Histogram Modified Reversible Data Hiding based on Canny Edge Detection and Prediction Error expansion

Revathi A.S, S.H. Karamchandani, PhD ,Yukti Bandi, Deepshikha Hinger
Assistant professor
EXTC,DJSCOE,Vile parle,
Mumbai

ABSTRACT
In this paper, based on histogram modification, a feasible reversible data hiding scheme is proposed by using prediction error expansion and edge detected information**.**The benefit of RDH lies on security of hidden data as well as quality of color image used to carry this data. RDH provides the facility of retrieving the embedded data as well as the cover medium used for hiding data without any degradation. The algorithm makes use of canny edge detection to exploit correlation between the three channels in color image. This will add the advantage of increasing accuracy of prediction errorin one channel .Histogram modification is the method to extract and embed the data in image. For this, image is divided into two blocks and histograms of these blocks are modified. For finding overflow and underflow pixels during embedding, it uses a location map. The algorithm can embed and extract the data even though the color image has subjected for different noises like Gaussian noise, salt and pepper noise, image rotation and staircase artifacts.

Keywords
Prediction error precision, Histogram modification, Canny edge detection, Location map

1. INTRODUCTION
The communication of informationthrough an open media produces the chance of losing security of that information. Cryptography, steganography and reversible data hiding are some of the techniques to deal with this issue. Reversible data hiding [1] is a significant method by which sender can conceal secret information inside a cover medium and the receiver is able to extract embedded data as well as cover medium without any distortion. The hiding information can be anything like image, video, data and cover medium can also be gray scale/color image, video, DNA.etc. Steganography [2] also deals with data embedding but this technique will destroy the quality of cover medium. Reversible data hiding has key role in the circumstances where cover medium and hidden information have equal importance as in the field of medical science[3] , image transcoding, video transcoding [4] etc.

Reversible data hiding algorithms broadly fall in 3 domains. Compression domain algorithms are the basis of compression domain RDH schemes. In [5] which is based on compression algorithm, to obtain efficient hiding and extraction, side match vector quantization technique and search order coding were used. Transformed index table of cover image is generated and confidential data is embedded into it. Wavelet transform and cosine transform form basis of RDH algorithms fall in frequency domain. A wavelet lifting scheme is used in [6] and a sorting strategy is also employed to get high embedding capacity. Now a days the most popular RDH schemes fall under spatial domain. Histogram modification [7], difference error expansion [8] and prediction error expansion [9] are the main techniques used in spatial domain RDH schemes. In [7] the cover image is divided into non overlapping blocks and their original histograms are also generated. Here Histogram modification process is based on mathematical sets. There is a universal set which contains the whole pixel values in the specific image. Then there are two disjoint sets, one of which contains pixel values used for embeddingdata and second one contains pixel values used for shifting. A location map is also embedded here to understand the position of overflow and underflow pixels. Thodi and Rodrigues[8] combined difference expansion of pixels with histogram generation and achieved high PSNR values and embedding capacity. Li et al [9] demonstrated a hiding scheme for digital image. The algorithm gained the advantage of pixel value ordering and prediction error expansion .The digital image is separated into a number of non overlapping blocks. The maximum and minimum value of each block is predicted by using pixel value ordering based on other pixel values in one particular block. Data embedding is accomplished through prediction error expansion. For this, prediction error histogram is divided into inner and outer region. Use of prediction error, increased the data embedding capacity and decreased the distortion in extracted cover image. Presently reversible data hiding algorithms are combined with Genetic algorithms [10] to optimize the locations and attributes for hiding the confidential information.

