International Journal of Latest Trends in Engineering and Technology
Vol.(10)Issue(4), pp.005-013
DOI: http://dx.doi.org/10.21172/1.104.02
e-ISSN:2278-621X

IMAGE ENCRYPTION USING RANGOLI BASED SCAN PATTERN AND RANDOM NUMBER

T. Sivakumar[1], T.Anusha[2], S.Vasantha Keerthana[3]

Abstract-The sensitive data shared through public network are vulnerable to security attacks. Cryptography provides an effective means of protecting sensitive data communicated over public network.This paper presents an image encryption technique using rangoli based scan pattern and random number. The original image is scrambled by pixel permutation with rangoli based scan pattern. The cipher image is obtained by XORing the scrambled image with the random numbers. Random numbers are generated by using the Blum Blum Shub (BBS)generator. The obtained result is analyzed and compared with the existing image encryption methods.

Keywords - Image Encryption, Pixel Permutation, Rangoli,Scan pattern, Random Number

1. INTRODUCTION

The need for data privacy and security has increased in digital communication because of increase in security breaches. Transmission of sensitive information over the public networks has increased due to the advancement of computer networking and communication technologies [7].Cryptography is used to protect the data transmitted through an insecure network. Cryptography is concerned with developing algorithms that conceal the context of message from all except the sender and recipient.Symmetric key encryption and asymmetric key encryption are the two types of encryption methods. Symmetric encryption transforms plaintext into ciphertext using a secretkey and an encryption algorithm. Using the same key and a decryption algorithm, the plaintext is recovered from the ciphertext [20].In asymmetric key scheme, each user uses a pair of keys called as public key and private key. Data Encryption Standard (DES) is a symmetric key encryption technique used for the encryption of electronic data and it is now considered to be insecure because of brute force attack. The Advanced Encryption Standard (AES) proposed by Daemen and Rijmen is also a symmetric key algorithm for fixed block size of 128-bits and key size of 128,192 or 256 bits. The International Data Encryption Algorithm (IDEA) symmetric key operates on 64-bit block using 128-bit key [20]. The above algorithms are efficient when the volume of input data is small. These algorithms are widely used to encrypt text messages and not desirable to encrypt images [18].

Images have special features such as bulk capacity, high redundancy and high correlation among pixels. The high redundancy and bulk capacity make encrypted image vulnerable to attacks via cryptanalysis. An image is scrambled by arranging the bits, pixels and blocks in a suitable manner. The correlation among the bits, pixels and blocks in a given arrangement provides the intelligible information present in the image [13, 19]. Thus, the image encryption methods should focus on reducing the correlation among the bits, pixels, and blocks so that it should be less prone to attacks and the execution time should be minimum [18].

In this paper, a novel image encryption technique is proposed using rangoli based scan pattern and random number.Rangoli based scan pattern is used to perform pixel permutation. In pixel permutation, the pixels position of the original image is rearranged using scan key of size same as the size of the image block.Blum Blum Shub generator is used to generate the needed amount of random numbers. Next, random numbers are XORed with the scrambled image to obtain the cipher image.The cipher images are analyzed to confirm the resistance to security attacks.

The rest of the paper is organized as follows. Literature survey is provided in section II. The preliminary concepts needed for the proposed method is discussed in section III. Section IV provides the proposed image encryption method. Experimental results and analysis are presented in section V. Concluding remarks are given in section VI.

2. LITERATURE SURVEY

In this section, the existing image encryption methods which are based on pixel permutation are presented.

Adrian-Viorel Diaconu et al. [3 presented an image encryption algorithm using the Rubik's cube principle and a digital chaotic cipher. The method resists exhaustive, differential and statistical attacks but has no immunity to the additive noise and cropping attacks. Avi Dixit et al. [4] suggested an image encryption method using permutation and rotational XOR techniques. Here, 8 bit key is generated using pseudorandom index generator. The pixel decimal value is converted into

[1] Department of Computer Science and Engineering, Dr.Mahalingam College of Engineering and Technology, Pollachi, Tamilnadu-642 003, India

[2] Department of Computer Science and Engineering, PSG College of Technology, Coimbatore, Tamilnadu-641 004, India

[3] Microfocus Software Development,Bangalore, India

binary stream which contain 8 bits. Based on the 8 bit key, each pixel of the original image is permuted. Then the entire image is divided into blocks of size 8×8 pixels and the blocks are permutated using same 8 bits key. After that the binary stream is converted into decimal value and considered as cipher image.

Huang and Nien [7] proposed a pixel shuffling method for image encryption based on unpredictable character to reduce the exhaustive key search attack by disordering the distributive characteristics of RGB levels. Sathishkumar et al. [8] presented a pixel shuffling, base 64 encoding based algorithm, which is a combination of block permutation, pixel permutation and value transformation. The crypto system uses a simple chaotic map for key generation, and a logistic map was used to generate a pseudo random bit sequence.

