

## Reversible Data Hiding Based on Pairwise Prediction-Error Histogram

JU-YUAN HSIAO, ZHI-YANG LIN AND PO-YUEH CHEN\*

*Department of Computer Science and Information Engineering*

*National Changhua University of Education*

*Changhua, 500 Taiwan*

*\*E-mail: pychen@cc.ncue.edu.tw*

Reversible data hiding (RDH) can extract secret messages and restore the original image without distortion. The reversibility benefits many practical applications such as medical image processing and multimedia archive management. Because of high image quality, histogram modification is applied to RDH in many literatures. In this paper, two-dimensional histogram and prediction-error expansion are integrated with a well-designed difference-pair mapping (DPM) to improve embedding capacity. According to the simulation results, the proposed RDH scheme outperforms existing approaches, which are also based on two-dimensional histogram. On the average, image quality is improved by 3dB with the same embedding capacity. Maintaining at the same image quality, embedding capacity can be improved by approximately 30,000 bits.

**Keywords:** prediction-error expansion (PEE), two-dimensional histogram, reversible data hiding (RDH), histogram modification, difference-pair mapping (DPM)

### 1. INTRODUCTION

Data hiding is a process which embeds specific information into cover media, such as images, audios or videos. In addition to secret communications, it can be used as invisible watermarking to verify the source of an image and avoid unauthorized usage as well. Using images as the cover, reversible data hiding (RDH) schemes can extract secret messages and restore the original image with no distortion. The reversibility benefits many practical applications such as medical image processing [1, 2], multimedia archive management [3], image trans-coding [4], and video error-concealment coding [5]. In general, the performance of a RDH scheme [6-22] is evaluated by its capacity-distortion behavior. For a required embedding capacity (EC), one may aim to minimize the embedding distortion and maintain a satisfactory image quality.

In 2006, Ni [6] proposed using histograms, the frequency distribution of pixel values, for reversible data hiding. Because one pixel value is modified by 1 at most, histogram-based schemes exhibit high image quality after embedding. Many histogram-based schemes have been proposed [7-24] for RDH. In 2013, Wang [7] adopted prediction-errors of a pixel-pair to construct a two-dimensional histogram. Applying a two-dimensional histogram, more peak points are available for data embedding and more directions are available for pixel-shifting as well. Therefore, the embedding capacity is significantly increased while maintaining a similar image quality [8-10].

In this study, by considering a pixel-pair and its neighboring pixels, local image blocks are projected onto a two-dimensional space of prediction-error pairs. Based on the resulting prediction-error pairs, a two-dimensional histogram is generated by calculating the frequency of occurrence. On this two-dimensional histogram, a specifically designed

Received March 1, 2016; revised May 24 & July 6, 2016; accepted July 12, 2016.

Communicated by Jing-Ming Guo.

difference-pair mapping (DPM) is adopted to implement reversible data embedding. According to the statistics obtained from 6 frequently used cover images, we design a novel DPM, which is a natural extension of expansion and shifting, to significantly increase the embedding capacity. According to the simulation results, the proposed RDH scheme outperforms current approaches [8-10], in terms of capacity and image quality.

The rest of the paper is organized as follows. The related works are briefly reviewed in Section 2. Section 3 presents the proposed RDH scheme in detail. Simulation results and performance comparison are demonstrated in Section 4. Finally, Section 5 concludes this paper.

## 2. RELATED WORKS

In this section, conventional PEE used for one-dimensional prediction-error histogram (PEH) is reviewed in subsection 2.1. The superiority of two-dimensional PEH is explained in subsection 2.2. Subsection 2.3 illustrates the mechanism of applying pairwise PEE to a two-dimensional histogram.

### 2.1 Conventional PEE

The first step of traditional PEE is to calculate  $\bar{x}$ , the predicted value of a pixel, for each pixel. For the current pixel  $x_i$  in queue  $(x_1, x_2, \dots, x_N)$  of the original image, the prediction-error is obtained by:

$$e_i = x_i - \bar{x}_i. \quad (1)$$

Based on a well-designed prediction, the resulting prediction-error sequence  $(e_1, e_2, \dots, e_N)$  will be extremely concentrated in the center of its histogram, which is defined by:

$$h(k) = \#\{1 \leq i \leq N: e_i = k\} \quad (2)$$

where  $\#$  denotes the cardinal number of a set. In most cases, highest frequency is  $h(0)$  and called a peak point, in where each pixel can be used to embed one secret bit. Data embedding is achieved by shifting values as follows:

$$e'_i = \begin{cases} 2e_i + b, & e_i \in [-T, T) \\ e_i + T, & e_i \in [-T, +\infty) \\ e_i - T, & e_i \in (-\infty, -T) \end{cases} \quad (3)$$

where  $T$  is a capacity-dependent integer-valued parameter and  $b \in \{0, 1\}$  is the to-be-embedded secret bit. When the prediction-error falls in  $[-T, +\infty)$ , it indicates that this pixel can be used to embed one bit. Otherwise, this pixel shifts outward without embedding. During the embedding process, the transformation of prediction-errors is demonstrated in Fig. 1, for two different threshold values.

### 2.2 Two-Dimensional PEH

In two-dimensional PEH, each pixel  $x_i$  requires two prediction values  $\bar{x}_i$  and  $\bar{y}$  to construct a prediction-error pair  $(e_{1i}, e_{2i})$ , using Eqs. (4) and (5).

