generation.

**Table.7. NPCR Values for Proposed Method**

| Block Size (N) | Knight Starting Co-ordinate | Image | Audio | |
|---|---|---|---|---|
| | | | Method 1 | Method 2 |
| N=8 | (3,2) | Cameraman | 99.4690 | 99.3240 |
| | | Lena | 99.5026 | 99.5560 |
| N=16 | (12,7) | Cameraman | 99.4827 | 99.3088 |
| | | Lena | 99.5056 | 99.5468 |
| N=32 | (20,5) | Cameraman | 99.5361 | 99.3103 |
| | | Lena | 99.5163 | 99.5911 |

Table 8gives the NPCR values of the few existing methods. It is found that the NPCR value obtained for the proposed method is optimal and greater than 99%.

**Table. 8 NPCR Value of Existing Methods**

| S.No. | Existing Methods | NPCR Value |
|---|---|---|
| 1 | [5] | 99.7025 |
| 2 | [9] | 99.5850 |
| 3 | [10] | 98.4754 |

## 6. Conclusion

A novel image encryption method based on pixel permutation and true random numbers is proposed in this paper. The results obtained with correlation, NPCR of the proposed method are approximately nearing that of the existing methods. The cross correlation values indicate there is a weak relationship between original and encrypted images. The correlation between adjacent pixels is approximately close to zero. The NPCR results in an optimal value greater than 99%.The results may vary based on the audio file chosen for random key stream generation (key 2).

## References

[1]A. Mitra, Y. V. SubbaRao and S. R. M. Prasanna, "A New Image Encryption Approach Using Combinational Permutation Techniques",International Journal of Electrical and Computer Engineering, Vol.01, No.02, 2006.
[2]Ben Hill, Knight's Tours Kevin Tostado December 18, 2004.http://faculty.olin.edu/~sadams/DM/ktpaper.pdf , accessed on July 23, 2013.

[3]G. Nagaraju and T. V. Hyma Lakshmi, "Image Encryption using Secret-Key images and SCAN Patterns", International Journal in Advances in Computer, Electronic, Vol.02, 2012.
[4]Jawad Ahmad andFawad Ahmed, "Efficiency Analysis and SecurityEvaluation of Image Encryption Schemes",International Journal of Video & Image Processing and Network Security, Vol.12, No.04, 2012.
[5]Panduranga H.T and Naveen Kumar S.K, "Hybrid Approach for Image Encryption using SCAN Patterns and Carrier Images", International Journal on Computer Science and Engineering, Vol. 02, No. 02, 2010.
[6]Roger Morrison, "Design of a True Random Number Generator using Audio Input", Journal of Cryptology, Vol.01, No.01, 2001.
[7]S.S.Maniccam and N.G. Bourbakis, "Image and Video Encryption using SCAN patterns",Journal of Pattern Recognition, 725 – 737, 2004.
[8]Varsha Bhatt and Gajendra Singh Chandel,"Implementation of New Advanced Image Encryption Algorithm to Enhance Security of Multimedia Component", International Journal of Advanced Technology & Engineering Research, Vol 2, Issue 4, 2012.
[9]KhaledLoukhaoukha, Jean-Yves Chouinard, and AbdellahBerdai, "A Secure Image Encryption Algorithm Based on Rubik's Cube Principle", Journal of Electrical and Computer Engineering, 2011, pp. pp.1-13.
[10]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, 2011, pp. 181-194.
[11]William Stallings, "Cryptography and Network Security: Principles and Practice",Pearson Education, New Delhi, 2011.
[12]www.randomorg.com, accessed on July 20, 2013.