Panduranga et al. [9] suggested a hybrid technique for image encryption by using carrier image and basic scan patterns. The alphanumeric keyword is used to create the carrier image such that each alphanumeric key will have a unique 8 bit value generated by 4 out of 8-code.Khaled Loukhaoukha et al. [11] proposed an image encryption method based on the principle of Rubik's cube. The pixel positions are reordered based on Rubik's cube principle to produce scrambled version of the original image. The bitwise XOR operation is applied to the odd rows and columns and then to even rows and columns of the scrambled image using secret keys to get the cipher image.

Maniccam and Bourbakis (2001) [17] introduced a methodology which performs both lossless compression and encryption of binary and gray-scale images. An overview of SCAN, compression and decompression algorithms, encryption and decryption algorithms and the test results of the methodology are presented. Maniccam and Bourbakis (2004) [16] presented a technique for image and video encryption using SCAN patterns. The image encrypted by SCAN-based permutation of pixels and a substitution rule to form an iterated product cipher.

Lin and Wang [21] presented an image encryption algorithm based on chaos and piecewise linear memristor in Chua′s circuit. Image scrambling and pixel replacement are the two main operations in this encryption algorithm. The chaos-based image encryption methods offer limited security due to small key space.

Banthia et al [2] proposed an image encryption method using pseudo random number generators such as linear congruent generator and linear feedback shift register. Tang et al [19] presented an image encryption using chaotic coupled map lattices along with time-varying delays.Sivakumarand Venkatesan [18] introduced a novel image encryption approach using scan pattern. The scan patterns are derived from the notion of calligraphy.Vidhya Saraswathi and Venkatesulu [12] presented a block cipher algorithm for to encrypt multimedia content using binary tree traversal.

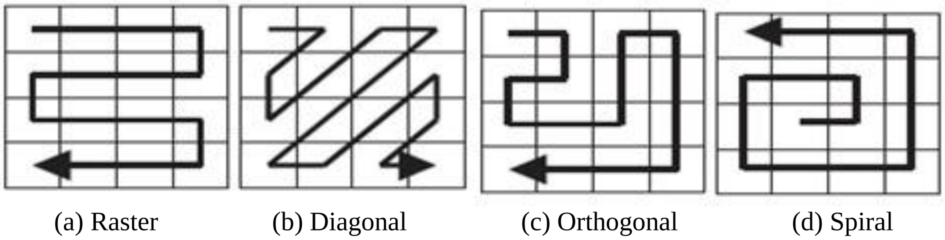
Thus, the image encryption methods based on pixel permutation have been studied and it is observed that there is a need to develop new image encryption methods based on pixel permutation. Hence, a simple and effective image encryption method using rangoli based scan pattern and random number is proposed.

3. PRELIMINARIES

In this section, the basic concepts behind the development of the proposed image encryption method are presented.

3.1 Scan pattern-

Scan pattern is a formal language-based two dimensional spatial-accessing methodology to generate a wide range of scanning paths to permute the pixels of an image. It is also defined as scanning of a two-dimensional array in which each element of the array is accessed exactly once [16, 17]. The pixels of an original image are permuted to obtain the scrambled image using scan patterns. Scan pattern has been used in several image encryption methods to achieve permutation [6, 7, 9, 18].The scan patterns shown in Fig. 1are usually used to permute the pixels of an image.



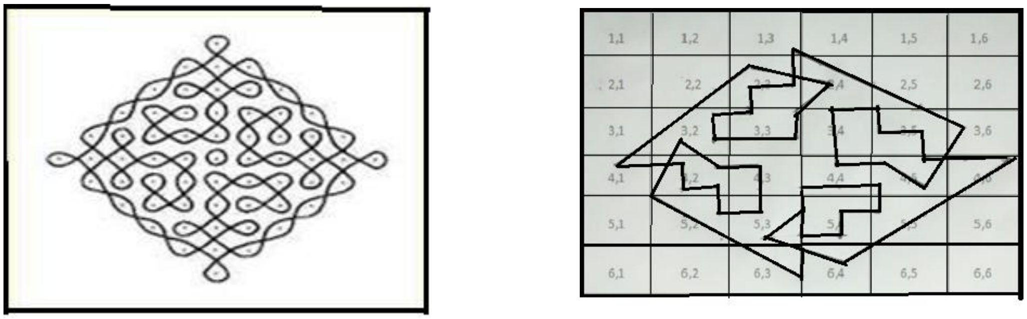
(a) Raster (b) Diagonal (c) Orthogonal (d) Spiral

Figure 1. Basic scan patterns

3.2 Rangoli based scan pattern-

Rangoli is an ancient ephemeral art system practiced throughout India. Designs are drawn onto the ground, usually in chalk or colored powder.Kolams are decorative geometrical patterns that adorn the entrances of households and places of worship especially in South India.Designs are composed of geometric and curvilinear patterns, usually derived from nature. Kolam is a line drawing of curves and loops around a regular grid of points. Kolams have symmetry, patterned repetition, closed continuous curves and curve families, all of which have applications and meaning in mathematics and computer science [14, 15]. With their impeccably logical building up of patterns and their algorithmic nature have attracted to use rangoli for generating scan patterns for pixel permutation. The rangoli based scan pattern is random and there are enormous numbers of rangoli available.

