A Reversible Data Hiding Scheme to Embed High Capacity Data in Two Dimensional Difference Histogram Modification

Shivani Sharma[1], Kulwinder Singh[2]
Department of Electronics and Communication Engineering, Punjabi University, Patiala, India
(E-mail: er.shivani91@gmail.com)

Abstract— In this paper, a modified reversible data hiding scheme is proposed which is based on two dimensional difference histogram modification. In this scheme, difference-pair-mapping is used. A sequence of pairs of difference values is computed by taking each pixel-pair and its context. Then, a two-dimensional difference-histogram is generated by counting the frequency of the resulting difference pairs. Then, a reversible data hiding is implemented according to a particularly designed DPM. We have presented a method of data hiding in which we have increased the average embedding capacity of the Li *et al.* method. We have also modified the algorithm to consider it working for textual data as well. Experimental results have been compared with Li *et al.* method and found considerable increase in embedding capacity of the algorithm.

Keywords— *Difference pair mapping (DPM), embedding capacity (E.C.), reversible data hiding (RDH) and two-dimensional-difference-histogram.*

I. INTRODUCTION

Data hiding is defined as a process to hide data into cover image. In most cases of data hiding, the cover image will experience slight distortion due to data hiding and the original image cannot be recovered. Some permanent distortion has happened to the cover image even after the embedded image has been extracted out [1]. Any existing digital media, like, audios, videos, and digital images can be used as carriers. The digital image is used as a carrier since it is mostly delivered over the internet. The image for carrying data is called a cover image and the image carrying embedded information is called a stego image. When the data is embedded into images, the pixel values of an image are changed, and thus the quality of image is degraded. Since, the modified pixels cannot be recovered into their original state after the secret messages has been extracted out, permanent distortion will occur. Distortion for some applications is unacceptable e.g., a distorted arm X-ray image could result in an incorrect medical diagnosis. In these applications , technique for reversible data hiding are necessary[2].The aim of reversible data hiding is to embed secret message into a cover image by changing its pixel value and at the end, cover image as well as embedded message should be completely recovered.[3].

This paper presents system for embedding text data using modified reversible data hiding in MATLAB. The paper is organized as follows: The recent work is presented in section II. The proposed Modified Reversible data hiding is presented in Section III. The results are presented in section IV. Finally, the conclusion is given in section V.

II. RELATED WORKS

Recently, some reversible marking techniques have been reported in literature. The first histogram-based RDH method is the one proposed by Ni *et al.* in [4]. In this method, they used peak and minimum points of the pixel-intensity-histogram to embed data. Each pixel value is changed at most by 1, and thus a good marked image quality can be obtained. Embedding capacity of this method is quite low and it does not work well if the cover image has a flat histogram. Tai *et al.* [5], proposed efficient extension of the histogram modification technique by considering differences between adjacent pixels rather than simple pixel value. They used binary tree structure to eliminate the requirement to communicate pairs of peak and zero points to the recipient and also used histogram shifting technique to prevent overflow and underflow. Since image neighbor pixels are strongly correlated so the difference was expected to be very close to zero. Hence, by using this method, they achieved large hiding capacity while keeping embedding distortion low. Zhincheng Ni *et al.* [6] proposed reversible data hiding technique which was able to embed about 5-80 kb into a 512×512×8 grayscale image and PSNR of marked image versus original image to be above 48dB. In this method, they utilized the zero or the minimum points of the histogram of an image and slightly modified the pixel grayscale values to embed data into an image. Even, the computational complexity of their proposed technique was low because in this method, they didn't use discrete cosine transform, discrete wavelet transform and fast fourier transform. All the processing was in spatial domain. Hence, the execution time was short. Many applications need high quality images, such as medical or military images. The well known reversible data hiding method proposed by Ni *et al.* can produce high quality stego image (less than or equal to 48.13 dB), but the embedding capacity is low and is limited by the distribution of image histogram. Wein Hong *et al.* [3] proposed a novel reversible data hiding technique based on modification of prediction errors. In this scheme, pixel values are first predicted, and then error values are obtained. The PSNR of the stego image produced by MPE is guaranteed to be above 48 dB. Thodi *et al.* [8] described two new reversible watermarking algorithms, which combines histogram shifting and difference expansion technique. And second one using flag bits. Then, a new reversible data embedding technique called prediction-error expansion was then introduced and watermarking. Algorithms based on the prediction-error expansion technique were presented.

Wein Hong [9], proposed an improved reversible data hiding technique which includes three schemes, dual binary tree (DBT), median edge detection (MED) and expansion embedding capacity (EEC). The embedding performance was good in terms of image quality and payload. Krishna *et al.* [10] proposed an efficient extension of the histogram modification technique by considering the differences between adjacent pixels instead of using simple pixel value. They used binary tree that predetermines the multiple peak points used to embed messages. Their proposed method achieved large hiding capacity while keeping distortion low.

III. MODIFIED RDH SCHEME

There are the two processes which we are discussing below:

A. Embedding Procedure

We now briefly introduce our embedding procedure.

Step 1) Read the text file to be embedded in MATLAB and convert it to ASCII code values and make a binary bit array used for data hiding.

Step 2) Read the image into MATLAB and convert it to gray scale if it is in RGB form. Then divide the test image into non overlapped pixel-pairs.