The above discussed algorithms used to hide data in gray scale images only. A few numbers of works are carried out on color images [11, 12]. These 2 algorithms considered color image as gray scale image and embedded data in 3 different channels by not exploiting any attribute relating these channels. In [12] Prediction error expansion is used to make the most of correlation of three channels in the experimented color image. But feasibility of this algorithm fails for attacks effected images.This work embeds the confidential data inside a color image. Data embedding has done sequentially in 3 color channels individually. By comparing with previous algorithms, this paper takes the improvement of increased accuracy in prediction error to achieve better embedding. Edge pixels in 3 color channels play key role for this. This is based on the fact that although pixel values differ in three color channels, it can be found similar edge information in channels. So it can be state that there is a correlation exist between the channels in a color image. This attribute of color image can be used to enhance precision of prediction in one channel. To accomplish this, with current pixel, a reference pixel from other channel is also considered. For red and blue channel pixels, green is the reference pixel and for green channel pixel, red channel pixel is the reference. Since cannyedge detection technique is capable of finding wide range of edges, it is used in this work. The correlation between red and green channels and green and blue channel is exploited here more. A fragment of cryptography is also added in this work. Before doing embedding sender has to enter a password and the same password should be entered by receiver during extraction. This algorithm is feasible with noise affected images also. For images with or without noise, this RDH scheme provides peak signal to noise ratio above 50dB.A location map also embedded to identify overflow and underflow pixels.

2. PROPOSED METHOD

This section includes details of color image, canny edge detection and correlation, prediction error and location map creation

2.1 Color images

An image can be defined as a two dimensional function that provides a measure of some characteristics such as brightness or color of viewed scene. Images lie under a wide range of classification. This paper deals with RDH on color images. The color image can be of 24 bit format or 8 bit format. In former, each pixel value is represented by using three bytes, indicating red green and blue (RGB) components. It can hold up 256*256*256 possible combination of colors but needs a storage space of 921.6kB.

2.2 Canny edge detection

Edge detection is a method that comprises a set of mathematical operations which help to find points in a digital image at which the intensity changes sharply or, more formally, has discontinuities . Edge detection reduces the amount of image details that have to be processed and filter out the unnecessarydata while maintaining the relevant structural information of the image. Thiswork uses canny edge detection since it avoids missing of edges, false edge detection..etc. Noise reduction, finding gradient intensity of the image, non maximum suppression and tracing edges through image and hysteresis threshold are the different processes involved in this canny algorithm .

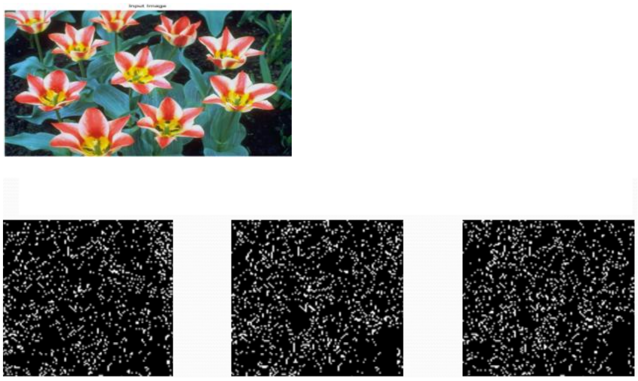


Fig 1: The image Tulips and canny edge detection of red, green and blue components

Specifically for the color image, it can efficiently take the benefit of the edge information obtained from another color channel. It is obvious that the edge details extracted from different color channels is alike to each other. Therefore if an edge is detected in one color channel, there would be a matching edge at the same position in the other channels. Even though hiding has done sequentially to different channels, when performing prediction for some pixel samples in one channel, prediction precision can be increased using the edge information obtained from another channel.In this paper prediction accuracy of red and blue channel pixel samples can be enhanced by taking green channel pixels as the reference and for green channel, red channel pixels are taken as the reference.