To describe the proposed rangoli based scan pattern, let us consider the rangoli as shown in Fig. 2(a). One such scan pattern derived from the rangoli is shown in Fig. 2(b). Different scan patterns can be generated from the same rangoli by changing the starting point.



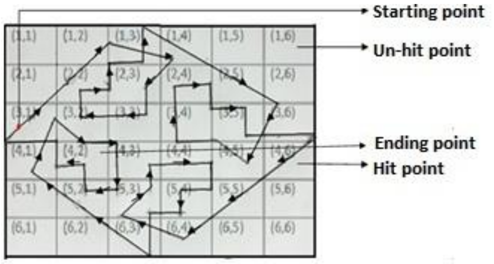
(a)Sample Rangoli (b) Sample Scan pattern derived from rangoli

Figure 2. Proposed rangoli based scan pattern

3.3 Pixel permutation using proposed scan pattern -

In this section, the proposed pixel permutation based on rangoli scan pattern is illustrated. The rangoli based scan pattern is applied on each block of the image, where the block size depends on the chosen rangoli.

To illustrate the proposed pixel permutation, let us consider the scan pattern shown in Fig 3(a). The starting point of the scan pattern is (3,1) and the ending point is at (4,2).The scan coordinates obtained by using this scan pattern is shown in Fig. 3(b).For permutation, let us consider the sample matrix shown in Fig. 3(c). The corresponding scrambled matrix obtained by using this scan pattern is shown in Fig. 3(d). To generate the scan pattern first the hit points are accessed and then the un-hit points are accessed.From the result, it is observed that the elements in the permuted matrix are scrambled for an acceptable level.



(a) Scan pattern

(3,1)	(2,4)	(3,6)	(6,4)	(6,2)	(1,5)
(2,1)	(2,3)	(3,5)	(6,3)	(5,2)	(1,6)
(2,2)	(3,3)	(4,5)	(5,3)	(5,1)	(6,1)
(1,2)	(3,2)	(3,4)	(4,3)	(4,1)	(5,6)
(1,3)	(2,5)	(4,6)	(4,4)	(4,2)	(6,5)
(1,4)	(2,6)	(5,5)	(5,4)	(1,1)	(6,6)

(b) Scan coordinates

	C1	C2	C3	C4	C5	C6
R1	1	2	3	4	5	6
R2	7	8	9	10	11	12
R3	13	14	15	16	17	18
R4	19	20	21	22	23	24
R5	25	26	27	28	29	30
R6	31	32	33	34	35	36

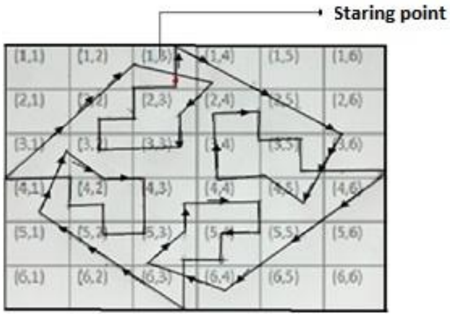
(c) Sample input matrix

	C1	C2	C3	C4	C5	C6
R1	13	10	18	34	32	5
R2	7	9	17	33	26	6
R3	8	15	23	27	25	31
R4	2	14	16	21	19	30
R5	3	11	24	22	20	35
R6	4	12	29	28	1	36

(d) Permuted matrix

Figure 3. Proposed pixel permutation with starting point (3,1)

The scan pattern generated using the same rangoli with starting point (1, 3) and the ending point (3,3) is shown in Fig. 4(a). The scan coordinates obtained by using this scan pattern is shown in Fig. 4(b).The sample input matrix is shown in Fig. 4(c). The permuted matrix obtained by using the scan pattern is shown in Fig. 4(d). From the results it is seen that thepermuted matrices shown in Fig 3(d) and Fig 4(d) are significantly different for an acceptable level. Thus, the same rangoli can be utilized to generate different scan patterns to achieve permutation.



