An Enhanced Image Encryption Scheme using 16 Byte Key

Shradha Suryawanshi[*1], Pankaj Kumar Sahu[*2]

[*1]Gargi Institute of Science & Technology, Bhopal, India

[*2]Barkatullah University, Bhopal, India

Abstract— *Nowadays, security and privacy has become one of the primary issues in data storage and transmission. In the age of multimedia, images are widely used in business, advertising and promotions. Therefore the protection of images from unauthorized access is important. Image encryption plays a significant role in the field of information hiding. In this article, we introduce a chaos based image encryption technique using method of shuffled operations. This technique uses 128-bit private key for encryption. Performance of proposed image encryption scheme is compared with the existing inter pixel displacement image encryption scheme. Performance evaluation reveals that proposed scheme performs better than existing scheme in terms of entropy of encrypted image.*

Keywords — *Image encryption, cipher, security.*

I. INTRODUCTION

Maintaining privacy in our personal communications is something everyone desires. Cryptography is a means to achieve that privacy. In layman terms, cryptography or encryption is a process of converting readable or understandable information to a form that cannot be understood or read. The modified form of information, which cannot be read or understood, is known as ciphered information or encrypted data. The process of recovering back the encrypted information is called decryption.

A secret key is long sequence of bits used by cryptography algorithms. During encryption, the algorithm alters original data based on the key's bits to create a new encrypted message. An image is an array, or a matrix, of square pixels (picture elements) arranged in columns and rows [20]. In an 8-bit grayscale image, each picture element is assigned an intensity that ranges from 0 to 255. A gray scale image is normally called a black and white image but the name emphasizes that such an image will also include many shades of gray.

Image encryption ensures protection of images from unauthorized access. Today it is used in almost every field where images are required to be shared in secret form. Government uses it for secure handling of intelligence data. Similarly in medical field, X-ray images containing vital information of patients are sent in encrypted form.

Remainder of the paper is organized as follow: Related work is discussed in section-2. Proposed architecture is described in section-3, proposed encryption and decryption process are illustrated in section-4 and section-5 respectively. Experimental results are discussed in section-6 and section-7 concludes the paper

II. RELATED WORK

Jolly Shah et. al [5] classified various image encryption schemes and analyzed them with respect to various parameters like tune ability, visual degradation, compression friendliness, format compliance, encryption ratio, speed, and cryptographic security. In [6], authors modified an existing algorithm proposed in which RGB attributes of a pixel were randomly scattered across the image. In [8], a scheme was proposed that employs one of the three dynamic chaotic systems (Lorenz or Chen or LU chaotic system selected based on 16-byte key) that shuffles the position of image pixels and uses another one of the same three chaotic maps for confusing the relationship between the cipher image and the plain-image thereby significantly increasing resistance to attacks. Karthigai Kumar et. al [3] enhanced the robustness of image encryption by applying only 2 rounds of iterations. They also analyzed chaos based technique using the method of parameter modulation, which reduced the problem of dynamical degradation. Amnesh Goel et. al [1] presented an image encryption presented an explosive block displacement followed by inter-pixel displacement of RGB values that is helpful for end to end secure transmission of digital information. In [7] Dongming Chen et. al used logistic maps with 80-bit secret key and two chaotic logistic maps. Manjunath Prasad et. al [4] presented a pixel scrambling which is based on chaos based algorithm where the randomness in chaos is used to scramble the position of data. The position of data is scrambled according to randomness of elements obtained from chaotic map. The original image is retrieved by rearranging them back to their original position during decryption.

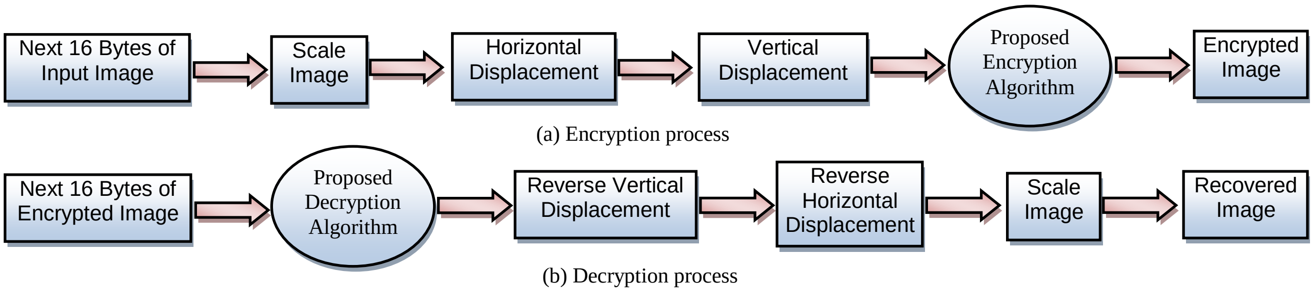
Next section presents the architecture of proposed block displacement based image encryption technique that performs horizontal and vertical block displacement in original image. The proposed encryption process performs bit shifting and ex-or operation on RGB values with a 128 bit secret key.