Step 3) For every pixel-pair (x, y), compute prediction of y to get z using GAP predictor given below:

$$z=\begin{cases} v_1, & if\ d_v-d_h>80 \\ \frac{(v_1+u)}{2}, & if\ d_v-d_h\in(32,80] \\ \frac{(v_1+3u)}{4}, & if\ d_v-d_h\in(8,32] \\ u, & if\ d_v-d_h\in[-8,8] \\ \frac{(v_4+3u)}{4}, & if\ d_v-d_h\in[-32,8) \\ \frac{(v_4+u)}{2}, & if\ d_v-d_h\in[-80,-32) \\ v_4, & if\ d_v-d_h<-80 \end{cases} \quad (1)$$

where, v_1 to v_{10} are neighboring pixels taken from surrounding window described below.

For a pixel-pair (x, y), we propose to compute two difference values difference values d_1= x-y and d_2= y-z to form a two-dimensional-difference-histogram of (d_1, d_2), where z is a prediction of y.

	j	j+1	j+2	j+3
i	x	y	V_1	V_2
i+1	V_3	V_4	V_5	V_6
i+2	V_7	V_8	V_9	V_{10}

Fig.1 Pixel window for calculation of z and threshold value t.

where {v_1,…., v_6, v_7, v_8} are neighboring pixels of (x,y) (see Fig. 1)

d_v= |v_1-v_5| + |v_3-v_7| + |v_4-v_8| and d_h= |v_1-v_2| +| v_3-v_4| + |v_4-v_5| represent the vertical and horizontal gradients and

u = (v_1+v_4) /2 + (v_3- v_5) /4.

Here i represent the row and j represents the column co-ordinate of the test image. In this, z has been rounded to its nearest integer by using ceil function in MATLAB.

Step 4) Calculate the noisy-level by summing both horizontal and vertical pixel differences of each two consecutive pixels in pixel window, and it is less than or equal to 13×255.

Step 5) For each pixel-pair with noisy-level less than T (threshold), compute the difference-pair (d_1, d_2) and implement data embedding according to the DPM defined below:

TABLE I. LOOK UP TABLE FOR DATA EMBEDDING PROCESS

Conditions on d_1 and d_2	Operation in data embedding	Modification move-ment to pixel-pair	Marked value
d_1= 0 & d_2<= -2	Expansion embedding	down	(x, y-b)
d_1= 0 & d_2= -1			
d_1= 1 &d_2= -1			
d_1=>=1&d_2=0			
d_1= -1&d_2<= -2	Expansion embedding	left	(x-b, y)
d_1= -1&d_2= -1			
d_1>= 1&d_2<= -2	Shifting	down	(x, y-3)
d_1>=2&d_2=-1			
d_1<= -2&d_2<=-2	Shifting	left	(x-3, y)
d_1=0&d_2= -1			
d_1= 0 & d_2= 0	Expansion embedding	up	(x, y+b)
d_1= 0 & d_2>= 1			
d_1>=2 & d_2>=1	Shifting	right	(x+3, y)
d_1=1 & d_2>= 1	Expansion embedding	right	(x+b, y)
d_1<= -1&d_2>= 0	shifting	up	(x, y+ 3)

Step 6) For reversible data hiding, some pixels are chosen for embedding process and changed in order to done embedding of bits and some are shifted without any embedding in order to make the algorithm reversible. Table I gives information about shifted and embedded pixels which are chosen according to look up table.

Step 7) Keep all the changed values of pixel pairs in a matrix and save the watermarked image to the folder when embedding has been done on all image.

B. Extraction and image restoration procedure

The extraction process is the reverse of embedding process. We have to extract the textual data which is used in embedding process. As we are working on reversible data hiding, so we have to extract the textual data as well as to restore the cover image.

Following are the steps of extraction process.

Step 1) Load watermarked image in MATLAB workspace.

Step 2) Start extraction process from last pixel-pair as extraction will be done in reverse order of embedding.

Step 3) Evaluate the z (gap predictor) and t (threshold) using (1) and Fig.1 in the embedded algorithm.

Step 4) Find direction variables d_1 and d_2 using formula d_1=x-y and d_2=y-z where x is first pixel of pixel pair and y is second pixel of pixel pair.

Step 5) Apply the extraction of embedded bits and shifted bits according to the DPM described in Table II.

Step 6) Store the extracted bits found in reverse order in an array and also store newly generated values of x and y in a matrix.

Step 7) Store the newly generated matrix in the form of image and compare with the original to check the performance of the algorithm.

Step 8) Store the extracted bits in an array and reverse the order. Finally make alphabets and numbers from it by making eight-bit pair and then conversion from ASCII to alphabet form.

Step 9) Compare the embedded and extracted bits for checking the accuracy and performance of the algorithm.

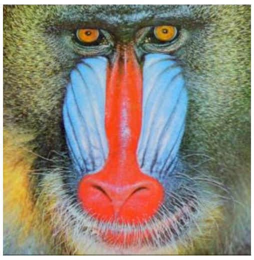
Below is the Look up table for extraction process