Fig 2: Canny edge detection of red and blue channels by taking green channel as reference and of green channel by taking red channel as reference

To determine edge pixels in each color channel, with the current pixel P_c, it has to take a reference pixel P_r.Two different parameters have to be calculated to exploit the edge information. For this consider Fig 3.It contains 2 separate group of pixels, indicated by white and gray colors. Data embedding is done into these two pixels in two stages [12,13]

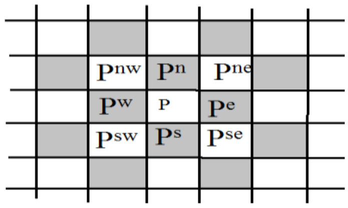


Fig 3: Neighborhood of pixel P

One parameter is difference of average of P_r's eight neighbors and P_r itself.

$$B_d = \left|\sum_{j=1}^{8} \lambda_d^j P_r^j - P_r\right| \quad (1)$$

Where $\lambda_d^j = 1/8$ and P_r^jindicates 8 neighbors of P_r . Second parameter is related with edge estimation. $B_e = \left|\sum_{j=1}^{8} \lambda_e^j P_r^j - \mathrm{Pr}\right|$(2)

λ_e Contains zero and nonzero values. Where

$B_e = \min\{B_h, B_v, B_d, B_{ad}\}$(3)

B_h, B_v, B_d, B_{ad} are edge estimation parameters in horizontal, vertical, diagonal and anti diagonal directions. From figure 3 $B_h = \left|\frac{P_r^w + P_r^e}{2} - P_r\right|$(4)

This is horizontal edge estimation. Similarly B_v, B_d, B_{ad} can be calculated from figure 3. The value of λ_e depends upon B_e value. When $B_e = B_h$, $\lambda_e = (0,0,0,0.5,0.5,0,0,0)$. ($B_d$ -B_e) has worth meaning in determining a pixel's position. In edge regions, B_d cannot be zero, since nearby pixels values differ by large values and in the smooth regions, the situation is vice versa. Hence when (B_d-B_e) is greater than a predefined threshold value, then it can be guaranteed that the current pixel lies in edge region. The threshold value can be selected by maintaining the criteria to get high PSNR value. This work deals with threshold equal to 10.

2.3 Accuracy of prediction error

Extraction of edge information enhances the precision of prediction .When current pixel lies in edge area, then it can be found that (B_d- B_e) is greater than threshold .Then prediction of current sample is given by [12,13]

$$\widehat{P_c} = \sum_{j=1}^{8} \lambda_e^j P_c^j (5)$$

If the current pixel is not in the edge area,that is when (B_d- B_e) is smaller than threshold ,then no significance for edge information components. The prediction of such a pixel is

$$\widehat{P_c} = \frac{P_c^w + P_c^e + P_c^n + P_c^s}{4} \quad (6)$$

Prediction error can be given by the equation

$$E_p = P_c - \lfloor P_c^{\wedge} + 0.5 \rfloor (7)$$

To deal with prediction error expansion hiding, it needs to divide prediction error histogram into one inner field and two outer fields by introducing two integers K_1 and K_2.Hence prediction error in these regions can be modifies as

$$E_p{}' = \begin{cases} 2E_p + n & \text{if } E_p \epsilon (K_1, K_2) \\ E_p + K_1 & \text{if } E_p \geq K_1 \\ E_p + K_2 & \text{if } E_p < K_2 \end{cases} \quad (8)$$

Where n is the embedded bit, either 0 or1

$$P_c' = \lfloor P_c^{\wedge} + .5 \rfloor + E_p{}'(9)$$

$$E_p = \begin{cases} \left\lfloor E_p{}'/2 \right\rfloor & \text{if} E_p{}' \epsilon (2K_1, 2K_2) \\ E_p{}' - K_1 & \text{if} E_p{}' \geq 2K_1 \\ E_p{}' - K_2 & \text{if} E_p{}' < 2K_2 \end{cases} \quad (10)$$

$$P_c = \lfloor P_c^{\wedge} + .5 \rfloor + E_p (11)$$

By performing equation 9, the marked or embedded pixel can be obtained.At the receiver , to extract exact pixel value, prediction error should be retrieved first.This can be accomplished by equation 10.Then with the help of equations 7 and 11,exact value of pixel can be obtained

