Reversible Data Hiding Method Based on Two-Dimensional Difference Histogram and Block Division

Nitheesh C. N.
M. Tech Student
Dept. of Electronics and Communication Engineering
MBCET, Nalanchira
Kerala, India

Arathy C. Haran V.
Assistant Professor:
Dept. of Electronics and Communication Engineering
MBCET, Nalanchira
Kerala, India

***Abstract*— Reversible data hiding (RDH) aims to embed secret message into a cover image by slightly modifying its pixel values, and the embedded message as well as the cover image can be completely recovered from the marked content. Many RDH methods have been proposed e.g., the methods based on lossless compression, difference expansion, histogram modification, prediction-error expansion, and integer transform etc. The histogram-based methods modify the histogram in such a way that certain bins are shifted to create vacant space while some other bins are utilized to carry data by filling the vacant space. This type of methods can control the embedding distortion and provide a sufficient embedding capacity. The existing technique is based on two-dimensional difference histogram modification. Here reversible data hiding (RDH) scheme is done by using difference-pair-mapping (DPM). First, by considering each pixel-pair and its context, a sequence consisting of pairs of difference values is computed. Then, a two -dimensional difference-histogram is generated by counting the frequency of the resulting difference-pairs. Reversible data embedding is implemented according to a specifically designed DPM. Here, the DPM is an injective mapping defined on difference-pairs. The embedding capacity of the existing data hiding technique can be increased by combining block division of the input image and the 2-D Difference histogram modification. The block division is done to improve the data hiding capacity, the marked image quality and to improve the hiding places of information in the image.**

Keywords*— Embedding capacity, RDH, DPM, two-dimensional difference histogram.

I. INTRODUCTION

Large amount of data in digital form is transferred through internet. Data hiding is necessary to improve security. Data hiding is a form of steganography that embed a secret message into the cover image. The data to be embedded is referred to as the payload and the embedded image is known as stego image. The main requirement of data hiding is that the hidden information should be invisible and also the marked image quality should be high. Many data hiding techniques have been proposed so far. Among them reversible data hiding has attracted the attention. The term reversible in reversible data hiding referred to as lossless, i.e. both the embedded data and the cover image can be successfully decoded.

The histogram based method attained attention among the reversible data hiding scheme [4]. Ni et al. [5] proposed histogram based method. It considers maximum and minimum points of the histogram in data embedding. But this method has the disadvantage of low embedding capacity and it cannot embed data if the histogram is flat. In 2006, Lee *et al*. [7] proposed the reversible data hiding scheme based on difference histogram. The method uses the pixel pair with difference -1 or 1 to carry the data. In 2009 Tai et al. [8] used binary tree structure (BTS) in difference histogram. The embedding capacity has been improved by Tai et al. [8] method. Hong [9] improved Tai et al.'s method by using dual binary tree structure (BTS). The embedding capacity is further improved with distortion at the same level. Then the median edge detector and error energy estimators are used to identify more pixels which can be used for embedding.

In histogram based method the pixel pair of the given image is considered. Then, the difference value of the pixel pair is obtained and it is projected to one dimensional space [3]. The one dimensional histogram is generated by counting the difference value. The histogram bin with high peak is used to embed data and other bins are shifted. In the two dimensional histogram based method, by considering a pixel-pair and its context, a local image region is projected to a two-dimensional space to obtain a difference pair sequence [1]. A two-dimensional difference-histogram is then generated by counting the difference-pairs. Finally, reversible data embedding is implemented according to a particularly designed difference-pair-mapping (DPM) [1]. Here, the DPM is an injective mapping defined on difference-pairs and it is a natural extension of expansion embedding and shifting techniques used in current histogram-based methods [1]. By using the two-dimensional difference-histogram and this specific DPM compared with the conventional one-dimensional histogram based methods more pixels are used for carrying data and thus an improved embedding performance is achieved [1]. In the proposed method by using block division on the image and applying DPM to each block more data can be embedded.