TABLE II. LOOK UP TABLE FOR EXTRACTION PROCESS

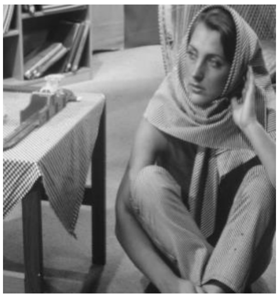
Conditions on d_1 and d_2	Extracted data bit	Recovered value of pixels
(d_1= 3or d_1=2) &(d_2<= -2)	d_1 -1	(x, y+b)
(d_1= 3or d_1=2) & (d_2<= -4) or ((d_1= 0or d_1=1) &(d_2<= -2))	d_1	(x, y+b)
(d_1= -4 or d_1=-3 or d_1=-2 or d_1=-1) & d_2<= -2	1 – d_1	(x+b, y)
(d_1 >=4 & d_2<= -5) or (d_1>= 5 & d_2= -4)	No change	(x, y+3)
d_1<=-5 & d_2<= -2	No change	(x+3, y)
(d_1= 1 & d_2= -1) or (d_1= 2 &(d_2= -2)	d_1-1	(x, y+b)
(d_1= 3& d_2=-4) or (d_1= 2 & d_2=-3) or (d_1= 1& d_2=-1) or (d_1= 4 & d_2=-4) or (d_1=3 & d_2= -3) or (d_1=2 &d_2=-2) or (d_1=1 &d_2=-1)	-1 - d_2	(x, y+b)
d_1= -4 or d_1=-3 or d_1=-2 or d_1=-1) & d_2= -1)	-1 – d_1	(x, y+b)
(d_1= 7& d_2=-2) or (d_1= 8 & d_2=-3) or (d_1>= 1& d_2=-1) or (d_1= 4 & d_2=-0) or (d_2= 0 or d_2= -1 or d_2= -2 or d_2= -3)	-d_2	(x, y+ b)
d_1<=-5 & d_2= -1	No change	(x+3, y)
d_1<=0 & d_1>=-3) & (d_2= 0 or d_2=1) or (d_2= 2 or d_2=3) or (d_1= 0 or d_1=-1) or (d_1=-2 or d_1= -3) &d_2>=1)	-d_1	(x, y-b)
d_1>=5 & d_2>=1	No change	(x- 3,y)
d_1= 1or d_1=2 or d_1=3 or d_1=4)&(d_2>=1or d_2>=1)	d_1 -1	(x-b, y)
d_1<=-4 & d_2>=3	No change	(x, y-3)

IV. RESULTS AND DISCUSSIONS

Four 256×256 sized gray scale images: Airplane (F-16), Baboon, Barbara and Fishing Boat are used in our experiment. Fig. 3 shows the histograms of original images and watermarked images and Fig. 4 shows the performance comparison between the proposed method and Li *et al* method [3]. Tables III, IV, V and VI represent the Peak signal to Noise Ratio (PSNR) and Embedding capacity (E.C.) of proposed method and Li et al. method [3]. Referring to Fig. 4, we can see that our scheme is better than Li et al. scheme [3]. The different threshold values are taken for images.

(a) (b)




(c) (d)

Fig. 2. Test images (a) Airplane (F-16) (b) Baboon (c) Barbara (d) Fishing boat

The results of the test images show that there is a tradeoff between PSNR and embedding capacity. As the threshold value increases, the value of PSNR decreases. This means, at zero thresholds, the value of PSNR is maximum whereas Embedding capacity is minimum and at higher threshold values or maximum threshold value, the embedding capacity is maximum, whereas the value of PSNR decreases. In test image Barbara, at zero threshold value, the value of PSNR is 45.7785 dB and embedding capacity is 1800 and at maximum threshold value, the PSNR is 42.2102 and embedding capacity is 12744.

TABLE III. PEAK SIGNAL TO NOISE RATIO (dB) AND EMBEDDING CAPACITY (bits) FOR AIRPLANE (F-16)

Threshold values	Proposed Method		Li *et al.* [3]	
	PSNR	E.C.	PSNR	E.C.
0	49.5734	3072	59.0259	1768
20	47.4657	5776	56.9179	3376
40	44.6102	17672	54.0065	10328
Max.	43.2970	31200	52.6493	18192

TABLE IV. PEAK SIGNAL TO NOISE RATIO (dB) AND EMBEDDING CAPACITY (bits) FOR BABOON

Threshold values	Proposed Method		Li *et al.* [3]	
	PSNR	E.C.	PSNR	E.C.
0	45.1977	1408	54.7334	872
20	44.6566	1704	54.1905	1048
40	44.1739	2008	53.7060	1224
60	43.7506	2336	53.2866	1424
70	43.5495	2472	53.0856	1512
Max.	41.8667	5776	51.3990	3592

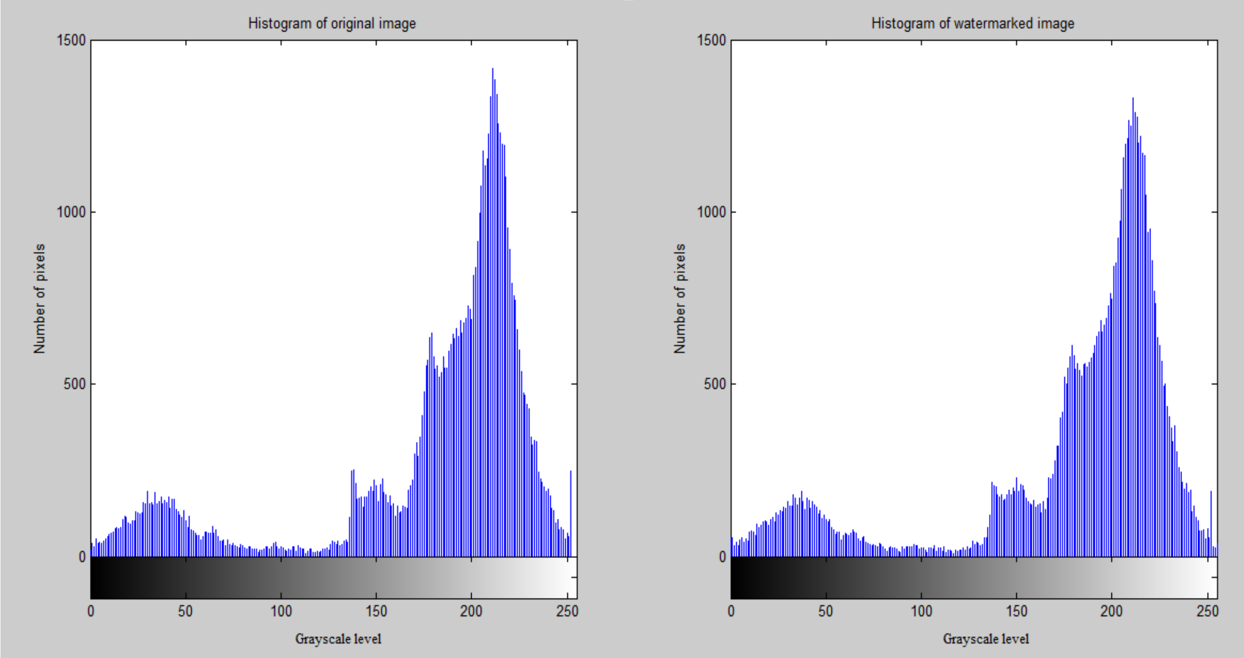
TABLE V. PEAK SIGNAL TO NOISE RATIO (dB) AND EMBEDDING CAPACITY (bits) FOR BARBARA