(1,3)	(4,5)	(6,3)	(5,2)	(2,1)	(1,5)
(1,4)	(3,4)	(5,3)	(5,1)	2,2)	(6,1)
(2,5)	(2,4)	(4,3)	(4,1)	(1,2)	(1,6)
(2,6)	(4,6)	(4,4)	(3,1)	(2,3)	(6,5)
(3,6)	(5,5)	(5,4)	(3,2)	(3,3)	(5,6)
(3,5)	(6,4)	(6,2)	(4,2)	(1,1)	(6,6)

(a) Scan pattern (b) Scan coordinates

	C1	C2	C3	C4	C5	C6
R1	1	2	3	4	5	6
R2	7	8	9	10	11	12
R3	13	14	15	16	17	18
R4	19	20	21	22	23	24
R5	25	26	27	28	29	30
R6	31	32	33	34	35	36

	C1	C2	C3	C4	C5	C6
R1	3	23	33	26	7	5
R2	4	16	27	25	8	31
R3	11	10	21	19	2	6
R4	12	24	22	13	9	35
R5	18	29	28	14	15	30
R6	17	34	32	20	1	36

(c) Sample input matrix (d) Permuted matrix

Figure 4. Proposed pixel permutation with starting point (1,3)

3.4 Bitwise XOR Operation using Random Number -

In this level of encryption, bitwise XOR operation is performed to substitute the intensity value of each individual pixel of the scrambled image. Random numbers are widely used in image encryption methods to change the pixel values [2, 3, 18]. The necessary amounts of random numbers are generated by using the BBS random number generator algorithm.The Blum Blum Shub (BBS) pseudorandom number generator is used to generate the random numbers. The BBS generator produces a sequence of bits according to the following algorithm [20].

```
X0 = S2 mod n
for i = 1 to ∞
Xi = (X i-1)2 mod n
Bi = Xi mod 2
```

Where, S is the seed value and n is the product of two prime numbers (p and q). Both p and q have a remainder of 3 when divided by 4 and S is relatively prime to n.

4. PROPOSED IMAGE ENCRYPTION METHOD

In this section, the proposed image encryption method which employs rangoli based scan patternand random number is described. The overall working model of the proposed image encryption method is shown in Fig. 5.

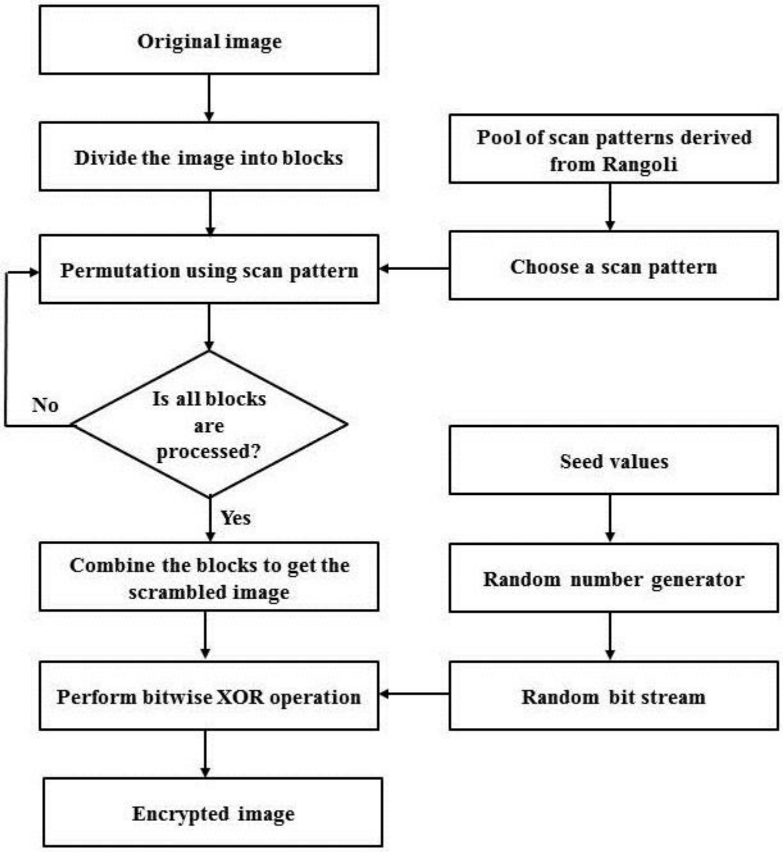

Original image
Divide the image into blocks
Pool of scan patterns derived from Rangoli
Permutation using scan pattern
Choose a scan pattern
No
Is all blocks are processed?
Yes
Seed values
Combine the blocks to get the scrambled image
Random number generator
Perform bitwise XOR operation
Random bit stream
Encrypted image

Figure 5. Working Model of Proposed Image Encryption Method

The original image is given as input to the algorithm and it is scrambled by pixel position permutation with scan pattern derived using rangoli. This image is then XORed with random number to obtain the encrypted image. To decrypt the cipher image, first the sufficient amount of random numbers is generated using the same random number generator with seed values used to the encryption side. Further, the cipher image is XORed with therandom numberto extract the scrambled version of the image. Next, inverse permutation is applied using the scan pattern used during encryption process.