II. PROPOSED METHOD

The proposed method based on block division is used to improve the data hiding capacity, the marked image quality and to improve the hiding places of information in the image. In this reversible data hiding technique image is divided into blocks [2] and the data are embedded into each block. The data embedding, takes place in three steps: (1) Divide the image into blocks (2) Two dimensional histogram generation of each block (3) Data embedding using difference pair mapping. By block division more pixels of each block can be selected suitable for data embedding. And, thus the overall data embedding capacity can be increased. Also, the PSNR value of each block can be modified so that the overall PSNR value can be increased.

The data embedding is based on difference pair mapping (DPM) and the two dimensional difference histogram. The difference pair mapping (DPM) is a natural extension of expansion embedding and shifting algorithm. The embedding process is based on pixel pairs. The cover image is divided into non-overlapping pixel pairs [1]. One of the pixels in the pixel pair is modified during embedding and the other is shifted. The maximum modification to the pixel will be one so the marked image quality will be high. If the pixel value of the pixel pair is 0 or 255, overflow or underflow will occur and the data embedding is not done.

In difference pair mapping, a pixel pair (x, y) is considered and the difference pair (d_1, d_2) is calculated as $d_1 = x - y$ and $d_2 = y - z$. The count of (d_1, d_2) gives the two dimensional difference histogram shown in fig. 5. Here z is the prediction of y, z can be calculated using GAP predictor as:

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

Where $\{v_1, \ldots . v_5, v_7, v_8\}$ is neighbouring pixels of (x, y) (see in fig.1), $dv = |v_1 - v_5| + |v_3 - v_7| + |v_4 - v_8|$ and $dh = |v_1 - v_2| + |v_3 - v_4| + |v_4 - v_5|$ represent the vertical and horizontal gradients, and $u = \frac{(v_1+v_4)}{2} + \frac{(v_3-v_5)}{4}$. The noise level in the image can be calculated by taking the sum both horizontal and vertical difference of pixels in the block. The pixel block for the GAP predictor is shown in fig. 1.

	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 block for GAP predictor

The modification of the difference pair (d_1, d_2) is in four directions: $(d_1 - 1, d_2)$, $(d_1 + 1, d_2)$, $(d_1 + 1, d_2 - 1)$ or $(d_1 - 1, d_2 + 1)$ as shown in fig. 2. Based on four modifications the reversible data hiding is defined and the reversible data hiding is defined based on this difference pair mapping as shown in fig. 3.

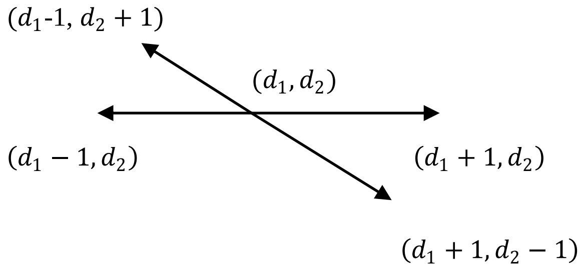

Fig. 2: The four modifications of the difference pair (d_1, d_2)

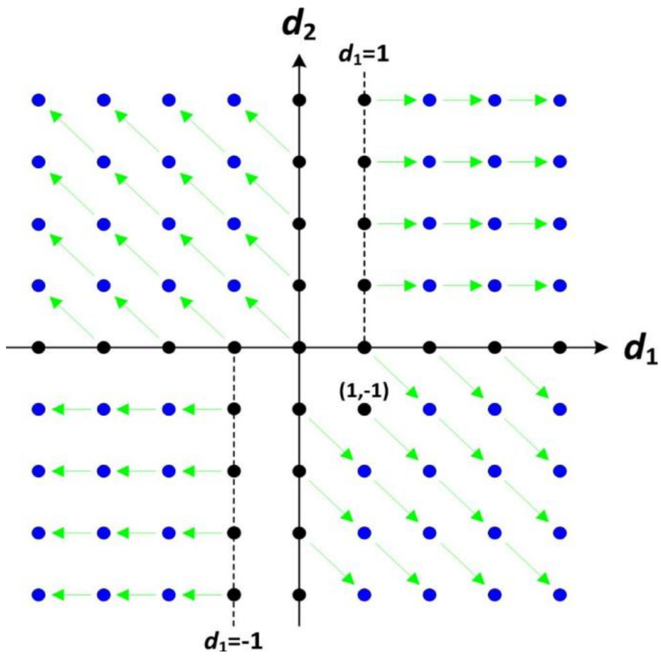

Fig. 3 Illustration of difference pair mapping

Table 1. Marked value of cover image (x,y) according to DPM and different cases of difference pair (d_1, d_2)