\

III. PROPOSED ARCHITECTURE

Fig.1(a) and Fig.1(b) depict the architecture of proposed encryption and decryption architecture respectively. In encryption process, the input image is firstly divided into four equal sized quadrants. Each quadrant is considered as separate image, which undergoes same set of operations. These original quadrant images are scaled so that the number of horizontal and vertical pixels in input image becomes a multiple of block length (which is equal in length and height).

After this, pixels are interchanged which leads to horizontal and vertical displacement of pixels. The resulting image undergoes encryption process where in every successive iteration; next 128-bits are encrypted using the 128-bit secret key. This encryption process continues until all the bits of images are encrypted and at last, the encrypted image is obtained after joining all the quadrants of input image. Fig.1(b) depicts proposed decryption process where input image is divided into equal sized quadrants. Each quadrant is considered as separate image and undergoes all the operations individually but in reverse order.


Next 16 Bytes of Input Image
Scale Image
Horizontal Displacement
Vertical Displacement
Proposed Encryption Algorithm
Encrypted Image
Next 16 Bytes of Encrypted Image
Proposed Decryption Algorithm
Reverse Vertical Displacement
Reverse Horizontal Displacement
Scale Image
Recovered Image

(a) Encryption process

(b) Decryption process

Fig.1. Architecture of proposed scheme

The horizontal and vertical displacements are defined in *Horizontal Displacement* and *Vertical Displacement* methods, which identifies the number of horizontal and vertical blocks in an image and accordingly displace the blocks in horizontal and vertical directions. In *Horizontal Displacement* method, the odd blocks are interchanged with previous horizontal block otherwise it is interchanged with next horizontal block in the image. This displacement of blocks is exchanged which further means that there will be no loss of data in displacement. Number of pixels in a block will be determining by BSize, rows and cols are the number of rows and number of columns in the image.

The *Vertical Displacement* method defines the vertical displacement of image blocks. This method works just like the *Horizontal Displacement* method except it moves the blocks in vertical direction. Inverse vertical and horizontal displacements are carried out during decryption where odd blocks are interchanged with next horizontal/vertical block otherwise it is interchanged with previous horizontal/vertical block in the image.

HorizontalDisplacement (image, block_size)

```
1. Img=image
2. BSize= block_size
3. No. of Horizontal Blocks= width of img / BSize
4. No. of Vertical Blocks= height of img / BSize
5. for i=1 to No. of Vertical Blocks
for j=1 to No. of Horizontal Blocks
if i is odd then
Interchange current Block with last horizontal block
if i is even then
Interchange current Block with next horizontal block
End For
EndFor
End
```

Vertical Displacement (image, block_size)

```
1. Img=image
2. BSize= block_size
3. No. of Horizontal Blocks= width of img / BSize
4. No. of Vertical Blocks= height of img / BSize
5. for i=1 to No. of Vertical Blocks
for j=1 to No. of Horizontal Blocks
if i is odd then
Interchange current Block with last horizontal block
if i is even then
Interchange current Block with next horizontal block
End For
EndFor
End
```

IV. ENCRYPTION ALGORITHM

Fig.2 shows the block diagram of proposed encryption process, which consists of two modules i.e. pre-processing of key and the encryption of pixels using the key. Firstly, the pre-processing of 128 bit key (K) is done. After which, K is divided into four equal parts K1, K2, K3 and K4, each of which represents 32 bits. Now each Ki is again split into four parts Ki_1, Ki_2, Ki_3 and Ki_4, each of which represents 8 bits. Once this pre-processing of key is completed the encryption of image takes place as follow:

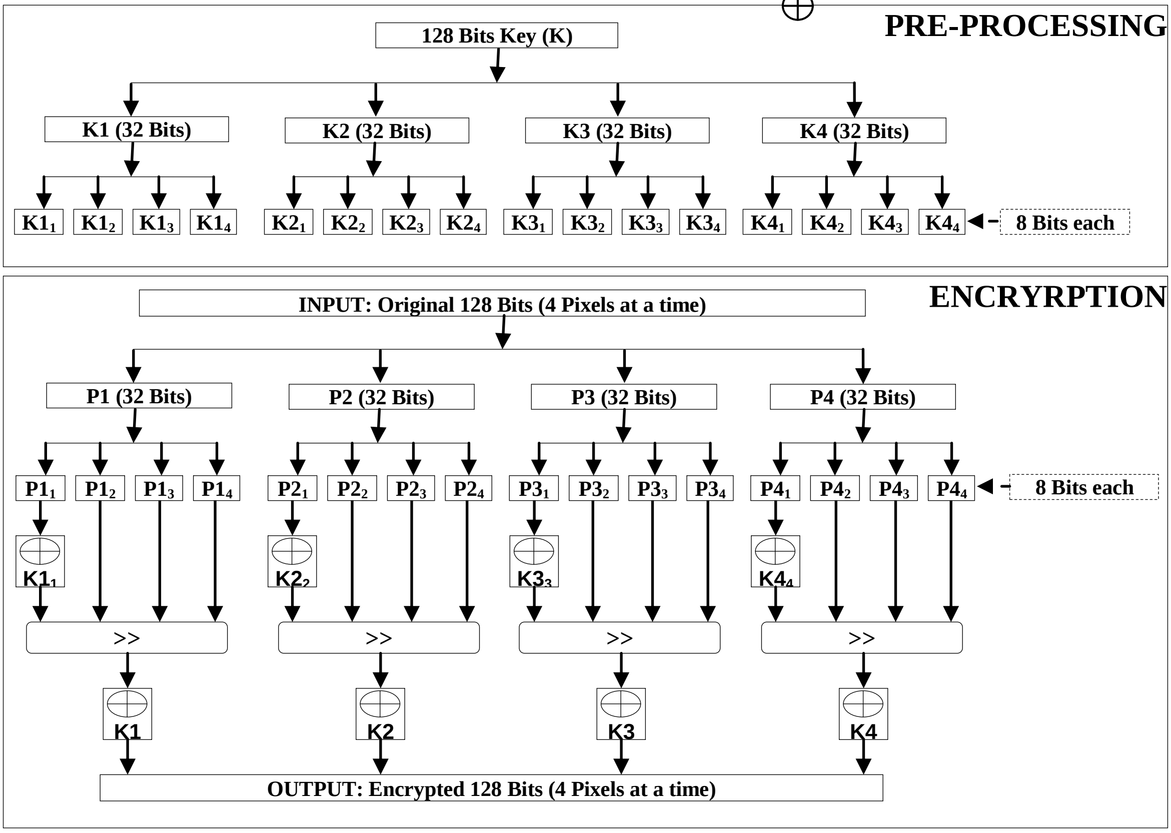


Fig.2. Proposed Encryption Process

1. Read and pre-process the 128-bit key
2. Start reading 128-bits of image at a time
3. Break the 128 bits into four equal parts P1, P2, P3 and P4, each of which represents 32 bits
4. Now break each Pi into four equal parts Pi1, Pi2, Pi3 and Pi4, each of which represents 8 bits
5. Perform EX-OR operation between P11 and K11, and store the result in P11 as shown below:

   $P1_1 = P1_1 \oplus K1_1$

   Similarly update $P2_1$, $P3_1$ and $P4_1$ as follow:

   $P2_1 = P2_1 \oplus K2_2$

   $P3_1 = P3_1 \quad K2_3$

   $P4_1 = P4_1 \oplus K3_4$

6. Apply 2 bits right shift on resultant P1, P2, P3 and P4
7. Now perform EX-OR operation between 32 bit K1 and 32 bit P1, and store the result in P1 as follow:

   $P1 = P1 \oplus K1$

   Similarly update P2, P3 and P4 as follow:

   $P2 = P2 \oplus K2$

   $P3 = P3 \oplus K3$

   $P4 = P4 \oplus K4$

8. Reform the 128 bits from P11,P12, P13 and P14, which represent the required encrypted values of original 128 bits
9. Repeat Step.2 to Step.8 until all the pixels of input image are encrypted