Threshold values	Proposed Method		Li *et al.* [3]	
	PSNR	E.C.	PSNR	E.C.
0	45.7785	1800	55.3147	1120
20	45.2008	2192	54.7392	1368
40	44.6717	2776	54.2069	1712
60	44.1677	3424	53.6987	2112
70	43.9054	3824	53.4391	2384
Max.	42.2102	12744	51.7321	7936

TABLE VI. PEAK SIGNAL TO NOISE RATIO (dB) AND EMBEDDING CAPACITY (bits) FOR FISHING BOAT

Threshold values	Proposed Method		Li *et al.*[3]	
	PSNR	E.C.	PSNR	E.C.
0	46.6353	2264	56.1634	1376
20	45.8742	2912	55.3955	1752
40	45.0646	3842	54.5839	2304
60	44.2741	5080	53.7916	3048
70	43.9286	5840	53.4414	3512
Max.	42.3182	14800	51.8237	9096

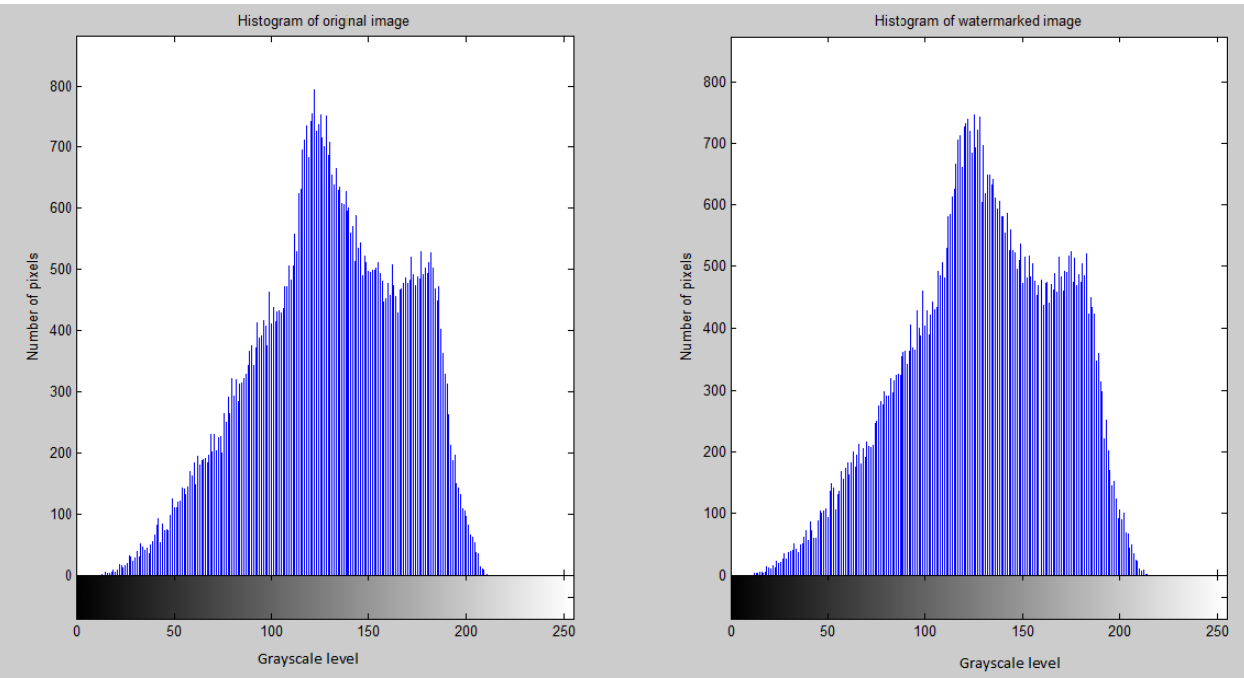
(i) AIRPLANE (F-16)



(a) Histogram of Airplane image (F-16)

(b) Histogram of marked Airplane image (F-16)