Conditions on (d_1, d_2)	Operation in data embedding	Modification direction of the difference-pair	Modification direction of the pixel-pair	Marked value
$d_1 = 1$ and $d_2 > 0$	Expansion embedding	Right	Left	$(x+b, y)$
$d_1 = -1$ and $d_2 < 0$	Expansion embedding	Left	Left	$(x-b, y)$
$d_1 = 0$ and $d_2 \geq 0$	Expansion embedding	Upper-left	Up	$(x, y+b)$
$d_1 < 0$ and $d_2 = 0$				
$d_1 = 0$ and $d_2 < 0$	Expansion embedding	Lower-right	Down	$(x, y-b)$
$d_1 > 0$ and $d_2 = 0$				
$d_1 = 1$ and $d_2 = -1$				
$d_1 > 1$ and $d_2 > 0$	Shifting	Right	Right	$(x+1, y)$
$d_1 < -1$ and $d_2 < 0$	Shifting	Left	Left	$(x-1, y)$
$d_1 < 0$ and $d_2 > 0$	Shifting	Upper-left	Up	$(x, y+1)$
$d_1 > 1$ and $d_2 < 0$	Shifting	Lower-right	Down	$(x, y-1)$
$d_1 = 1$ and $d_2 < -1$				

A. Embedding Algorithm

The embedding process was done as follows:

1) Divide the image into two non-overlapping image blocks.
2) Take the first block and resize the block to the size of the whole image and do the following steps:

a) Except last two rows and last two columns, divide the image block into non-overlapping pixel pairs.
b) The location map L is defined as, for the pixel pair with pixel values 0 or 255 the location map value will be one. Otherwise, it is zero.
c) The location map is losslessly compressed using arithmetic coding.
d) For the pixel pair (x, y) with L=0, compute the noise level and prediction of y.
e) The data will be embedded according to the location map and pixel pair selection threshold T.
f) For a pixel pair (x, y) with location map L=1 the data embedding is not done.
g) If the pixel pair with L=0 and the noise level is greater than threshold T the data embedding is not done.
h) If L=0 and the noise level is less than threshold T the data embedding is done according to table 1. The last embedded pixel pair is ***kstar***.
i) Record the LSB of the first $(12 + 3\lceil log_2 k \rceil + l)$ of the image pixel to obtain the binary sequence S.
j) Embed S to remaining pixel pair, from ***kstar*** to ***kend.***
k) Embed the LSB of the first $(12 + 3\lceil log_2 k \rceil + l)$ pixels with auxiliary information shown below and repeat the embedding steps for the second block.

i. Pixel pair selection threshold: T(12 bits)
ii. Last data embedded pixel pair: ***kstar*** $(\lceil log_2 k \rceil)$
iii. Length of compressed location map: l bits.

B. Extraction Algorithm

The data extraction is done in the following steps:

1) Read the LSB of the first $(12 + 3\lceil log_2 k \rceil + l)$ pixels of the marked image and determine the values of T, ***kstar***, l and ***kend.***
2) From ***kend*** pixel pair the sequence S was extracted.
3) Then the marked image was divided into non-overlapping blocks.
4) The first block was taken and the following extraction steps were done.

a) Divide the block into non-overlapping pixel pairs.
b) The location map information was found out by arithmetic decoding.
c) The noise level and the pixel pair selection threshold T of the marked image was found out.
d) The data extraction would be done if noise level less than threshold and data extraction was based on table 2.
e) Then the data extraction was repeated for the second block.

Table 2. Extracted data bit and recovered pixel pair.