The rangoli, starting and ending coordinates of the rangoli to generate scan patterns, and the seed values of the random number generator are securely shared between the communicating persons.

4.1 Encryption Algorithm

Input: Original image, Scan pattern, seed values
Output: Encrypted image
Step 1: Input the original image and the scan pattern obtained using Rangoli.
Step 2: Input the seed values of random number generator.
Step 3: Divide the image into blocks of size equal to the rangoli.
Step 4: Perform pixel permutation with scan pattern on each block.
Step 5: Repeat the above step until processing all the blocks.
Step 6: Generate random numbers by using the BBS generator.
Step 7: Obtain the cipher image by XORing the scrambled image obtained in step 4 and the random number generated in step 6.
Step8: Store the cipher image.

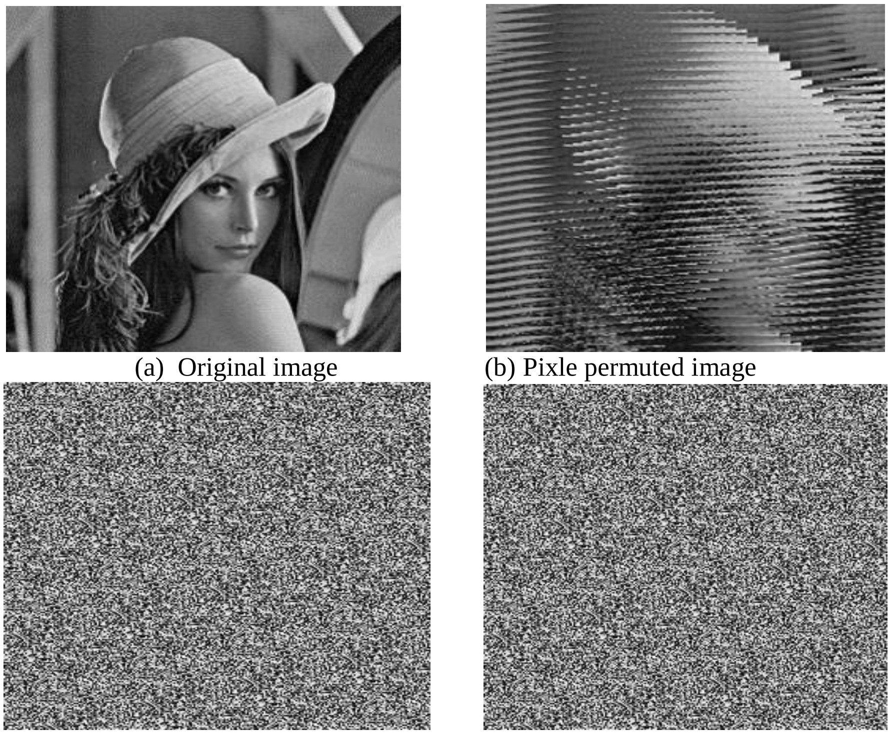
4.2 Decryption Algorithm

Input: Encrypted image, Scan pattern, seed values
Output: Decrypted image
Step 1: Input the cipher image.
Step 2: Input the seed values of random number generator.
Step 3: Generate random numbers by using BBS generator.
Step 4: Perform bitwise XOR operation between the random numbers and encrypted image to get the scrambled image.
Step 5: Select the appropriate scan pattern derived from the rangoli.

Step 6: Divide the image in to blocks.
Step 7: Perform inverse pixel permutation on each block with the scan pattern on thescrambled image to get the decrypted image.
Step 8: Store the decrypted image.

5. EXPERIMENTAL RESULTS AND SECURITY ANALYSIS

The proposed image encryption method was experimented and the evaluation metrics are measured to confirm the effectiveness of the encryption method. The obtained results are presentedwith standard test images of size 256×256 pixels and the results are compared with few existing image encryption methods. The experiment is done in Matlab 2010a with Intel Core i3 Duo Processor, 2 GB RAM, 160 GB Hard Disk Drives, Clock Speed is 2 GHz, and Windows 7 Operating System. The results of proposed image encryption method using rangoli based scan pattern and random number is shown in Figure 6.



(a) Original image (b) Pixle permuted image

(c) Sample random numbers (d)Encrypted image

Figure 6. Resultof proposed image encryption method

Fig. 6(a) is the original Lena image used as input and the corresponding image after pixel permutation is shown in Fig. 6(b). The sample random numbers generated using BBS generator is shown in Fig. 6(c). The encrypted image obtained by XORing the random numbers with the scrambled image is shown in Fig. 6(d).