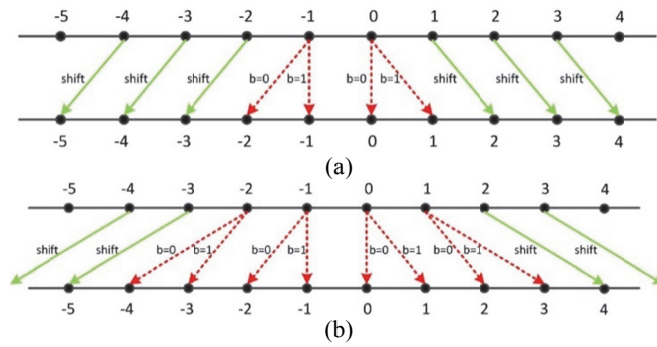


Fig. 1. Transformation of prediction-errors in conventional PEE (a)  $T=1$  (b)  $T=2$ .

$$e_{1i} = x_i - \bar{x}_i \tag{4}$$

$$e_{2i} = x_i - \bar{y}_i \tag{5}$$

Using  $e_{1i}$  and  $e_{2i}$  as the  $X$  and  $Y$  coordinates respectively, Wang [7] established a two-dimensional histogram according to these prediction-error pairs. As shown in Fig. 2, the resulting histogram was divided into several channels according to diagonals on the plane and the channel labels can be conveniently defined by Eq. (6).

$$e_{1i} - e_{2i} \tag{6}$$

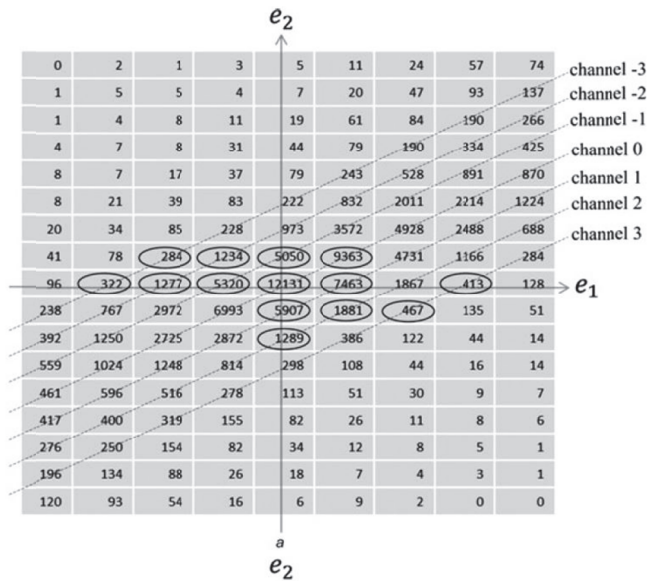


Fig. 2. Part of a practical histogram  $H(e_1, e_2)$  generated from Lena.

In each channel, two peak points with highest frequencies were selected for data embedding. Before data embedding, unselected points were preliminarily shifted outward using Eq. (7):

$$x_{i,j} = \begin{cases} x_{i,j} + 1, e_1 > p_r \\ x_{i,j} - b, e_1 < p_l \\ x_{i,j}, \text{otherwise} \end{cases} \quad (7)$$

where  $p_l$  and  $p_r$  denote the  $X$  coordinates of the left and right peak points in a channel, respectively. This preliminary shifting process creates empty points to embed secret bits into the selected peak points, using Eq. (8).

$$x_{i,j} = \begin{cases} x_{i,j} + b, e_1 = p_r \\ x_{i,j} - b, e_1 = p_l \\ x_{i,j}, \text{otherwise} \end{cases} \quad (8)$$

### 2.3 Pairwise Prediction

Pairwise prediction is a new concept for two-dimensional histogram applications. Single pixel prediction results in concentrated histogram and convenient embedding. However, the directions of shifting are limited to two, along the diagonals. Because of not being top 2 in a channel, many high-frequency points cannot be selected as a peak point. To efficiently use the high-frequency points on the histogram, we can remove the channel definition and set two pixels to be one pixel-pair. As a result, two pixel values can be modified simultaneously and 8 different shifting directions are feasible.

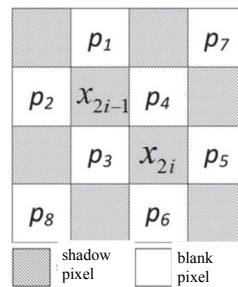


Fig. 3. Pixel referencing for the pairwise prediction defined in [10].

Employing pairwise prediction, Ou [10] proposed a new shifting scheme on a two-dimensional histogram. As shown in Fig. 3, pixels in the host image were first divided into two sets, denoted as “shadow” or “blank”. Shadow pixels were used to embed secret bits whereas blank pixels served as the references for prediction. For example, pixel-pair  $(x_{2i-1}, x_{2i})$  adopted blank pixels  $p_1$  to  $p_6$  as prediction references.

After prediction-error pair  $(x_{2i-1}, x_{2i})$  was computed for all pixels, a two-dimensional histogram can be established for shifting and embedding, as shown in Fig. 4. Nodes with a green arrow denote shifting-only points whereas nodes with a blue arrow and a red arrow can carry two bits and one bit, respectively. Peak points with highest frequencies, which gather around the origin of the plane, can all be selected as embedding points since there is no channel constraint.