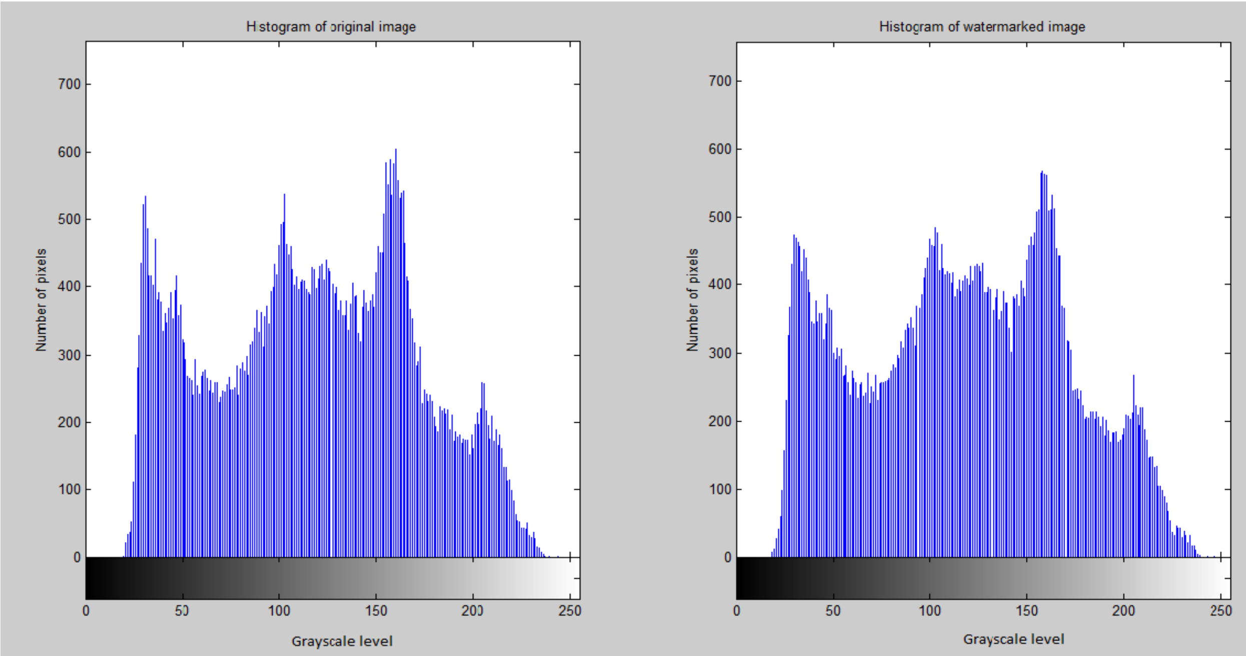
(ii) BABOON



(a) Histogram of Baboon image

(b) Histogram of marked Baboon image

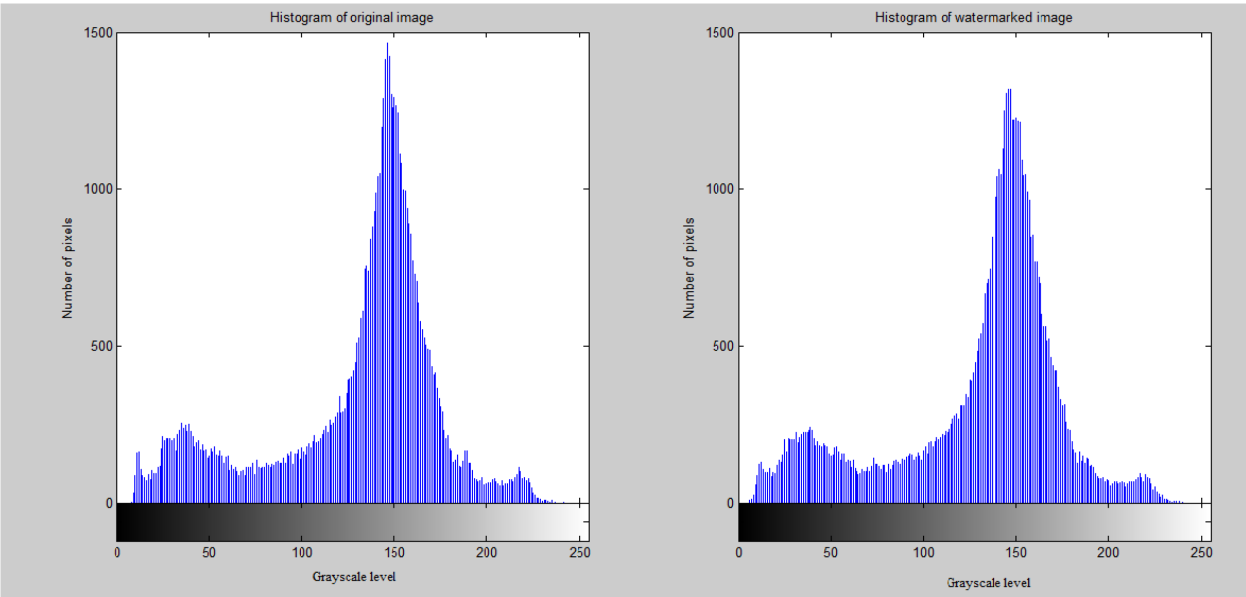
(iii) BARBARA



(a) Histogram of baboon image