Number of Pixel Change Rate (NPCR) -

The first measure is the number of pixels change rate (NPCR), which indicates the percentage of different pixels between two images. For the plaintext image Io(i, j) and encrypted image IENC(i, j) the equation (1) gives the mathematical expression to compute the NPCR value [10].

$$NPCR = \frac{\sum_{i,j} D(i,j)}{W*H} * 100\% \qquad (1)$$

Where, W and H are the width and height of the images. If Io(i, j) = IENC(i, j), then D(i,j) = 0 Otherwise, D(i,j)=1.

Unified Average Changing Intensity (UACI) -

A small change in plaintext image must cause some significant change in cipher text image. UACI is helpful to identify the average intensity of difference in pixels between the two images. For the plaintext image Io(i, j) and encrypted image IENC(i, j) the equation (2) gives the mathematical expression to compute the UACI value [10].

$$UACI=\frac{1}{W*H}\left[\sum_{i,j}\frac{Io(i,j)-Ienc(i,j)}{255}\right] * 100\% \qquad (2)$$

Where, W and H are the width and height of the images. Table 1 gives NPCR and UACI values of images encrypted using Rangoli based scan pattern. The same values for the existing image encryption methods are shown in Table 2. The result obtained by the proposed method is acceptable and comparable with the existing methods.

Table - 1 NPCR and UACI values of Proposed Method

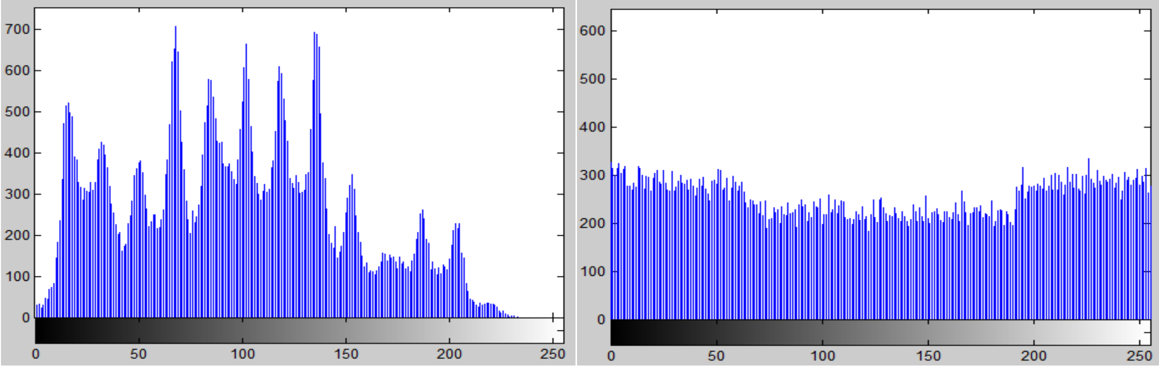
	NPCR (in %)	UACI (in %)
Lena	99.6200	30.8812
Baboon	99.6733	30.0713
Cameraman	99.3544	30.2466

Table - 2 NPCR and UACI values of Existing Methods

Existing method	NPCR (in %)	UACI (in %)
Diaconu & Loukhaoukha [3]	99.6120	30.5997
Huang et al [6]	99.5400	28.2700
Panduranga & Naveenkumar [9]	99.6185	32.0690
Loukhaoukha et al [11]	99.5850	28.6210
Vidhya Saraswathi et al [12]	99.8500	33.5800
Sivakumar et al [18]	99.6460	31.3903

5.1 Histogram Analysis

The histogram analysis clarifies that, how the pixel values of image are distributed. The histogram of original image contains great rises followed by sharp declines and the histogram of encrypted image has uniform distribution which is different from the original image. Figure 7 shows the histogram of the original and the corresponding encrypted images.



(a) Original Lena image (b) Encrypted Lena image

Figure 7.Histogram of Lena image

5.2 Correlation coefficient

Correlation computes the degree of similarity between two variables. This parameter is useful for measuring the effectiveness of the cryptosystem. An arbitrarily chosen pixel in an image is generally strongly correlated with adjacent pixels, and it's in horizontal, vertical or diagonal directions. A secure image encryption algorithm must produce an encrypted image having low correlation between adjacent pixels.The correlation coefficient is calculated by using the Equations (3) to (6).