Conditions on (d_1^m, d_2^m)	Extracted data bit b	Recovered value
$d_1^m \in (1,2)$ and $d_2^m > 0$	$d_1^m - 1$	$(x^m - b, y^m)$
$d_1^m \in (-1,-2)$ and $d_2^m < 0$	$-1 - d_1^m$	$(x^m + b, y^m)$
$(d_1^m = 0$ and $d_2^m \geq 0)$ or $(d_1^m = -1$ and $d_2^m \geq 0)$	$-d_1^m$	$(x^m, y^m - b)$

$(d_1^m < 0 \; and \; d_2^m = 0) \; or \; (d_1^m < -1 \; and \; d_2^m = 1)$	d_2^m	$(x^m, y^m - b)$
$(d_1^m = 0 \; and \; d_2^m < 0) \; or \; (d_1^m = 1 \; and \; d_2^m < -1)$	d_1^m	$(x^m, y^m + b)$
$(d_1^m > 0 \; and \; d_2^m = 0) \; or \; (d_1^m > 1 \; and \; d_2^m = -1)$	$-d_2^m$	
$(d_1^m = 1 \; and \; d_2^m = -1) \; or \; (d_1^m = 2 \; and \; d_2^m = -2)$	$d_1^m - 1$	
$d_1^m > 2 \; and \; d_2^m > 0$	No embedded data bit	$(x^m - 1, y^m)$
$d_1^m < -2 \; and \; d_2^m < 0$	No embedded data bit	$(x^m + 1, y^m)$
$d_1^m < -1 \; and \; d_2^m > 1$	No embedded data bit	$(x^m, y^m - 1)$
$d_1^m > 2 \; and \; d_2^m < -1$	No embedded data bit	$(x^m, y^m + 1)$

III. EXPERIMENTAL RESULTS

The improved reversible data hiding has been performed on the test images, including cameraman, tire, baboon etc. The average PSNR value after block division is higher than the PSNR value of the whole image. The variation of the PSNR value of the whole image to that of block division of dataset 1, dataset 2 and dataset 3 is shown in table 3, table 4 and table 5 respectively. The variation of embedding capacity versus PSNR is shown in fig. 6, fig. 7 and fig. 8. The improvement in PSNR value results in increasing the image quality. Also the embedding capacity has been improved. By block division more pixel pairs can be utilized for embedding. The use of DPM and two-dimensional difference histogram for embedding the redundancy in the pixel pair can be reduced and more pixels can be utilized for embedding. The plot of two-dimensional histogram for the cameraman image is shown in fig. 5. The peaks in the histogram show the occurrence of the particular difference pair (d_1, d_2).

The PSNR value can be calculated as:

$$\text{PSNR}=10 log_{10} \left(\frac{255^2}{mse}\right) \quad (2)$$

$$\text{mse} =(1/(M*N))\sum_{i=1}^{M} \sum_{j=1}^{N} (I'(i,j) - I(i,j))^2 \quad (3)$$

Where $I(i,j)$ is the input image and $I'(i,j)$ is the marked image.

(a) Dataset 1 (b) Dataset 2

(c) Dataset 3

Fig. 4. Test images (a) Dataset 1 (b) Dataset 2 (c) Dataset 3

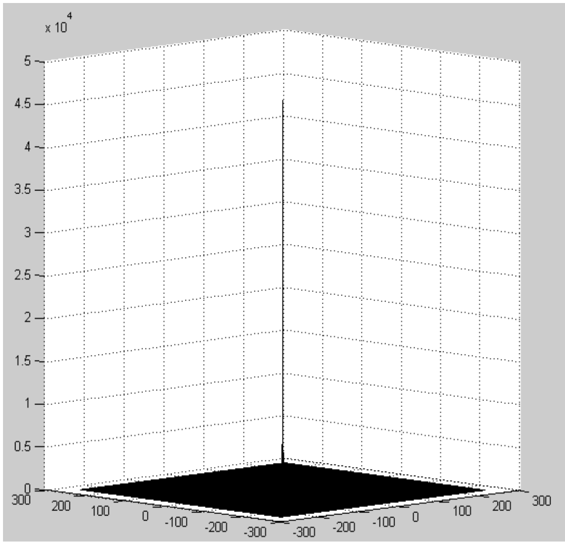

Fig. 5. Plot of two dimensional difference histogram of cameraman image

Table 3. Variation of PSNR(dB) with embedding of dataset 1.