V. DECRYPTION ALGORITHM

As the block diagram in Fig.3 shows, decryption algorithm also first performs pre-processing of 128 bit key (K), which is given by user. In decryption process,

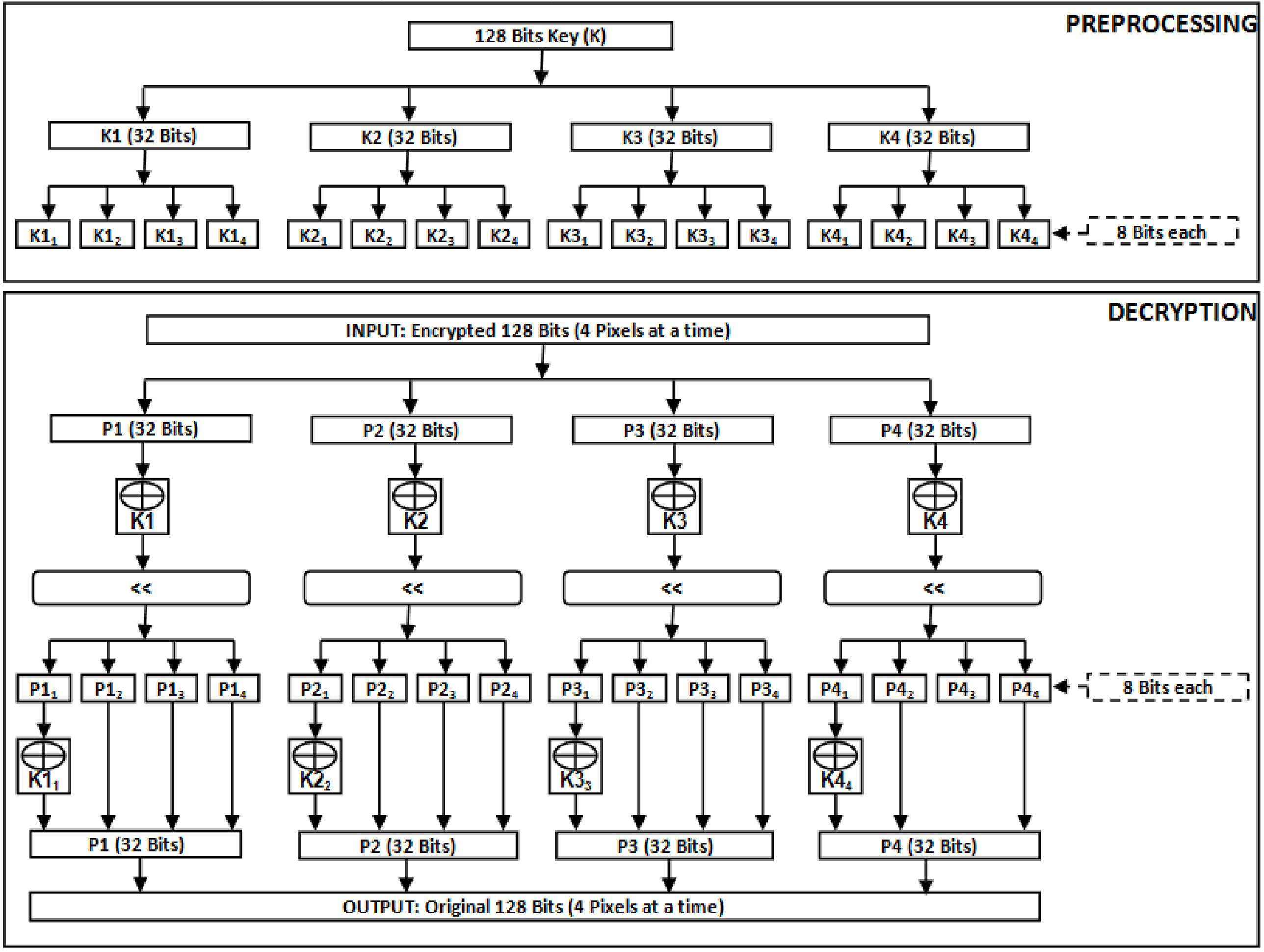
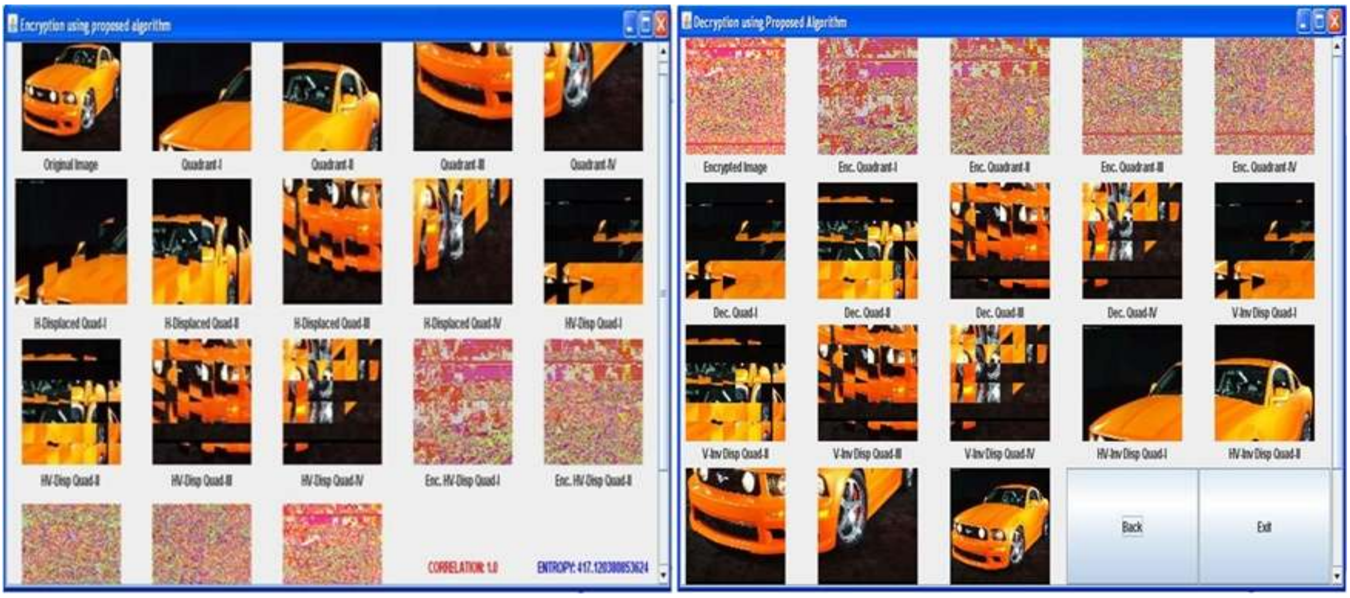