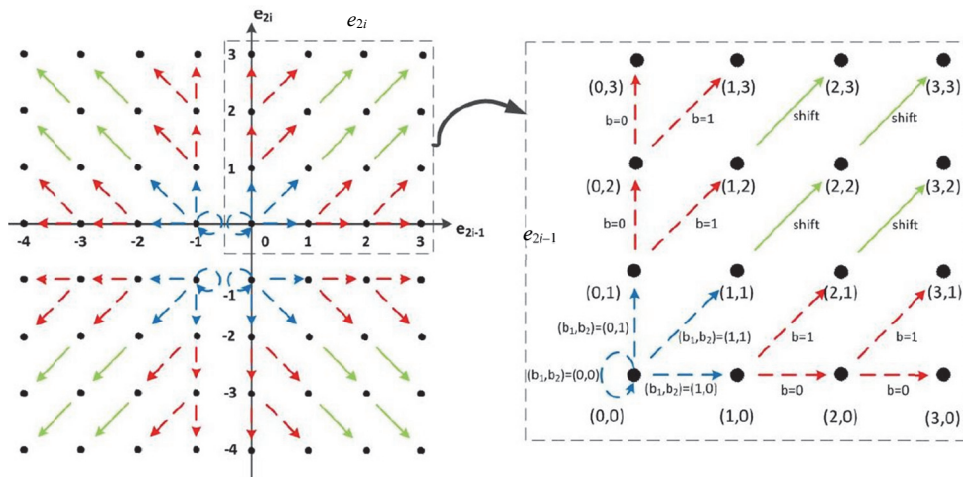


Fig. 4. A shifting mechanism of two-dimensional histogram based on pairwise prediction.

### 3. PROPOSED SCHEME

In this study, the authors proposed a novel RDH scheme based on pairwise prediction and two-dimensional histogram. A new pairwise prediction process is designed to further concentrate the histogram. According to the statistics obtained from 6 frequently used host images, Lena, Baboon, Pepper, Barbara, Airplane and Boat, the authors developed a novel difference-pair mapping (DPM) to improve the embedding capacity.

#### 3.1 Pairing and Prediction Design

Before pairing and prediction, a location map is generated to record the auxiliary information about which pixels have occurred overflow/underflow. To avoid overflow/underflow during shifting, pixel values of 255 and 0 are pre-modified to 254 and 1, respectively. Locations of the pre-modified pixels are recorded in the location map which serves as the prelude of embedding bit sequence. After extracting the location map and secret bits, the pre-modified pixels can be restituted accordingly.

The pixel pairing is performed horizontally in a zig-zag scan order. As shown in Fig. 5, a horizontal pair  $(X, Y)$  may adopt at most 10 references  $v1-v10$  to perform pairwise prediction. Based on such an arrangement, a parameter named Local Complexity (LC) is computed in advance for preliminarily screening.

	$j-1$	$j$	$j+1$	$j+2$
$j-1$	$v1$	$v2$	$v3$	$v4$
$j$	$v5$	$X$	$Y$	$v6$
$j+1$	$v7$	$v8$	$v9$	$v10$

Fig. 5. The pairing and referencing arrangement.

Taking the corners and edges into consideration, 4 cases of pairing position are described as follows.

**Case 1:** If the pixel-pair is at one of the corners, only 4 surrounding pixels (*e.g.* at the upper-left corner:  $v_6, v_8, v_9$  and  $v_{10}$ ) can be used as references and the LC is defined as:

$$LC = (|v_8 - v_9| + |v_9 - v_{10}| + |v_6 - v_{10}|) \times 6 \quad (9)$$

**Case 2:** If the pixel-pair is on the top or bottom edge, only 6 surrounding pixels (*e.g.* on the top edge:  $v_5 - v_{10}$ ) can be used as references and the LC is defined as:

$$LC = (|v_5 - v_6| + |v_7 - v_8| + |v_8 - v_9| + |v_9 - v_{10}| + |v_5 - v_7| + |v_6 - v_{10}|) \times 3 \quad (10)$$

**Case 3:** If the pixel-pair is on the left or right edge, only 7 surrounding pixels (*e.g.* on the left edge:  $v_2, v_3, v_4, v_6, v_8, v_9$  and  $v_{10}$ ) can be used as references and the LC is defined as:

$$LC = (|v_2 - v_3| + |v_3 - v_4| + |v_8 - v_9| + |v_9 - v_{10}| + |v_2 - v_8| + |v_3 - v_9| + |v_4 - v_6| + |v_6 - v_{10}|) \times 2 \quad (11)$$

**Case 4:** If the pixel-pair is in the center, all 10 surrounding pixels should be used as references and the LC is defined as:

$$LC = |v_1 - v_2| + |v_2 - v_3| + |v_3 - v_4| + |v_5 - v_6| + |v_7 - v_8| + |v_8 - v_9| + |v_9 - v_{10}| + |v_1 - v_5| \\ + |v_5 - v_7| + |v_2 - v_8| + |v_3 - v_9| + |v_4 - v_6| + |v_6 - v_{10}| \quad (12)$$

After obtaining the LC values of all pixel-pairs, users can set a threshold  $T$  to determine if a specific pair is suitable for embedding or not. In general, embedding data into a smooth region is recommended because it reduces the prediction errors and therefore concentrates the peak points around the origin of the prediction-error histogram plane. A large LC value indicates that the current pair is in a complex region and not suitable for embedding. On the other hand, a small value of LC indicates that the current pair is in a smooth region and selected for embedding. Before establishing the histogram, an appropriate threshold  $T$  is set to screen out pixel-pairs in complex regions, according to the user's embedding capacity demand. For example, if the capacity demand is large, the user can increase the value of threshold  $T$  and employ more pixels for embedding, at the expense of a lower image quality. According to the simulation results demonstrated in Section 4, it is natural to apply larger thresholds in images with complicated background. Specifically, in our experiments  $T$  ranges from 12 to 29 for Lena and 45 to 199 for Baboon, respectively.

The surviving pixel-pairs, mostly in smooth regions, are adopted for prediction-error computation. Similarly, 4 different cases should be considered as follows.