Length of input data (bits)	PSNR (dB) of the whole image	PSNR(dB) of two blocks	Average PSNR (dB) after block division
80	67.0813	67.1232	68.673
		70.2237	
160	67.0360	67.0996	68.63
		70.1757	
192	67.0168	67.0917	68.62
		70.1598	
240	66.9976	67.0774	68.604
		70.1308	
500	66.8830	67.0014	68.49
		69.9862	
1000	66.6530	66.8755	68.31
		69.7611	
5000	65.9218	65.9714	67.071
		68.0641	

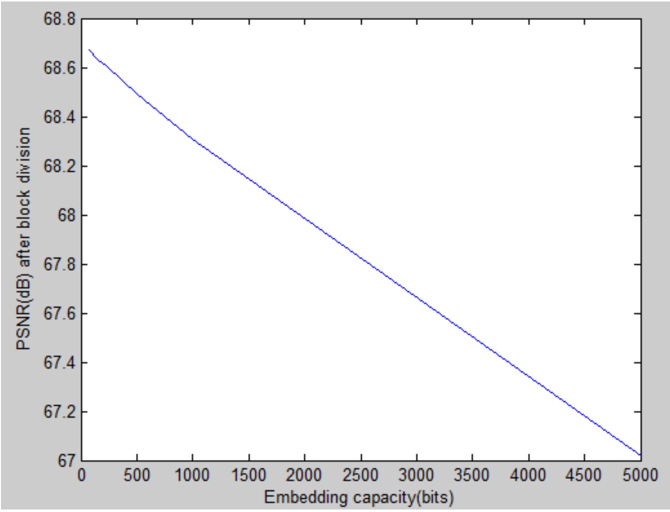

Fig. 6. Variation of PSNR(dB) versus embedding capacity of dataset 1.

Table 4. Variation of PSNR(dB) with embedding of dataset 2.

Length of input data (bits)	PSNR (dB) of the whole image	PSNR(dB) of two blocks	Average PSNR(dB) after block division
24	84.2956	84.1871 86.2956	85.241
32	84.6819	83.9911 86.2956	85.143
48	84.936	83.9911 85.8817	84.936
64	83.8652	83.9911 85.1562	84.573
80	83.7421	83.9911 84.6819	84.3365
160	82.5390	82.6783 84.2544	83.466
320	80.6136	81.4172 82.5849	82.001

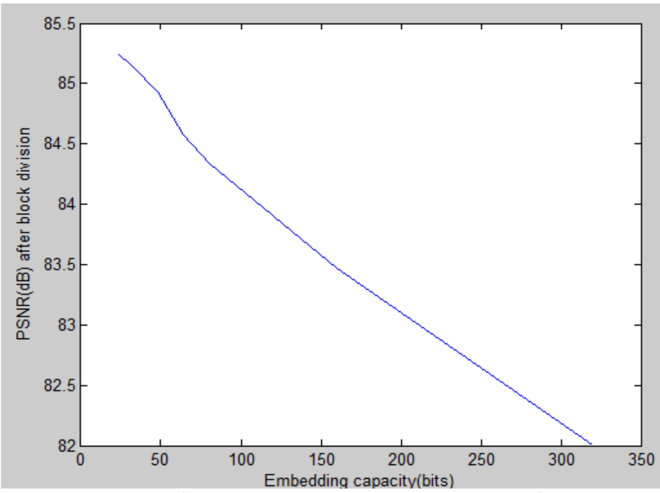

Fig 7. Variation of PSNR(dB) versus embedding capacity of dataset 2.

Table 5. Variation of PSNR(dB) with embedding of dataset 3.