(b) Histogram of marked Baboon image

(iv) FISHING BOAT

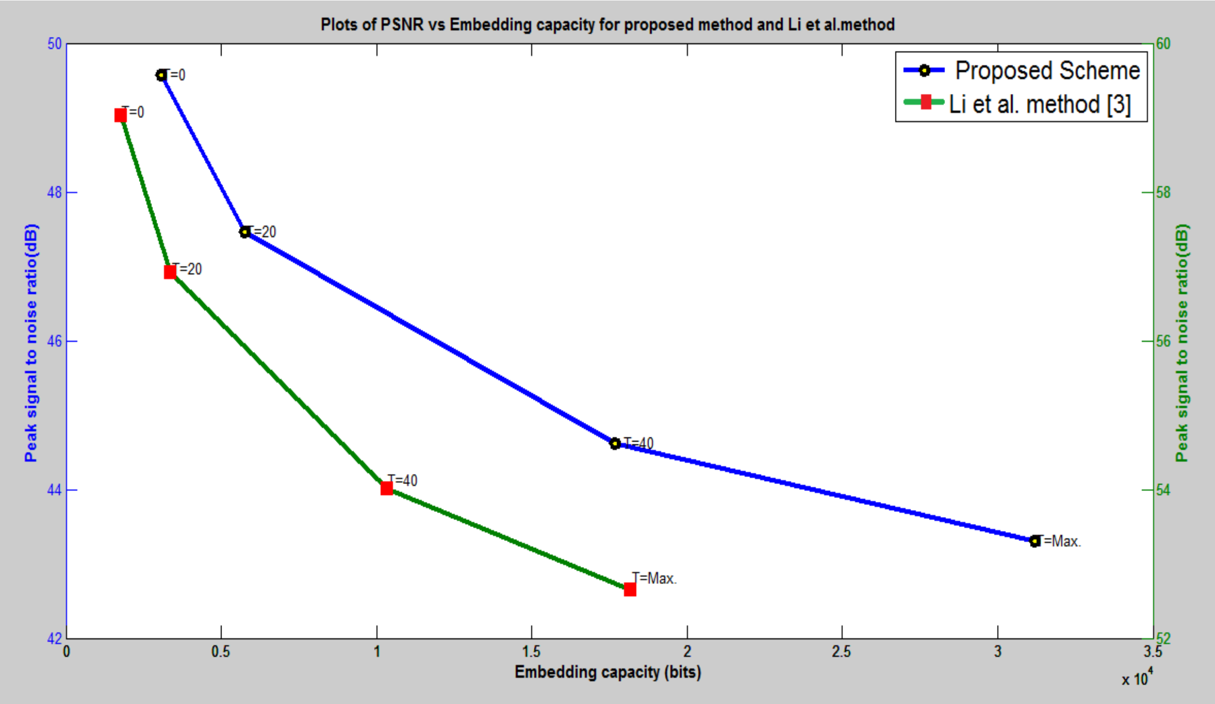


(a) Histogram of Fishing boat image

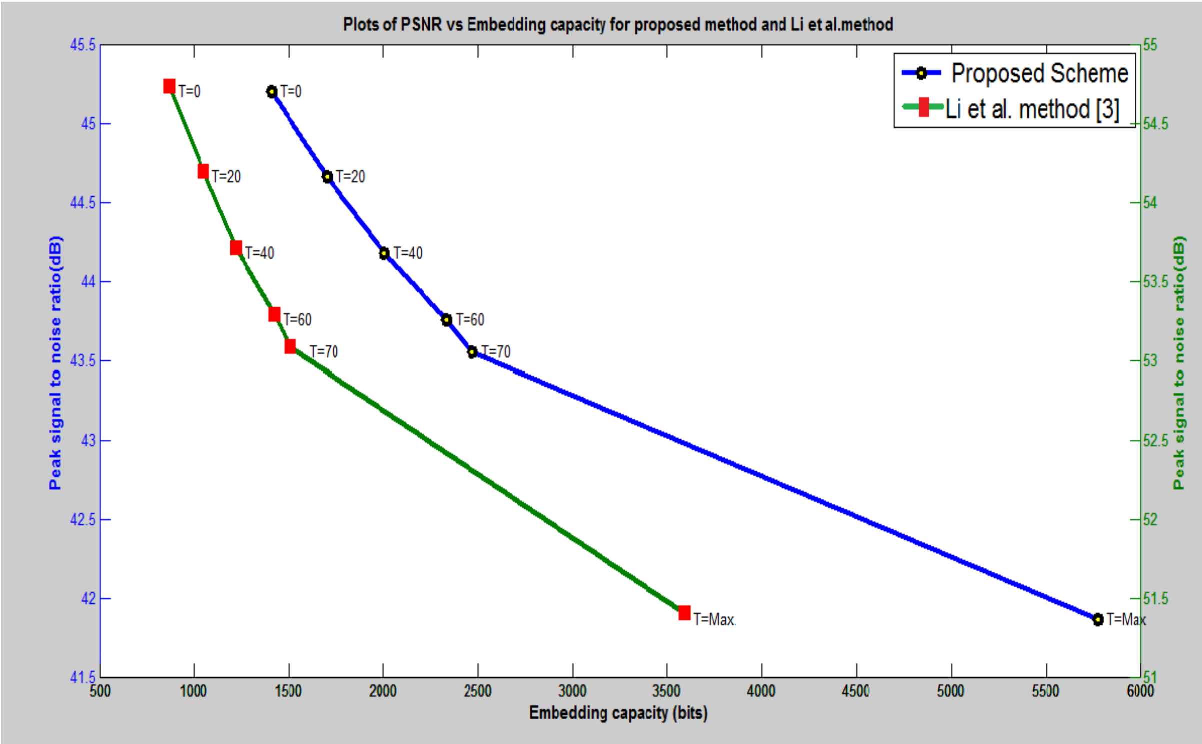
(b) histogram of marked Fishing boat image