**Case 1:** If the pixel-pair is at the upper-left corner, Eq. (13) (or a similar equation for the other 3 corners) is adopted to compute the prediction-error pair  $(D_1, D_2)$ :

$$\begin{cases} D_1 = X - \left[ \frac{v_8 * 3 + v_9}{4} \right] \\ D_2 = Y - \left[ \frac{(v_6 + v_9) * 2 + (v_8 + v_{10})}{6} \right] \end{cases} \quad (13)$$

**Case 2:** If the pixel-pair is on the top edge, Eq. (14) (or a similar equation for the bottom edge) is adopted to compute the prediction-error pair  $(D_1, D_2)$ :

$$\begin{cases} D_1 = X - \left[ \frac{(v5 + v8) * 2 + (v7 + v9)}{6} \right] \\ D_2 = Y - \left[ \frac{(v6 + v9) * 2 + (v8 + v10)}{6} \right] \end{cases} \quad (14)$$

**Case 3:** If the pixel-pair is on the left edge, Eq. (15) (or a similar equation for the right edge) is adopted to compute the prediction-error pair  $(D_1, D_2)$ :

$$\begin{cases} D_1 = X - \left[ \frac{(v2 + v8) * 2 + (v3 + v9)}{6} \right] \\ D_2 = Y - \left[ \frac{v3 + v6 + v9}{4} + \frac{v2 + v4 + v8 + v10}{16} \right] \end{cases} \quad (15)$$

**Case 4:** If the pixel-pair is in the center, Eq. (16) is adopted to compute the prediction-error pair  $(D_1, D_2)$ :

$$\begin{cases} D_1 = X - \left[ \frac{v2 + v5 + v8}{4} + \frac{v1 + v3 + v7 + v9}{16} \right] \\ D_2 = Y - \left[ \frac{v3 + v6 + v9}{4} + \frac{v2 + v4 + v8 + v10}{16} \right] \end{cases} \quad (16)$$

As indicated in Eqs. (13) to (16), the prediction adopts forms of various weights. Closer references are given higher weights whereas the other ones associate with lower weights. Consequently, the resulting prediction-error shrinks and the frequencies of center peak points increase. In other words, more pixels can carry secret bits, instead of shifting only.

### 3.2 The DPM Design

Six 512×512 gray images, Lena, Baboon, Pepper, Barbara, Airplane and Boat, are adopted as the host images in this study. Exploiting the pairing and prediction technique described in subsection A, a two-dimensional average histogram is obtained. As shown in Fig. 6, highest frequencies occur around the origin. For example, the highest frequency is at (0, 0) whereas the second and third ones are at (0, 1) and (1, 0) respectively.

4	2391	2059	2168	2390	5417	3836	3603	3194	2622
3	2689	2342	2622	3379	9701	6487	5737	4372	3325
2	2715	2715	3486	5376	17535	11259	8595	5601	3796
1	3455	3678	4971	8308	30526	17067	11096	6406	3823
0	4132	6244	11518	20919	74427	30341	16828	9413	5519
-1	2838	3985	6216	9678	21230	8148	5540	3945	3473
-2	2622	3556	4959	6164	1174	4627	3441	2518	2312
-3	2294	2793	3439	3717	6296	2848	2275	2039	2018
-4	1836	2100	2313	2380	3584	1815	1754	1680	1865
	-4	-3	-2	-1	0	1	2	3	4

Fig. 6. Center part of the average histogram of six images.

Based on such an extremely concentrated frequency distribution, a novel difference-pair mapping (DPM) is designed and illustrated in Figs. 7-10. In Fig. 7, nodes with a green arrow denote shifting without embedding. Around the origin, 25 peak points are selected for data embedding. The 22 peak points with blue arrows can carry one bit per pixel-pair, as shown in Fig. 8. Three specific peak points (0, 0), (0, 1) and (1, 0), which have much higher frequency than others, are associate with red arrows and can carry more than one bit per pixel-pair, as shown in Figs. 9 and 10.

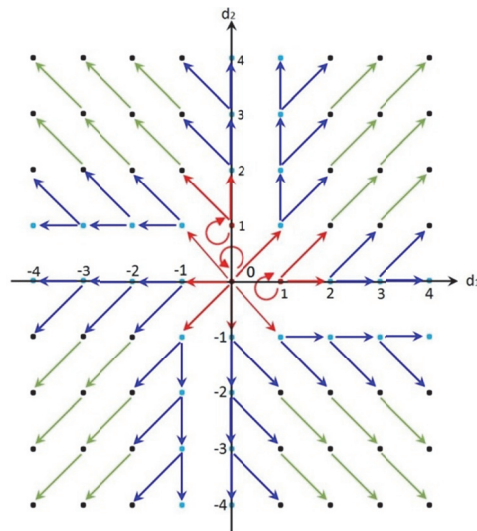


Fig. 7. The proposed DPM on a two-dimensional histogram.

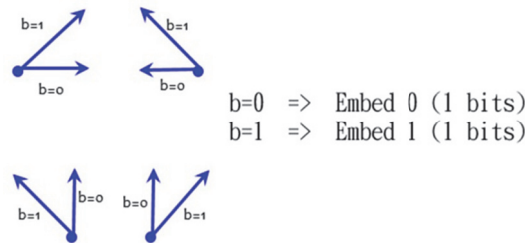


Fig. 8. Embedding and shifting rules for peak points with a blue arrow.

As shown in Fig. 9, each pixel-pair at (1, 0) or (0, 1) can carry one bit or two bits, depending on the two current secret bits. If the secret bits are 00 or 01, two secret bits are embedded successfully. If the secret bits are 10 or 11, only the first secret bit (“1”) is embedded and the remaining bit will be embedded at the next available pixel-pair. As shown in Fig. 10, each pixel-pair at (0, 0) can carry three or two bits, depending on the three current secret bits. Because no-shifting is possible for pixel-pairs at (1, 0) and (0, 1), only 6 of 8 outward arrows are available for pixel-pairs at (0, 0). Taking the loop (no shifting) into account, there are totally seven embedding rules for (0, 0). If the secret bits are 111 or 110, the lower-right shifting is adopted and only the first 2 bits (“11”) are embedded. The remaining bit will be embedded at the next available pixel-pair.