2.4 Location map creation

Each and every pixel in a image can have the intensity value from 0 to 255. When image is subjected for different process like hiding, then there is a possibility of varying this pixel intensity values. That is some pixel values may rise above 255 and some other values may decrement to less than 0. This issue often occur in images is known as overflowing or under flowing. As a remedy for this problem, before doing embedding it can find the location of underflow and overflow pixels and can keep a record of the same. Therefore the sender can hide confidential data as well as the location map inside a color image and receiver can make use of this location map to identify this problematic locations. Each current pixel can have a parameter value given by

$$D = \Omega + \Psi (12)$$

Ω is related with underflow or overflow problem and Ψ indicates surface smoothness of image.Ω is given by

$$\Omega = \lambda. .Q(b) \quad (13)$$

Where $Q(y) = \max(\boldsymbol{N}(y), \boldsymbol{N}(255\text{-}y))(14)$

Where y is the value of current sample and ***N(.)*** is standard Gaussian function. Since during embedding, the value of pixel gets changed, instead of y,it is more convenient to take mean value of 4 neighbors of y and is represented as b. λ is used to fine-tune the weighing relation between Ψ and Ω and this work takes its value as 1000. The value of Ψ is given by

$$\Psi = \frac{1}{4}\sum_{j=1}^{4}(\Delta_j - \Delta)^2 \quad (15)$$

$$\Delta_1 = |P^n - P^w| \quad (16)$$

$$\Delta_2 = |P^w - P^s| \quad (17)$$

$$\Delta_3 = |P^s - P^e| \quad (18)$$

$$\Delta_4 = |P^e - P^n| \quad (19)$$

And $\Delta = \frac{\Delta 1 + \Delta 2 + \Delta 3 + \Delta 4}{4}$ (20)

D is very helpful to find the location of underflow and overflow pixels. In location map , the pixels which can be modified are indicated by 0 and the problematic positions are indicated by 1.The embedding of location map can be done by using the equation

3. DATA HIDING AND EXTRACTION ALGORITHM

Embedding of information should be done in sequentially into 3 color channels. There is an order for embedding of data.First data should be hided into blue and red channels using green channel as reference. Then green channel is considered with red channel as reference. The data embedding process is achieved through 2 stages. To identify this, alternate pixel positions are colored as gray as in figure1.So In first stage data should be inserted into pixels indicated as white in figure 1 and in second stage , it is inserted into pixels indicated as gray.

3.1 Data hiding algorithm

Data embedding algorithm is as follows

Step 1: Indicate the pixels to be used for embedding with the sequence (P_1,P_2,….P_n). Calculate D value related with each P_i using equations from 12 to 20.Based on this sort the P_i sequence in ascending order and create a new sequence (P_1', P_2',…P_n')

Step 2:Assign K_1=0 and K_2= -1

Step 3: Embed the data into new sequence P_i'. For each value in sequence, calculate the prediction error value using equation 5 or 6.Then perform hiding based on equation 7 to 9.Denote embedded data samples as P_{ie}'.

Step 4:Location map (LM) can be created as follows

a) If P_{ie}' is not belong to [0,255], set LM(h)=1, h=h+1 and change P_{ie}' to P_i'.

b)If P_{ie}'belongs to [0,255], set LM(h)=0, h=h+1 and current sample is denoted as P_{ie}'.If all the information can be embedded into the color image, keep the final position of data embedding as f and move to step 5. Otherwise increase K_1 or decrease $K_2$and repeat step 3 and 4.

Step 5: There may be some remaining expandable sample after the position of f. To determine it use equations 6 to 9. If number of remaining expandable sample is not large enough for embedding the created LM and 40 LSB of host , increase K_1 or decrease K_2 by one and repeat step 3 and 4.Otherwise record the length of LM as l and go to next step.