Length of input data (bits)	PSNR (dB) of whole image	PSNR (dB) of two blocks	Average PSNR (dB) after block division
16	61.5491	63.1307 61.8244	62.477
48	61.5447	63.1275 61.8205	62.474
80	61.5411	63.1239 61.8170	62.4704
128	61.5523	63.1186 61.8143	62.4664
160	61.5287	63.1155 61.8112	62.4633
320	61.5110	63.0874 61.8112	62.4493
500	61.4883	63.0832 61.7846	62.4339
1000	61.4232	63.0451 61.7513	62.398
5000	61.1835	62.6816 61.4451	62.063

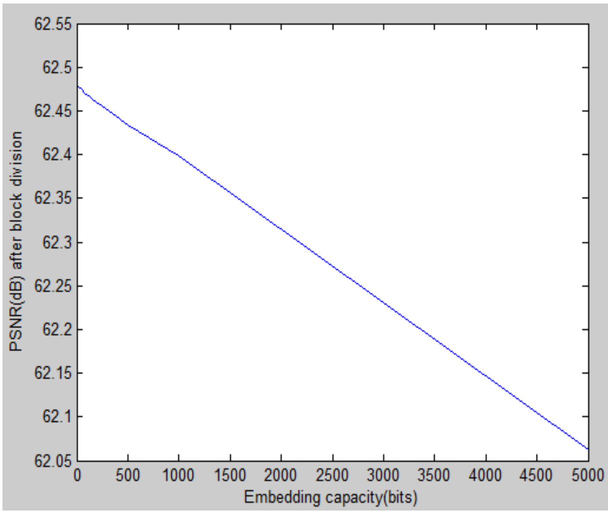

Fig. 8. Variation of PSNR(dB) versus embedding capacity of dataset 3.

IV. CONCLUSION

In this paper an improved reversible data hiding that is based on difference pair mapping and two-dimensional difference histogram have been proposed. The data is embedded by modifying the pixel value. The two-dimensional difference histogram will better exploit spatial redundancy. The proposed method based on block division of the image more data can be embedded. The experimental results show that this method provides good marked image quality.

REFERENCES

[1] Xiaolong Li, Weiming Zhang, and Bin Yang, "A Novel Reversible Data Hiding Based on Two-Dimensional Difference-Histogram Modification," *IEEE Transcations on Information Forensics and Security*, vol. 8, no. 7, July 2013.

[2] P. H. Pawar, K. C. Jondhale, "Histogram – Based Reversible Data Hiding Using Block Division," *IEEE International Conference on Advanced Communication Control and Computing Technologies*, pp. 295-299, 2012

[3] Xiaolong Li, Bin Yang, and Tieyong Zeng, "Efficient Reversible Watermarking Based on Adaptive Prediction-Error Expansion and Pixel Selection," *IEEE Transactions on image processing*, vol. 20, no. 12, December 2011.

[4] De-Sheng Fu, Zheng-Jun Jing, Song-Gang Zhao and Jing Fana, "Reversible data hiding based on prediction-error histogram shifting and EMD mechanism," *International Journal of Electronics and Communications* , pp. 933-943, 2006.

[5] Zhicheng Ni, Yun-Qing Shi, Nirwan Ansari and Wei Su, "Reversible Data Embedding," *lEEE Transactions on Circuits and Systems for Video Technology*, vol. 16, no.3, 2006.

[6] Masaaki Fujiyoshi, "A Histogram Shifting- Based Blind Reversible Data Hiding Method with a Histogram Peak Estimator," *International Symposium on Communications and Information Technologies*, pp. 333-318, 2012.

[7] S. K. Lee, Y. H. Suh, and Y. S. Ho, "Reversible image authenticationbased on watermarking," in *Proc. IEEE ICME*, 2006, pp. 1321–1324.

[8] W. L. Tai, C. M. Yeh, and C. C. Chang, "Reversible data hiding basedon histogram modification of pixel differences," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 19, no. 6, pp. 906–910, Jun. 2009.

[9] W. Hong, "Adaptive reversible data hiding method based on error energy control and histogram shifting," *Opt. Commun.*, vol. 285, no. 2, pp. 101–108, 2012.