According to the average frequency distribution shown in Fig. 6 and the embedding rules shown in Fig. 4, Ou [10] embedded two bits per pixel-pair at (0, 0), (-1, 0), (-1, -1) and (0, -1) and resulted in a partial capacity of  $(74427+20919+9678+21230) \times 2 = 252508$  bits. Exploiting the embedding rules shown in Figs. 9 and 10, the proposed scheme embeds  $(3 \times 6 + 2 \times 2) / 8 = 2.75$  bits per pixel-pair and  $(2 \times 2 + 1 \times 2) / 4 = 1.5$  bits per pixel-pair at (0, 0) and  $\{(1, 0) \text{ or } (0, 1)\}$ , respectively. Therefore, the resulting partial capacity is  $74427 \times 2.75 + (30526 + 30341) \times 1.5 = 295974.75$  bits, which is significantly greater than 252508 bits. Specifically, the proposed DPM uses the center part of histogram more efficiently.

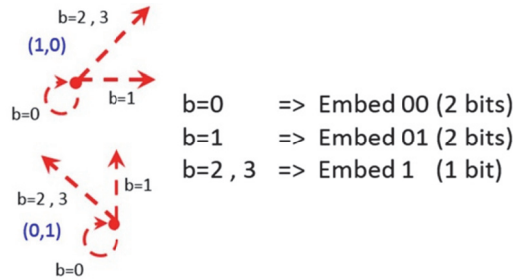


Fig. 9. Embedding and shifting rules for peak points (1, 0) and (0, 1).

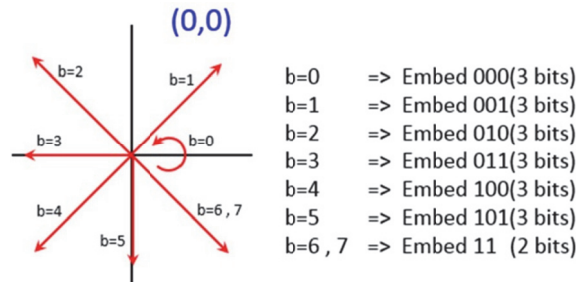


Fig. 10. Embedding and shifting rules for peak point (0, 0).

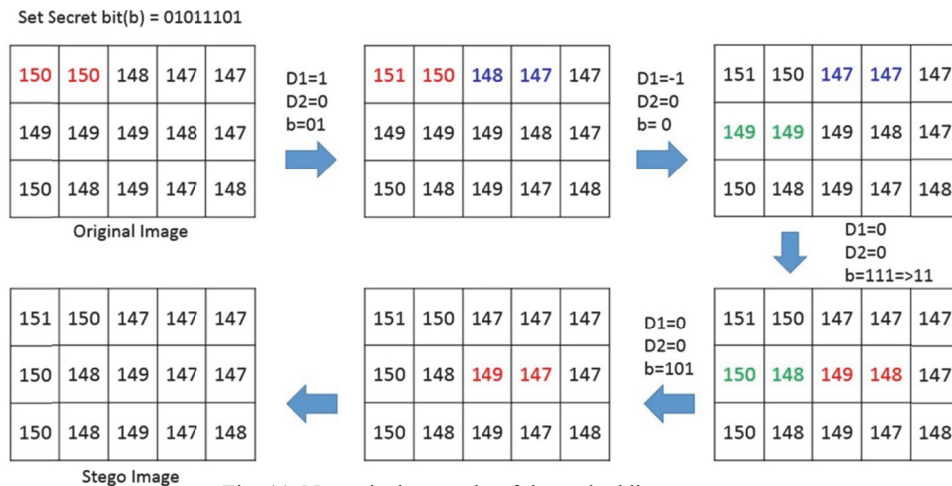


Fig. 11. Numerical example of the embedding process.

In Fig. 11, a numerical example is adopted to demonstrate the embedding process. In a zig-zag scan order, 4 pixel pairs are employed to embed 8 secret bits, 01011101. Because the first pair has a prediction error pair  $(D_1, D_2) = (1, 0)$ , the first 2 secret bits (01) are embedded successfully. According to the embedding rules listed in Fig. 9, the horizontal shifting from  $(1, 0)$  to  $(2, 0)$  is achieved by modifying  $(150, 150)$  to  $(151, 150)$ . Only one bit (0) is embedded in the second pair,  $(148, 147)$ , because the prediction error pair  $(D_1, D_2) = (-1, 0)$  and embedding rules in Fig. 8 apply. The horizontal shifting from  $(-1, 0)$  to  $(-2, 0)$  is achieved by modifying  $(148, 147)$  to  $(147, 147)$ . Although the third pixel pair has a prediction error pair  $(D_1, D_2) = (0, 0)$ , only 2 secret bits (11) are embedded where the 7th rule in Fig. 10 applies. The lower-right shifting from  $(0, 0)$  to  $(1, -1)$  is achieved by modifying  $(149, 149)$  to  $(150, 148)$ . Finally, the last pair can embed 3 secret bits (101) since it has a prediction error pair  $(D_1, D_2) = (0, 0)$  and the 6th rule in Fig. 10 applies. The vertical shifting from  $(0, 0)$  to  $(0, -1)$  is achieved by modifying  $(149, 148)$  to  $(149, 147)$ . To preserve the reversibility, the extraction process should be performed in a reversed zig-zag order. In other word, the last 3 secret bits (101) are extracted first and  $(149, 147)$  should be corrected back to  $(149, 148)$  before extracting the next 2 bits (11) from pair  $(150, 148)$ . In such a reversed zig-zag order, we can guarantee all reference pixels ( $v_i$ 's) retain their original values and reconstruct the original image without distortion. The details of extraction are similar to the embedding, just performed in a reversed order, and omitted here.