Fig.3. Proposed Decryption



(a) Encryption of original image (b) Decryption of encrypted image

Fig.4. Manipulation of original image during encryption and decryption process

same operations performed in encryption are carried out in reverse order. Once the pre-processing of key is completed, decryption of image takes place as follow:

1. Read and pre-process the 128-bit key
2. Start reading 128-bits of encrypted image at a time
3. Break the 128 bits into four equal parts P1, P2, P3 and P4, each of which represents 32 bits
4. Now perform *EX-OR* operation between 32 bit K1 and 32 bit P1, and store the result in P1 as follow:
   $P1 = P1 \oplus K1$
   Similarly update P2, P3 and P4 as follow:
   $P2 = P2 \oplus K2$
   $P3 = P3 \oplus K3$
   $P4 = P4 \oplus K4$
5. Apply 2 bits left shift on 32 bits of P1, P2, P3 and P4
6. Now break each Pi into four equal parts Pi_1, Pi_2, Pi_3 and Pi_4, each of which represents 8 bits
7. Perform *EX-OR* operation between $P1_1$ and $K1_1$, and store the result in $P1_1$ as shown below:
   $P1_1 = P1_1 \oplus K1_1$
   Similarly update $P2_1$, $P3_1$ and $P4_1$ as follow:
   $P2_1 = P2_1 \oplus K2_2$
   $P3_1 = P3_1 \oplus K2_3$
   $P4_1 = P4_1 \oplus K3_4$
8. Reform the 32 bits of P1 from $P1_1$, $P1_2$, $P1_3$ and $P1_4$.
   Reform the 32 bits of P2 from $P2_1$, $P2_2$, $P2_3$ and $P2_4$
   Reform the 32 bits of P3 from $P3_1$, $P3_2$, $P3_3$ and $P3_4$
   Reform the 32 bits of P4 from $P4_1$, $P2_2$, $P4_3$ and $P4_4$
   Now P1, P2, P3 and P4 are the required encrypted values of original four pixels
9. Repeat Step.2 to Step.9 until all the pixels of input image are decrypted