Fig. 3: Histograms of original and marked images (i) Airplane (F-16) (ii) Baboon (iii) Barbara (iv) Fishing boat

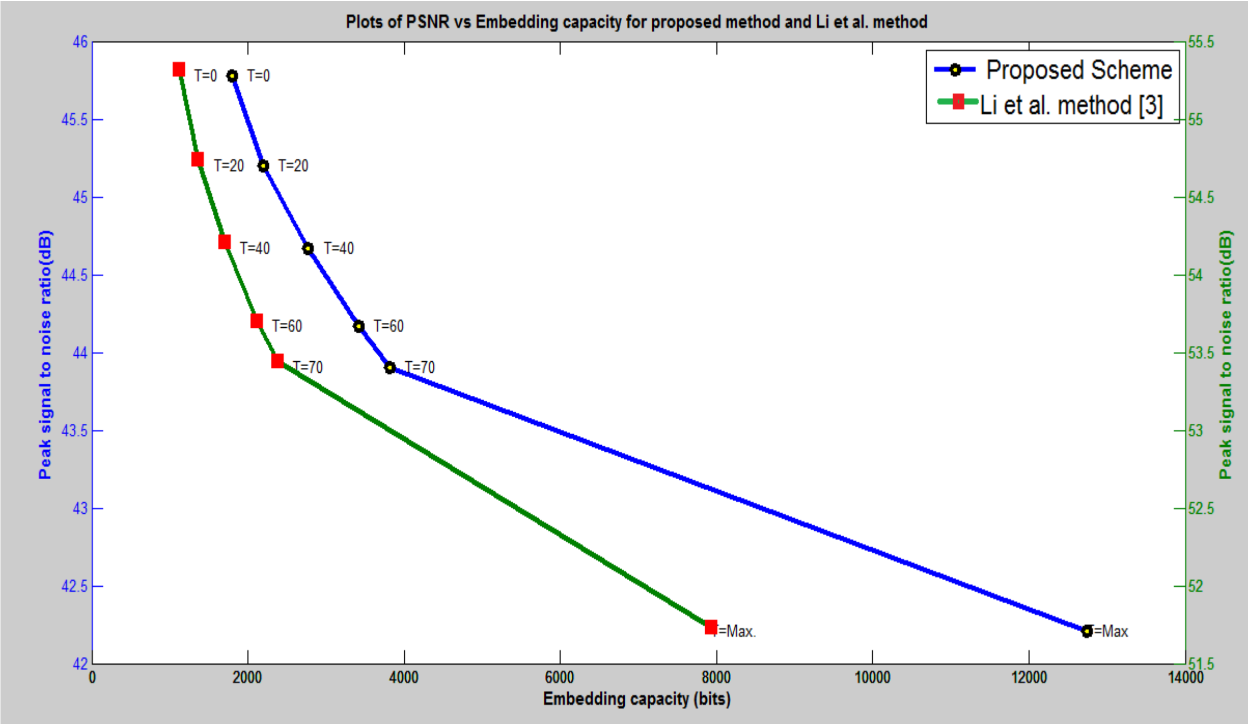
(i) AIRPLANE (F-16)