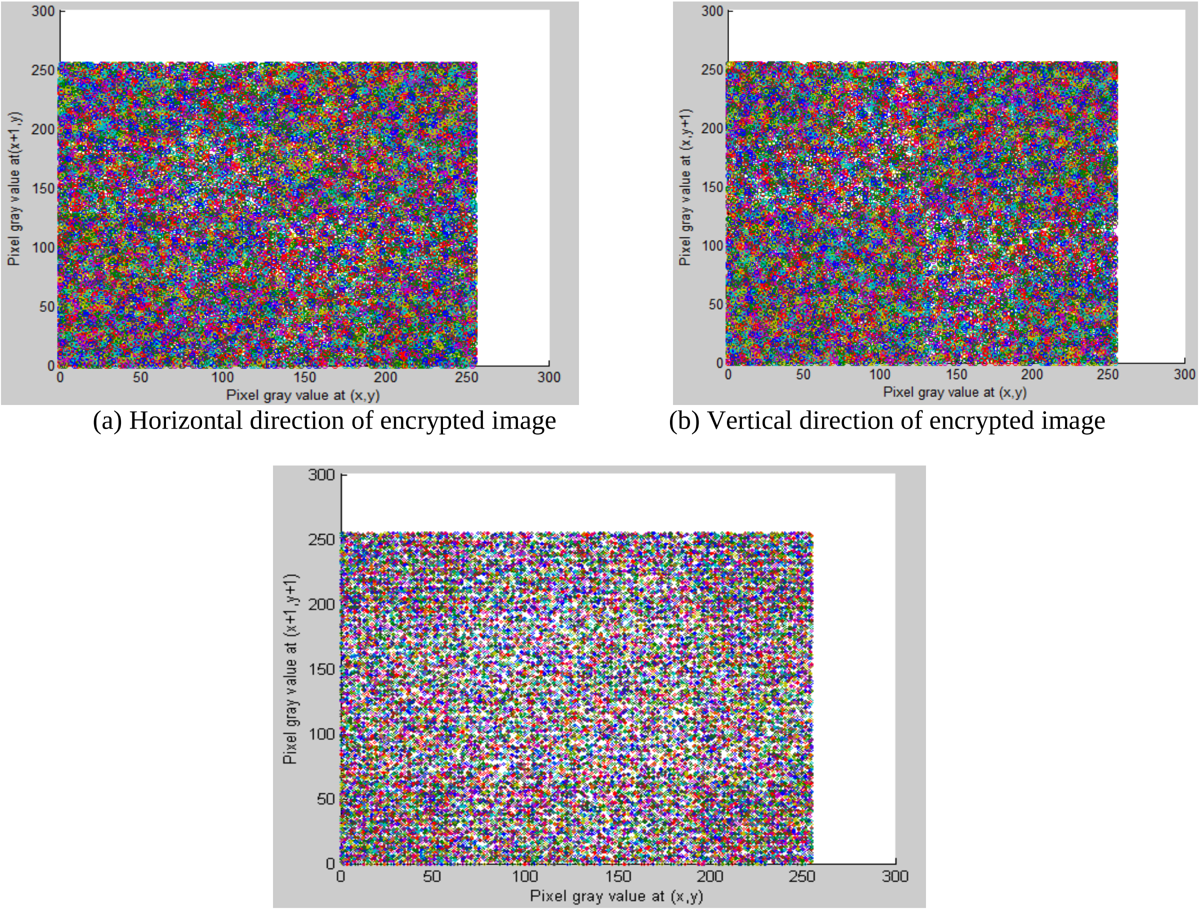
$$\gamma_{xy} = \frac{Cov(x,y)}{\sqrt{D(x)}\sqrt{D(y)}} \quad (3)$$

$$Cov(x,y)=\frac{1}{N}\sum_{i=1}^{N}(xi - E(x))(yi - E(y)) \quad (4)$$

$$D(x) = \frac{1}{N}\sum_{i=1}^{N}(xi - E(x))^{2} \quad (5)$$

$$E(x) = \frac{1}{N}\sum_{i=1}^{N}x_{i} \quad (6)$$

Where, Cov(x,y) is the covariance of x and y; x,y are values of adjacent pixels in image, N is the number of pixel pairs (xi, yi), and E(x) and E(y), are the mean values of xi and yi respectively.The graphical view of the correlation between the adjacent pixels in the encryption image is shown in Figure 8(a) to 8(c).



(a) Horizontal direction of encrypted image

(b) Vertical direction of encrypted image

(c) Diagonal direction of encrypted image

Figure 8. Adjacent Pixel Correlation distribution

5.3 Information Entropy

The entropy of a message source is a measure of the amount of information the source has. The entropy of gray-scale images is theoretically equal to 8 Sh, if each level of gray is assumed to be equiprobable[1, 5]. If the entropy values of the encrypted images are close to the ideal value of 8 Sh, then the encryption algorithm is highly robust against entropy attacks. The entropy of the information is computed by using the Equation (7).

$$H(m) = \sum_{i=0}^{m-1} p(mi) \log \left({}^{1}/_{p(mi)}\right) \qquad (7)$$

Where, m is the total number of symbols in mi∈ m; p(mi) represents the probability of occurrence of the symbol mi and log denotes the base 2 logarithm. The obtained entropy value of the proposed method and few existing methods are given in Table 3. It is observed that the result obtained by the proposed method is acceptable and comparable with the existing methods.