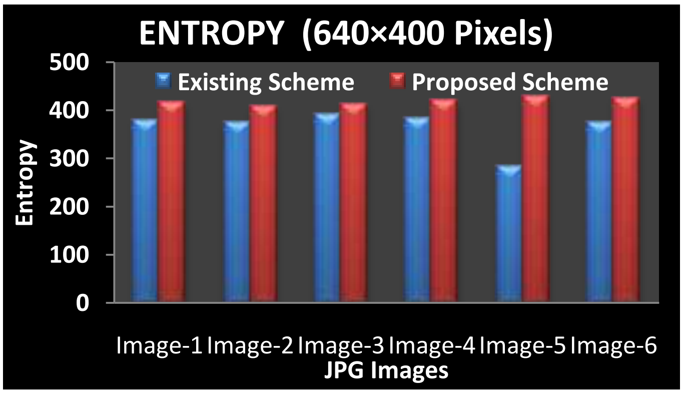


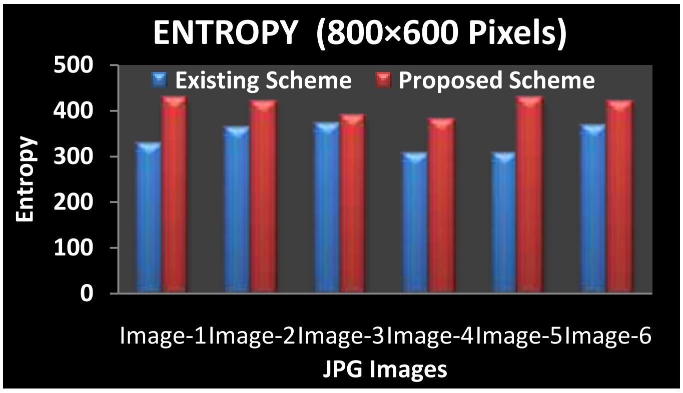
Fig.4 (a) and (b) depicts manipulation of input image during encryption and decryption process.

VI. Results and Comparison

Entropy of an image can be defined as the statistical measure of randomness that can be used to characterize the texture of an image. Higher entropy of an image represents more randomness, which is good for encrypted images. For performance evaluation, the proposed scheme is implemented using JDK1.7 and Java Advanced Imaging API. The entropy of encrypted images using proposed scheme is compared with the entropy of image encrypted using the inter pixel displacement and block displacement scheme presented by Amnesh Goel et. al in [1].

The performance of proposed scheme is evaluated with three different data sets: first data set contains images of size 640×400 pixels, second contains images of size 800×600 pixels and the third data set contains images of arbitrary size. All the data set contains images with different contrast, brightness. Some images are coloured whereas others are gray scaled.

Fig.5 (a), (b) and (c) show the comparison between resulting entropy of images encrypted with proposed scheme and the existing scheme. Fig.5(a) illustrates the entropy of images with size 640×400 Pixels. Similarly Fig.5(b) and Fig.5(c) illustrate the entropy of images with size 800×600 Pixels and arbitrary size respectively. These graphs reflect that when an image is encrypted using proposed encryption scheme then the entropy of encrypted image is always higher than the same image encrypted using the existing scheme. This reveals that the proposed encryption scheme works better than the existing scheme in terms of randomness, regardless of image size, contrast, brightness and colour.