In summary, the proposed DMP takes advantage of the extremely concentrated distribution shown in Fig. 6 and increases the capacity effectively. In addition, searching for peak points is unnecessary because all peak points are stationary in the proposed scheme. Due to the same reason, no memory is required to record the auxiliary information about the locations of peak points and zero points. In other words, the proposed scheme provides much greater embedding capacity and reduces the required memory and computation as well.

#### 4. EXPERIMENTAL RESULTS

Six  $512 \times 512$  gray-scale images adopted as host are shown in Fig. 12. In this paper, Peak signal-to-noise ratio (PSNR), defined in Eq. (17), is adopted for image quality evaluation.



Fig. 12. Six sample images serving as host.

$$\begin{cases} MSE = \frac{1}{m * n} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} [I(i, j) - K(i, j)]^2 \\ PSNR = 10 * \log_{10}(\frac{MAX_I^2}{MSE}) \end{cases} \quad (17)$$

where *MSE* stands for Mean Square Error and  $m \times n$  is the image size.  $I(i, j)$  and  $K(i, j)$  denote pixel values of the original and modified images, respectively. *MAX<sub>I</sub>* denotes the maximum pixel value, which is 255 for grey level images. Table 1 shows that, except for Baboon, all host images can provide a large embedding capacity while maintaining at a satisfactory image quality. For example, Lena provides an embedding capacity of 120066 bits and a PSNR of 49.82 dB.

**Table 1. Image quality performance at various amounts of embedding.**

	5000	10000	20000	30000	50000	max
Lena	66.09	62.23	59.06	57.02	54.74	120066(49.82)
Baboon	61.69	58.16	54.52	51.44	×	36634(48.74)
Pepper	64.63	61.55	58.24	56.33	53.85	110205(49.64)
Barbara	63.93	61.17	58.07	56.09	53.47	72290(49.24)
Airplane	67.54	64.04	60.82	59.01	56.39	141713(49.74)
Boat	64.01	61.03	58.17	56.36	54.28	94143(49.67)

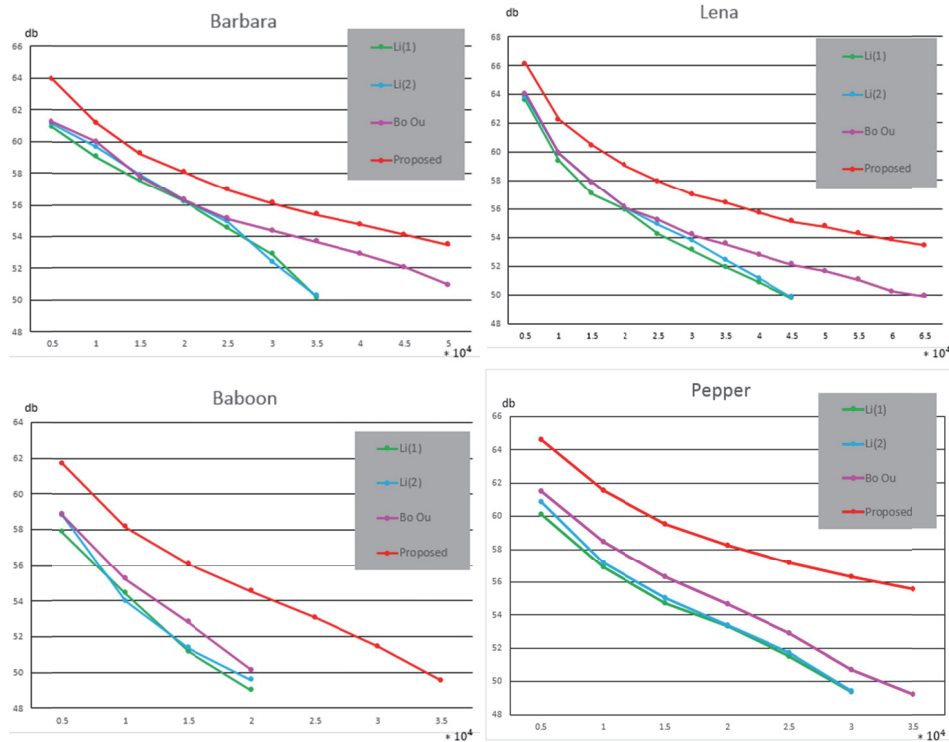


Fig. 13. Capacity-distortion performance of four 2D-histogram based approaches.

To fairly compare the proposed scheme with three existing 2D-histogram based approaches, Li (1) [8], Li (2) [9] and Ou [10], the capacity-distortion behaviors of all methods are exhibited in Fig. 13. The horizontal and vertical axes denote embedding capacity and PSNR, respectively. As shown in Fig. 13, the proposed scheme possesses superior capacity-distortion behavior, for all samples (results for the other two samples are similar and omitted). Specifically, using Lena as the host and fixing PSNR=54dB, the embedding capacity increases from 30,000 to 60,000 bits approximately. As shown in Tables 2 and 3, if a fixed embedding capacity is applied, the proposed scheme provides higher image quality than the other existing methods. On the average, image quality is improved by at least 2.2 dB and 3dB while the embedding capacity is fixed at 10,000 bits and 20,000 bits respectively.