Table – 3Comparison of Entropy Value

Encryption Method	Entropy Value (Sh)
Proposed Method	7.9662
Adrian Viorel Diaconu et. al. [3]	7.9992
G.A. Sathishkumar et. al. [8]	7.8101
Khaled Loukhaoukha et. al. [11]	7.9968
Sivakumar et al [18]	7.9970
Z Lin and H wang [21]	7.9890

6. CONCLUSION

In this paper an image encryption technique using rangoli based scan pattern and random number is developed. The pixels position of the original image is permuted with rangoli based scan pattern. The cipher image is obtained by XORing the scrambled image with the random numbers.The obtained result is analyzed and compared with the existing image encryption methods. The proposed method is simple and easy to implement.

7. REFERENCES

[1] A.J Menezes, P.C Van Oorschot & S.A Vanstone, “Handbook of Applied Cryptography”, CRC Press, New York, 2010.

[2] A.K. Banthia, Namita Tiwari, “Image Encryption using Pseudo Random Number Generators”, International Journal of Computer Applications, Vol. 67, No. 20, pp. 1-8, 2013.

[3] Adrian Viorel Diaconu and Khaled Loukhaoukha, “An improved secure image encryption algorithm based on Rubik’s cube principle and digital chaotic cipher,” Mathematical Problems in Engineering, Vol. 2013, pp. 1-10, 2013.

[4] Avi Dixit, Pratik Dhruve and Dahale Bhagwan “Image encryption using permutation and rotational XOR technique,” Computer Science & Information Technology, Vol. 2, No. 3, pp. 01-09, 2012.

[5] C.E Shannon, “A mathematical theory of communication”, Bell Systems Technical Journal, Vol. 27, No. 3, pp. 379-423, 1948.

[6] C.K Huang, C.W Liao, S.L Hsu & Y.C Jeng, “Implementation of gray image encryption with pixel shuffling and gray-level encryption by single chaotic system”, Telecommunication Systems, Vol. 52, pp. 563–571, 2013.

[7] C.K. Huang and H.H. Nien, “Multi chaotic systems based pixel shuffle for image encryption,” Optics Communications, vol. 282, no. 11, pp. 2123-2127, 2009.

[8] G.A. Sathishkumar and K Bhoopathy Bagan, “A novel image encryption algorithm using pixel shuffling and Base-64 encoding based chaotic block cipher,” WSEAS Transactions on Computers, Vol. 10, No. 6, pp. 169-178, 2011.

[9] H.T. Panduranga and SK Naveen Kumar, “Hybrid approach for image encryption using scan patterns and carrier images,” International Journal on Computer Science and Engineering, Vol. 2, No. 2, pp. 297-300, 2010.

[10] Jawad Ahmad and Fawad Ahmed, “Efficiency analysis and security evaluation of image encryption schemes”, International Journal of Video & Image Processing and Network Security, Vol. 12, No. 04, 2012, pp. 18-31.

[11] 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. 2012, pp. 1-13, 2012.

[12] P. Vidhya Saraswathi and 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, 2012, pp. 1541-1546.

[13] Pareek Narendra, Vinod Patidar and Krishan K Sud, “Image encryption using chaotic logistic map”, Image and vision computing, Vol. 24, No. 9, pp. 926-934, 2006.

[14] S. Naranan, “Kolam Designs Based On Fibonacci Numbers – Part 1&2”, August 2007.

[15] S. Naranan, “Kolam Designs Based On Fibonacci Numbers – Part 3”, November 2015.

[16] S.S Maniccam and N.G Bourbakis, “Image and video encryption using scan patterns,” Pattern Recognition Society, Vol. 37, No. 4, pp. 725-737, 2004.

[17] S.S Maniccam and N.G Bourbakis, “Lossless image compression and encryption using scan,” Pattern Recognition, Vol. 34, No. 6, pp. 1229-1245, 2001.

[18] T. Sivakumarand R. Venkatesan, “A Novel Image Encryption Using Calligraphy Based Scan Method and Random Number”, KSII Transactions on Internet and Information Systems, Vol. 09, No. 06, pp. 2317 - 2337, 2015.

[19] Tang, Yang, Zidong Wang, and Jian-an Fang. "Image encryption using chaotic coupled map lattices with time-varying delays”, Communications in Nonlinear Science and Numerical Simulation, Vol. 15, No. 9, pp. 2456-2468, 2010.

[20] William Stalling, “Cryptography and Network Security – Principles and Practices”, Pearson Education, New Delhi, 2013.

[21] Z. Lin and H. wang, “Efficient image encryption using a chaos-based PWL memristor”, IETE Technical Review, Vol. 27, No. 4, pp.318–325, 2010.