Step 6: Record LSB's of first 40 pixels in the current channel. Replace these LSBs by auxiliary information like K_1, K2,l and f.Embed the LSB sequence and LM into remaining sorted samples after the position of f using equations 7 to 9.

3.2 Data extraction algorithm

The extraction algorithm can be conducted as reverse of hiding algorithm. The secret information should be extracted from green first and then from red and blue. First stage of extraction consists of extraction from the gray pixels and second stage consists of extraction from white pixels.

Step1: Determine all pixels for data extraction. Arrange them in ascending order based on D and get the sequence P_i'.

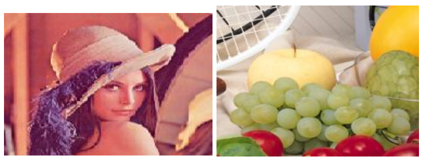
Step 2:Read LSBs of first 40 pixel to get values of K_1 , K_2, f and l. In a reverse manner, extract LSB values and Location map. Along with this original image can be constructed using equations 10 and 11.

Step 3: Reveal the secret data from the sequence(P_f', P_{f-1}',P_{f-2}',….P_1') using equations 10,11.

4. RESULTS AND DISCUSSIONS

To carry out the experiments, four 24 bit color images are used. They are Tulips.png, Lena.jpg, Airplane.png, Fruits.png.Based on above algorithm and using [7] reversible data hiding has achieved in all four experimental images.

(a) **(b)**



(c) **(d)**

Fig 4: Original color images used for embedding data.

(a) Tulips(b) Airplane (c)Lena (d) Fruits

For embedding and extraction of data, a few steps have followed based on algorithm. As the first step, image has divided into 3 color channel separately. After this, canny edge detection has applied on separate channel components as in figure 1. To take the advantage of channel correlation, canny edge detection has applied on blue and red channel by taking green as reference channel and on green channel by taking red as reference channel as in figure 2.For the same purpose, difference images have also generated.

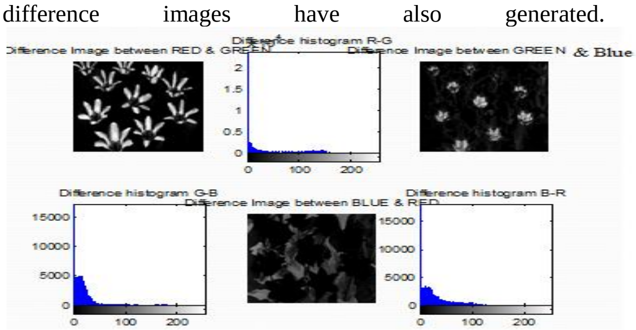


Fig 6: Difference images between red & green,green & blue, red & blue and corresponding histograms.

The following figures show the results of embedded and extracted images based on the algorithms..

Fig 7:Data embedded blue, red and green channel images

Fig 8: Data embedded combined image

Fig 9: Data extracted green,red and blue channel images

Fig 10: Extracted original image

4.1Security of information

This RDH scheme itself maintains the confidentiality of information. To ensure that, a private key or password has given to both embedding and extracting stages of each color channel separately. The password for embedding and extraction should be same for red channel. Likewise there is password for blue and green channels. It adds the benefit of security as well helps to distinguish the data in different channels.

4.2 Embedded data & Extracted image

Different data are embedded in 3 channels. The details are following .

Number of bits embedded in blue channel : 304

Number of bits embedded in red channel : 2184

Number of bits embedded in green channel: 2056

This is same for the 4 experimented images. Following shows the different parameters measured from the extracted image.