(ii) BABOON



(iii) BARBARA



(iv) FISHING BOAT

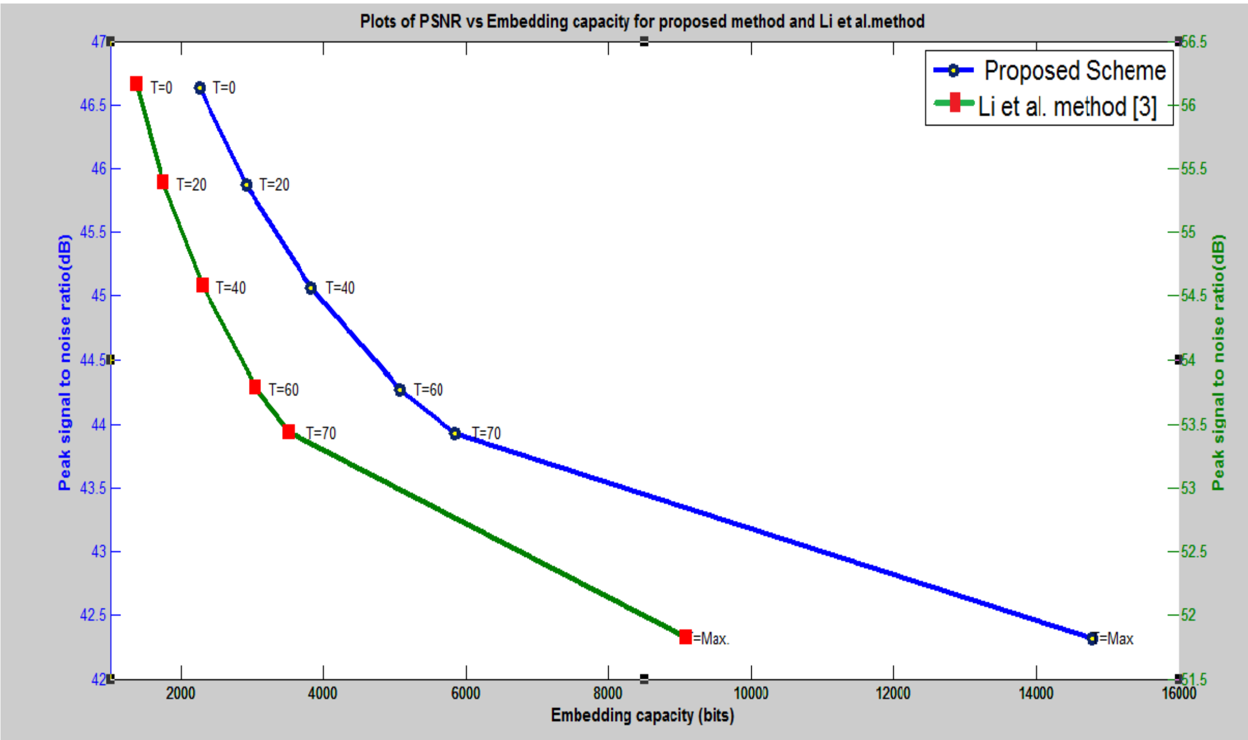


Fig. 4: Performance comparisons between our scheme and Li *et al.* method [3] for (i) Airplane (F-16) (ii) Baboon (iii) Barbara (iv) Fishing boat

It has been found that presented method can embed two bits of data instead of one bit proposed in Li *et al.* method [3]. The threshold parameter has been evaluated in this algorithm which can control the embedding capacity of the algorithm in such a way that at zero or lower thresholds, the embedding capacity is less. Referring to Tables III, IV, V and VI, we can see that our method improves Li *et al.* [3] by increasing embedding capacity. In this method, PSNR is decreased because when we increase the embedding capacity, the quality of watermarked image decreases, but we cannot see it through naked eyes or it is not visible to human perception. In this work, PSNR comes out to be near 42 dB and PSNR upto 30 dB is acceptable and is considered to be of good quality image [11]. The text data is used for embedding and it has been found that the presented algorithm is fully reversible and able to extract original text used in embedding process.

V. CONCLUSION

This paper covers the study of different techniques for reversible watermarking, which is mostly adopted in application such as data hiding and watermarking. Along with this, we have modified an algorithm to make it useful for embedding binary data in the images. It has been found that our presented method can embed two bits of data instead of one bit proposed in Li *et al.* method [3]. The threshold parameter has been evaluated in this algorithm which can control the embedding capacity of the algorithm in such a way that at zero or lower thresholds, the embedding capacity decreases as it chooses those pixel-pairs of embedding where there is variation of intensity values i.e. edges etc. Hence smooth regions and homogeneous region in the image can be excluded while embedding hence increase impercibility of the image. We choose different threshold values to compare the performance of existed method and Li *et al.* method [3]. It has been found that the presented algorithm is fully reversible and able to extract used in embedding process.

ACKNOWLEDGMENT

The Authors are grateful to Department of Electronics and Communication, Punjabi University, Patiala; Research Lab, University College of Engineering, Punjabi University, Patiala.

VI. REFERENCES

[1] Zhicheng Ni, Yun-Qing Shi, Nirwan Ansari, and Wei Su "Reversible data hiding", *IEEE Transactions On Circuits and Systems For Video Technology,* Vol. 16, No. 3, pp. 354-362,March 2006.

[2] W. Hong, T. S. Chen, and C. W. Shiu, "Reversible data hiding for high quality images using modification of prediction errors," *Elsevier-The Journal of Systems and Software* , Vol. 82, No. 11, pp. 1833–1842, Nov. 2009.

[3] [Xiaolong Li, Weiming Zhang, Xinlu Gui, and Bin Yang "A novel reversible data hiding scheme based on two- dimensional difference-histogram modification" *IEEE Transactions on Information Forensics And security*, Vol. 8, No. 7, pp. 1091-1099, July, 2013.

[4] Sang-Kwang Lee, Young-Ho Suh and Yo-Sung Ho, "Reversible image authentication based on watermarking", *in Proceedings of IEEE ICME,* pp. 1321-1324, Year 2006.

[5] Wei-Liang Tai,, Chia-Ming Yeh, and Chin-Chen Chang," Reversible data hiding based on histogram modification of pixel differences" *IEEE Transactions On Circuits and Systems For Video Technology,* Vol. 19 , No.6, pp.906-910, Year 2009.

[6] Zhicheng .Ni, Y.Q. Shi, N. Ansari, andW. Su,"Reversible data hiding," *IEEE Transactions on Circuits Systems and Video Technoogy.*, Vol. 16, No. 3, pp. 354–362, March 2006.

[7] D. M. Thodi and J. J. Rodriguez, "Expansion embedding techniques for reversible watermarking," *IEEE Transactions on Image Processing,* Vol. 16, No. 3, pp. 721–730, March 2007.

[8] Wein Hong, " Adaptive reversible data hiding method based on error energy control and histogram shifting" *Elsevier- Optics Communication* , pp. 101-108, Year 2012.

[9] S.L.V. Krishna , B. Abdul Rahim, Fahimuddin Shaik and K. Soundara Rajan, "Lossless embedding using pixel differences and histogram shifting technique" , *IEEE Transactions on Information Forensics and security,* Year 2010.

[10] A Randall "A Novel Semi-Fragile Watermarking Scheme with Iterative Restoration" *Department of Computing, School of Electronic and Physical Sciences, University of Surrey, Guildford, Surrey, GU27XH.*

Shivani Sharma currently pursuing M.Tech at Department of Electronics and Communication Engineering, Punjabi University, Patiala, India. Area of interest is Digital Image Processing.

Kulwinder Singh Associate Professor at Department of Electronics and Communication Engineering, Punjabi University, Patiala, India. He is M.Tech. and pursuing Ph.D. His area of interest is Fiber Optic Communication , Wireless Communication and Embedded Systems.