(a) When size of original images is 640×400 pixels

(b) When size of original images is 800×600 pixels

(c) When size of original images is arbitrary

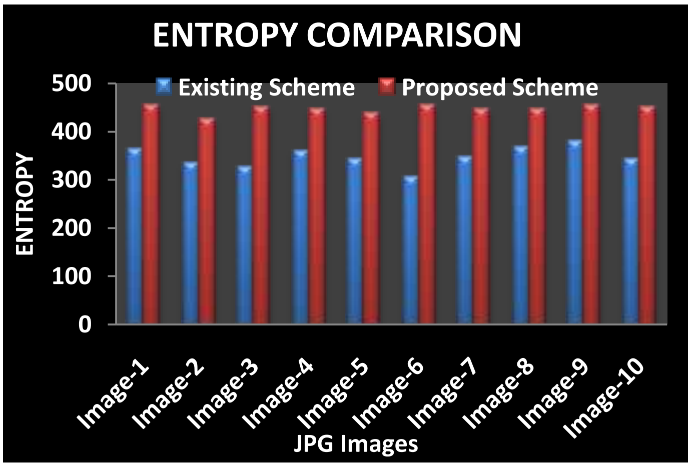


Fig.5. Entropy of images encrypted with proposed scheme against existing scheme

The performance of proposed scheme is evaluated with three different data sets: first data set contains images of size *640×400* pixels, second contains images of size *800×600* pixels and the third data set contains images of arbitrary size. All the data set contains images with different contrast, brightness. Some images are coloured whereas others are gray scaled.

Fig.5 (a), (b) and (c) show the comparison between resulting entropy of images encrypted with proposed scheme and the existing scheme. Fig.5 (a) illustrates the entropy of images with size 640×400 Pixels. Similarly Fig.5 (b) and Fig.5(c) illustrate the entropy of images with size 800×600 Pixels and arbitrary size respectively. These graphs reflect that when an image is encrypted using proposed encryption scheme then the entropy of encrypted image is always higher than the same image encrypted using the existing scheme. This reveals that the proposed encryption scheme works better than the existing scheme in terms of randomness, regardless of image size, contrast, brightness and colour.

VII. Conclusion

In this article, a block displacement based image encryption scheme using 128-Bit secret key was presented. The performance of proposed scheme was compared with the existing scheme in terms of entropy of resulting encrypted image. Results reveal that the proposed image encryption scheme introduces more arbitrariness as compared to existing scheme irrespective of dimensions, contrast, brightness and colour of the input image.

References

[1] Amnesh Goel and Nidhi Chandra "A Technique for Image Encryption Based On Explosive n*n Block Displacement Followed By Inter-Pixel Displacement of RGB Attribute of A Pixel" 2012 IEEE International Conference on Communication Systems and Network Technologies

[2] P.Karthigaikumar, Soumiya Rasheed "Simulation of Image Encryption using AES Algorithm" *IJCA Special Issue on "Computational Science - New Dimensions & Perspectives" NCCSE, 2011*

[3] D. Chattopadhyay, M. K. Mandal and D. Nandi Robust Chaotic Image Encryption based on Perturbation Technique published in ICGST-GVIP Journal, Volume 11, Issue 2, April 2011.

[4] Manjunath Prasad1 and K.L.Sudha2and named "Chaos Image Encryption using Pixel shuffling " published in D.C. Wyld, et al. (Eds): CCSEA 2011, CS & IT 02, pp. 169–179, 2011.

[5] Jolly Shah and Dr. Vikas Saxena "Performance Study on Image Encryption Schemes" published in IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 4, No 1, July 2011 ISSN (Online): 1694-0814.

[6] Reji Mathews, Amnesh Goel and Nidhi Chandra "Image Encryption based on Inter Pixel Displacement of RGB Values inside Custom Slices" International Journal of Computer Applications (0975 – 8887) Volume 36– No.3, December 2011.

[7] Dongming Chen, and Yunpeng Chang "A Novel Image Encryption Algorithm based on Logistic Maps" Advances in Information Sciences and Service Sciences, Volume3, Number7, August 2011.

[8] K.Sakthidasan and B.V.Santhosh Krishna"New Chaotic Algorithm for Image Encryption and Decryption of Digital Color Images" International Journal of Information and Education Technology, Vol. 1, No. 2, June 2011.

[9] Kamlesh Gupta1, Sanjay Silakari2 "Choase Based Image Encryption Using Block-Based Transformation Algorithm"(IJCNS) International Journal of Computer and Network Security,Vol. 1, No. 3, December 2009.

[10] Mohammad Ali Bani Younes and Aman Jantan "An Image Encryption Approach Using a Combination of Permutation Technique Followed by Encryption" IJCSNS International Journal of Computer Science and Network Security, VOL.8 No.4, April 2008.

[11] Mohammad Ali Bani Younes and Aman Jantan "Image Encryption using Block-Based Transformation Algorithm" IAENG International Journal of Computer Science, 35:1, IJCS_35_1_03 Advance online publication: 19 February 2008.