Table 1. Parameters measured from extracted images

Tulips.png	R	48%	55.0228	R-G	.8159
	G	45.24%	55.3090	G-B	.6751
	B	6.7%	63.5598	R-B	.4473
Lena.jpg	R	48%	54.2084	R-G	.9105
	G	45.24%	54.9830	G-B	.8785
	B	6.7%	63.5781	R-B	.6763
Airplane.png	R	48%	54.2179	R-G	.9390
	G	45.24%	54.9927	G-B	.9212
	B	6.7%	62.4891	R-B	.8410
Fruits.png	R	48%	54.2188	R-G	.7353
	G	45.24%	54.9843	G-B	.7176
	B	6.7%	63.5994	R-B	.5767

From the Table , it can be understand that there is no distortion occurred for the images. For red and green channels of 4 images, the Peak signal to noise ratio (PSNR) value , that is measure of quality is above 54 dB and for blue channel, it is above 63 dB since blue channel contains less number of bits. The peak signal to noise ration can be calculated by

$$\text{PSNR} = 10 \times \log_{10} \frac{255^2}{MSE} \qquad (21)$$

For original image O and stego image S with size p× q, the MSE is

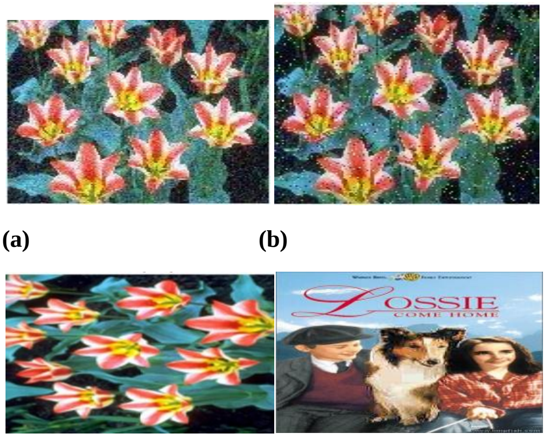
$$\text{MSE} = \frac{1}{p \times q} \sum_{i=0}^{p-1} \sum_{j=0}^{q-1} [O(i,j) - S(i,j)]^2 \quad (22)$$

It can be realized that, the images used for embedding data can be retrieved with high level of quality by using this algorithm. The performance of algorithm is stable for all four images. Expected values are obtained for correlation also. Correlation between two color channels gives the measure of relation between those two channels. Correlation coefficient value ranges from -1.00 to +1.00. The value of Correlation coefficient also denotes that there is no alteration happened for images after embedding and extraction. It can be shown that high correlation is with red and green components and less correlation is with red and blue components. From the table 1, it can be analyzed that sum of cross correlation of RG and G-B is greater than sum of cross correlation of R-B and

R-G is greater than sum of cross correlation of R-B and G-B. So from table 1, it can be stated that by using this algorithm, reversible data hiding can be achieved successfully.

4.3 feasibility of Algorithm with noisy images

There may be chances for so many attacks, when the image is transmitted through a channel. Gaussian noise, salt and pepper, image rotation and staircase artifacts are some examples. This section proves that, the prescribed algorithm is practicable , eventhough some noises effected image during the transmission through the channel.From the noisy image, the algorithm is capable of extracting data and removing the particular noise and presents high quality extracted image.



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

Fig 11: Image effected with (a) Gaussian noise (b) Salt and pepper noise (c) Image rotation (d) Staircase artifact

The amount of data embedded in red channel of these images is 2184 bits, blue channel is 304 bits and in green channel, it is 2056 bits. This confidential data has extracted without any distortion from these noisy images. After extraction the quality of images is measured and it showed high value for PSNR of images. Table 2 gives PSNR value of extracted red channel, blue channel and green channel images of tulips.png.