As shown in Tables 2 and 3, if a fixed embedding capacity is applied, the proposed scheme provides higher image quality than the other existing methods. On the average, image quality is improved by at least 2.2 dB and 3dB while the embedding capacity is fixed at 10,000 bits and 20,000 bits respectively.

**Table 2. PSNR comparison (EC = 10,000 bits).**

	Xiaolong Li(1)	Xiaolong Li(2)	Bo Ou	Proposed scheme
Lena	59.37	59.78	59.75	62.23
Baboon	54.41	53.96	55.21	58.16
Pepper	56.89	57.19	58.49	61.55
Barbara	57.46	59.67	60.01	61.17
Airplane	62.23	63.17	63.77	64.04
Boat	57.15	57.43	57.56	61.03
<b>Average</b>	57.92	58.53	59.13	61.36

**Table 3. PSNR comparison (EC = 20,000 bits).**

	Xiaolong Li(1)	Xiaolong Li(2)	Bo Ou	Proposed scheme
Lena	55.93	56.15	56.29	59.06
Baboon	48.97	49.56	50.12	54.52
Pepper	53.31	53.39	54.64	58.24
Barbara	55.12	56.24	56.27	58.07
Airplane	59.26	59.45	60.20	60.82
Boat	53.05	53.12	53.34	58.17
<b>Average</b>	54.27	54.65	55.14	58.15

The weights employed in Eqs. (13)-(16) are set by instinct, based on a simple principle where closer surrounding pixels are assigned greater weights. To verify the weight setting process, we performed a counterpart experiment. Specifically, if Eq. (16) is replaced by Eq. (18) and the other 3 are reversed accordingly as well, the resulting PSNR is only slightly lower, as shown in Table 4. This is because that a large portion of surviving pixel pairs belongs to smooth regions where all surrounding pixels take the same (or similar) value as the center pixel. Therefore, the weight setting is actually not critical in practical applications. We also tried many other weight settings with various embedding capacity, comply-

ing or not complying the instinct principle, and the results are all similar.

The complexity of the proposed scheme is compared to that of three existing 2D-histogram based approaches, Li (1) [8], Li (2) [9] and Ou [10], in terms of processing time. All four approaches were implemented with the same notebook with a CPU of Intel i7-2670QM@2.2GHz and a RAM of DDR3 4GB×2. As shown in Table 5, the proposed scheme is slower but competitive to Li’s first approach. Compared to the other two, speed performance is significantly improved. The main reason is that the proposed approach establishes a 2-D error-pair histogram, not an ordinary pixel histogram which requires statistical and time-consuming computations.

$$\begin{cases} D_1 = X - \left[ \frac{v_2 + v_5 + v_8}{15} + \frac{v_1 + v_3 + v_7 + v_9}{5} \right] \\ D_2 = Y - \left[ \frac{v_3 + v_6 + v_9}{15} + \frac{v_2 + v_4 + v_8 + v_{10}}{5} \right] \end{cases} \quad (18)$$

**Table 4. PSNR(dB) under two weight setting approaches (capacity = 10000 bits).**

	Eqs. (13)-(16) applied	Reserved ones like Eq. (18) applied
Lena	62.23	61.90
Baboon	58.16	57.94
Pepper	61.55	61.44
Barbara	61.17	61.06
Airplane	64.04	63.35
Boat	61.03	60.75
<b>Average</b>	61.36	61.07

**Table 5. Processing time (ms) of three existing schemes and the proposed one.**

	Xiaolong Li(1)	Xiaolong Li(2)	Bo Ou	Proposed scheme
Lena	68	175	150	88
Baboon	83	193	164	100
Pepper	65	173	156	84
Barbara	64	164	153	86
<b>Average</b>	70	176.25	155.75	89.5

### 5. CONCLUSIONS

In this paper, an efficient RDH scheme based on pairwise PEE is proposed. The adopted pairwise PEE is a novel reversible mapping that exploits statistics generated from 6 frequently used host images. The proposed pairwise prediction is based on the fact that with high probability, a pixel takes a value similar to its adjacent pixels, especially in smooth regions. Such a flexibly weighting prediction effectively reduces the prediction-errors and makes the histogram further concentrated around the origin. As the peak values increase in general, total embedding capacity significantly increases as well. Based the proposed scheme, big data applications, such as multilayer data hiding and multimedia management, might be feasible because the image quality is still much higher

than 30dB, under which human vision cannot distinguish the differences. Similar statistic-based processes can be employed in various application fields. If the target host-images are in a specific image database, it is possible to design a more efficient DPM and more precise predicting equations for that database. The resulting quality-capacity performance might be even superior than those exhibited in Section 4.