Table 2: PSNR (dB) values of noise effected tulips.png image

Noise	PSNR of red channel	PSNR of Green Channel	PSNR of blue channel
Gaussian noise	54.2113	54.9691	63.5353
Salt & Pepper	54.2402	55.0106	63.5202
Image rotation	54.2015	54.9923	63.5397
Staircase artifact	54.2159	54.9763	63.5302

5. CONCLUSION

The paper presents a feasible and efficient algorithm of reversible data hiding using 512x 512, 24 bit color images. Prediction error expansion is the back bone of this algorithm. To increase the precision of this in one color channel, information from other color channel used based on the concept of channel correlation. Although the pixel values are not similar in different channel, it can be seen that there is similar edge distribution between the three channels. Canny edge detection is used to find this. Using this algorithm , information can be embedded and extracted in more secured way. The color image used for embedding is extracted with a peak signal to noise ratio above 50 dB for the 3 channels of all four experimented images. The other statistics of image like correlation is also maintained in extracted image in proper manner. The possibility of algorithm has checked with noisy images. Algorithm keeps the confidentiality of information in these images also. Data embedded and cover image has extracted without distortion in this case too. Comparing to the previous algorithms, which embeds data into three channels independently without considering similar edge distribution, this algorithm gives promising results.

6. REFERENCE

[1]. Z.Ni, Y. Q. Shi, N. Ansari, and W. Su, “Reversible data hiding,” *IEEE Trans.Circuits syst. Video Technol.*, vol.16, no.3, pp. 354-362, Mar.2006.

[2]. Yang,Liu,Luo, and Zeng “ Pixel Group Trace Model-Based QuantitativeSteganalysis for Multiple Least-Significant Bits Steganography”, ieee transactions on information forensics and security, vol. 8, no. 1, january 2013

[3]. G.Coatrieux, C.L. Guillou, J.M. Cauvin, and C. Roux “Reversible watermarking for knowledge digest embedding and reliability control in medical images” *IEEE Trans. Inf. Technol.Biomed*, vol. 13, no. 2,pp. 158-165, Mar. 2009

[4]. Kuo-Liang Chung, Huang, Chang and Mark Liao “Reversible data hiding -based approach for intra frame concealment in H.264/AVC ” ieee transactions on circuits and systems for video technology, vol. 20, no. 11, november 2010

[5]. Chang,Nguyen,and Lin “A novel VQ- based RDH scheme by using hybrid encoding strategies”The Journal of Systems and Software 86 (2013) 389– 402.

[6]. L.Kamastra and H .J. A. M. Hejimans, Reversible data embedding into images using wavelet techniques and sorting,” *IEEE Trans. Image Process*., vol.14. no. 12, pp.2082-2090, Dec-.2005.

[7]. Xiaolong Li, Bin Li, Bin Yang, and Tieyong Zeng “General framework to histogram shifting based reversible data hiding”ieee transactions on image processing, vol. 22, no. 6, june 2013

[8]. D.M. Thodi and J.J. Rodriguez,” Expansion embedding techniques for reversible watermarking.” *IEEE Trans. Image Process*., vol.16, no. 3, pp.721-730, Mar.2007

[9]. Xiaolong Li,Jian Li,Bin Li,Bin Yang “ High fidelity RDH scheme based on pixel value ordering and prediction error expansion” Signal Processing 93 (2013) 198–205

[10]. Khurram Jawad, Asifullah Khan “Genetic algorithm and difference expansion based reversiblewatermarking for relational databases”The Journal of Systems and Software 86 (2013) 2742– 2753

[11]. A.M. Alattar, Reversible watermark using the difference expansion of a generalized integer transform, IEEE Transactions. on Image Processing 13 (8) (2004) 1147–1156

[12]. A.S.Revathi,A.Amsaveni,Dr.P.T.Vanathi “Prediction error based reversible data hiding for color images based on cross channel correlation”, Conference proceedings, national conference on Communication, Information& Telematics(CITEL 2014) conducted by ECE Dept. of Kumaraguru College of Technology, Coimbatore

[13]. Jian Li,Xialong Li,Bin Yang “Reversible data hiding scheme for color image based on prediction-error expansion and cross channel correlation ”Signal Processing 93 (2013) 2748–2758