## REFERENCES

1. G. Coatrieux, C. L. Guillou, J. M. Cauvin, and C. Roux, "Reversible watermarking for knowledge digest embedding and reliability control in medical images," *IEEE Transactions on Information Technology in Biomedicine*, Vol. 13, 2009, pp. 158-165.
2. M. Fontani, A. D. Rosa, R. Caldelli, F. Filippini, A. Piva, and M. Consalvo, "Reversible watermarking for image integrity verification in hierarchical PACS," in *Proceedings of the 12th ACM Workshop on Multimedia and Security*, 2010, pp. 161-168.
3. S. Lee, C. D. Yoo, and T. Kalker, "Reversible image watermarking based on integer-to-integer wavelet transform," *IEEE Transactions on Information Forensics and Security*, Vol. 2, 2007, pp. 321-330.
4. R. Li, O. C. Au, C. K. M. Yuk, S. Yip, and T. Chan, "Enhanced image trans-coding using reversible data hiding," in *Proceedings of IEEE International Symposium on Circuits and Systems*, 2007, pp. 1273-1276.
5. K.-L. Chung, Y.-H. Huang, P.-C., Chang, and H.-Y. Liao, "Reversible data hiding-based approach for intra-frame error concealment in H.264/AVC," *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 20, 2010, pp. 1643-1647.
6. Z. Ni, Y. Q. Shi, N. Ansari, and W. Su, "Reversible data hiding," *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 16, 2006, pp. 354-362.
7. S.-Y. Wang, C.-Y. Li, and W.-C. Kuo, "Reversible data hiding based on two-dimensional prediction errors," *IET Image Processing*, Vol. 7, 2013, pp. 805-816.
8. X. Li, B. Li, B. Yang, and T. Zeng, "General framework to histogram – Shifting-based reversible data hiding," *IEEE Transactions on Information Forensics and Security*, Vol. 22, 2013, pp. 2181-2191.
9. X. Li, W. Zhang, X. Gui, and B. Yang, "A novel reversible data hiding scheme based on two – Dimensional difference-histogram modification," *IEEE Transactions on Information Forensics and Security*, Vol. 8, 2013, pp. 1091-1110.
10. B. Ou, X. Li, Y. Zhao, R. Ni, and Y.-Q. Shi, "Pairwise prediction-error expansion for efficient reversible data hiding," *IEEE Transactions on Image Processing*, Vol. 22, 2013, pp. 5010-5021.
11. X. Li, J. Li, B. Li, and B. Yang, "High-fidelity reversible data hiding scheme based on pixel-value-ordering and prediction-error expansion," *Signal Processing*, Vol. 93, 2013, pp.198-205.
12. D.-S. Fu, Z.-J. Jing, S.-G. Zhao, and J. Fan, "Reversible data hiding based on prediction-error histogram shifting and EMD mechanism," *International Journal of Electronics and Communications*, Vol. 68, 2014, pp. 933-943.
13. I-C. Chang, Y.-C. Hu, W.-L. Chen, and C.-C. Lo, "High capacity reversible data hiding scheme based on residual histogram shifting for block truncation coding," *Signal Processing*, Vol. 108, 20145, pp. 376-388.

14. X. Gui, X. Li, and B. Yang, "Efficient reversible data hiding based on two-dimensional pixel-intensity-histogram modification," in *Proceedings of IEEE International Conference on Acoustic, Speech and Signal Processing*, 2014, pp. 7420-7424.
15. X.-T. Wang, C.-C. Chang, T.-S. Nguyen, and M.-C. Li, "Reversible data hiding for high quality images exploiting interpolation and direction order mechanism," *Digital Signal Processing*, Vol. 23, 2013, pp. 569-577.
16. T.-C. Lu, C.-Y. Tseng, and J.-H. Wu, "Dual imaging-based reversible hiding technique using LSB matching," *Signal Processing*, Vol. 108, 2015, pp. 77-89.
17. T.-C. Lu, C.-Y. Tseng, and K.-M. Deng, "Reversible data hiding using local edge sensing prediction methods and adaptive thresholds," *Signal Processing*, Vol. 104, 2014, pp. 152-166.
18. W. Zhang, X. Hu, X. Li, and N. Yu, "Recursive histogram modification: Establishing equivalency between reversible data hiding and lossless data compression," *IEEE Transactions on Image Processing*, Vol. 22, 2013, pp. 2775-2785.
19. T.-C. Lu, J.-H. Wu, and C.-C. Huang, "Dual-image-based reversible data hiding method using center folding strategy," *Signal Processing*, Vol. 115, 2015, pp. 195-213.
20. X. Ma, Z. Pan, S. Hu, and L. Wang, "High-fidelity reversible data hiding scheme based on multi-predictor sorting and selecting mechanism," *Journal of Visual Communication and Image Representation*, Vol. 28, 2015, pp. 71-82.
21. X. Hu, W. Zhang, X. Li, and N. Yu, "Minimum rate prediction and optimized histograms modification for reversible data hiding," *IEEE Transactions on Information Forensics and Security*, Vol. 10, 2015, pp. 653-664.
22. W. Zhang, X. Hu, X. Li, and Y. Nenghai, "Optimal transition probability of reversible data hiding for general distortion metrics and its applications," *IEEE Transactions on Image Processing*, Vol. 24, 2015, pp. 294-304.



**Ju-Yuan Hsiao (蕭如淵)** received his B.S. degree in Computer Engineering in 1986 from the National Chiao Tung University, Hsinchu, Taiwan and the M.S. degree in Applied Mathematic in 1988 from the National Chung Hsing University, Taichung, Taiwan. He received his Ph.D. degree in Computer Science and Engineering in 1992 from the National Tsing Hua University, Hsinchu, Taiwan. During the academic years of 1995-2000, he was on the faculty of the Department of Business Education at the National Changhua University of Education, Changhua, Taiwan. From 2000-2003, he was on the faculty of the Department of Information Management at the National Changhua University of Education, Changhua, Taiwan. Since August 2003, he has worked as a Professor in the Department of Computer Science and Information Engineering at the National Changhua University, Changhua, Taiwan. Dr. Hsiao's research interests include information security, algorithm design, and image processing.



**Zhi-Yang Lin (林志陽)** received his M.S. degree in CS from National Changhua University of Education. Mr. Lin is currently an Engineer in AU Optronics Corporation, Taiwan. His research interests include digital image processing and watermarking.



**Po-Yueh Chen (陳伯岳)** received his B.S., M.S., and Ph.D. degrees in EE from National Taiwan University in 1988, National Chiao Tung University in 1990 and University of Maryland at College Park in 1997 respectively. Currently Dr. Chen is a faculty member in Department of Computer Science and Information Engineering, National Changhua University of Education, Changhua, Taiwan. His research interests include image processing, FPGA implementations, VLSI architecture design and digital IC design.