The Journal of Systems and Software 86 (2013) 567–580

Contents lists available at SciVerse ScienceDirect

The Journal of Systems and Software

journal homepage: www.elsevier.com/locate/jss

Efficient reversible data hiding algorithm based on gradient-based edge direction prediction

Wei-Jen Yang[a,*], Kuo-Liang Chung[a,1], Hong-Yuan Mark Liao[b], Wen-Kuang Yu[a]

[a] Department of Computer Science and Information Engineering, National Taiwan University of Science and Technology, No. 43, Section 4, Keelung Road, Taipei 10672, Taiwan, ROC
[b] Institute of Information Science, Academia Sinica, No. 128, Section 2, Academia Road, Taipei 11529, Taiwan, ROC

ARTICLE INFO

Article history:
Received 12 November 2011
Received in revised form 21 June 2012
Accepted 27 September 2012
Available online 23 October 2012

Keywords:
Difference expansion
Edge direction
Embedding capacity
Marked image quality
Prediction
Reversible data hiding

ABSTRACT

In this paper, we present an efficient RDH algorithm based on a new gradient-based edge direction prediction (GEDP) scheme. Since the proposed GEDP scheme can generate more accurate prediction results, the prediction errors tend to form a sharper Laplacian distribution. Therefore, the proposed algorithm can guarantee larger embedding capacity and produce better quality of marked images. The determination of appropriate thresholds is also a critical issue for a RDH algorithm, so we design a new systematic way to tackle this problem. In addition, a modified embedding order determination strategy is presented to reduce the distortion of a marked image. Based on typical test images, experimental results demonstrate the superior properties of the proposed algorithm in terms of embedding capacity and marked image quality.

1. Introduction

Reversible data hiding (RDH) techniques can embed hidden data in a host image as well as allow the recovery of the original image without any distortion after extracting the hidden data (Honsinger et al., 2001). They are widely applied to the field of sensitive images, such as military, medical, and art work images, since the complete reconstruction of original images is required. For the RDH issue, the two most important measures are embedding capacity and quality degradation of a marked image. Therefore, a successful RDH algorithm not only can achieve large embedding capacity, but also minimize the distortion introduced by the embedding process.

Previously, many RDH methods were developed. Vleeschouwer et al. (2003) presented an RDH algorithm for media asset management based on circular interpretation of bijective transformations. Tian (2003) presented an RDH algorithm based on an integer Haar wavelet transform (Mallat, 1999) and difference expansion, and Alattar (2004) developed an RHD algorithm using the difference expansion on vectors formed by successive pixels. Kamstra and Heijmans' algorithm (Kamstra and Heijmans, 2005) fixed the image distortion problem that is existing in Tian's algorithm (Tian, 2003). Tsai et al. (2005) proposed an RDH algorithm for binary images. Ni et al. (2006) and Chang et al. (2006) developed RDH algorithms based on the peak-valley pairs of an image histogram and the outcome of side match vector quantization, respectively. Chang and Lu (2006) presented an RHD algorithm, which uses the indices of the codewords to embed hidden data, for side match vector quantization-compressed images. For joint photographic experts group (JPEG) images, Chang et al.'s RHD algorithm (Chang et al., 2007) embedded the hiding data in the medium-frequency part of the quantized discrete cosine transformation (DCT) coefficients. For vector quantization-compressed images, Chang and Lin (2007) presented an RHD algorithm based on a de-clustering strategy. Sachnev et al. (2007) enhanced the embedding capacity of Alattar's algorithm (Alattar, 2004) by exploiting quad pixels to embed hidden data and simplifying the location map. Using LOCO-I predictor (Weinberger et al., 1996), Thodi and Rodriguez (2007) presented a high capacity RDH algorithm based on the concept of prediction error expansion. For block truncation coding-compressed color images, Chang et al. (2008) developed an RHD algorithm utilizing the common bitmap to embed hidden data. Kim et al. (2008) presented a novel difference expansion transform to improve the capacity and quality performance of Kamstra and Heijmans'

* Corresponding author.
E-mail addresses: wjyang@mail.ntust.edu.tw (W.-J. Yang), klchung01@gmail.com (K.-L. Chung).
[1] Supported by the National Science Council of the R. O. C. under contract NSC98-2221-E-011-102-MY3..

0164-1212/$ – see front matter © 2012 Elsevier Inc. All rights reserved.
http://dx.doi.org/10.1016/j.jss.2012.09.041

algorithm (Kamstra and Heijmans, 2005). Lin et al. (2008) presented a multilevel RHD algorithm that modifies the difference image histogram and uses the peak value to embed hidden data. Chang et al. (2009) developed an RHD algorithm based on the joint neighboring coding technique for images compressed by vector quantization. Tai et al. (2009) presented an improved RHD algorithm to solve the problem of communicating multiple peak points to recipients in Lin *et al.*'s algorithm (Lin et al., 2008). Without using the threshold, Lin et al. (2010) enhanced the quality performance of Alattar's algorithm (Alattar, 2004) by embedding hidden data in the smooth areas determined through the proposed quad of quads structure. Sachnev et al. (2009) presented an efficient RDH algorithm by combining the sorting and error prediction concepts. For Chinese character data, Wang et al. (2009) developed an RDH algorithm using left-right and up-down Chinese character representation. Based on the interpolation technique, Luo et al.'s RHD algorithm (Luo et al., 2010) used the differences between interpolation values and actual pixel values to embed hidden data. Hwang et al. (2010) proposed a histogram shifting-based RHD algorithm exploiting the diamond prediction scheme and sorting strategy. Based on the spectral-spatial correlation in the color difference domain (Chung et al., 2008; Pei and Tam, 2003), Yang et al. (2012) developed the first RHD algorithm designed specifically for color filter array mosaic images.

After examining the previously developed RDH algorithms using the prediction errors and difference expansion, we know that the embedding capacity and marked image quality depend on the prediction scheme employed in the RDH algorithm. In this paper, we develop an improved RDH algorithm based on a new gradient-based edge direction prediction (GEDP) scheme. We try to model the prediction errors as a sharper Laplacian distribution. Therefore, the proposed algorithm can achieve larger embedding capacity and produce better quality of marked images. Since the determination of appropriate thresholds is also a critical issue for an RDH algorithm, we design a new systematic way to tackle this problem. In addition, we propose a modified version of the embedding order strategy which is better than one proposed by Sachnev et al. (2009) to improve the outcome. Eighteen images are used to evaluate the related performance and the results indicate that the proposed RDH algorithm is superior to four existing RHD algorithms, namely Tai et al.'s algorithm (Tai et al., 2009), Thodi and Rodriguez's algorithm (Thodi and Rodriguez, 2007), Luo et al.'s algorithm (Luo et al., 2010), and Sachnev et al.'s algorithm (Sachnev et al., 2009). We compare the proposed RHD algorithm with the above four RHD algorithms due to the following two reasons. First, like the proposed algorithm, the four compared algorithms are based on the concepts of prediction error expansion and histogram modification. Second, they are regarded as the state-of-the-art RHD algorithms. Note that since all the RDH algorithms mentioned above are discussed in an attack-free environment, out work follows the same environment assumption.

The three main contributions of this work are as follows. First, we develop a new GEDP scheme to reduce the prediction errors, and it is the most crucial factor that influences the performance of an RDH algorithm. Second, we design a new systematic way to determine the appropriate thresholds which can provide enough usable capacity to embed hidden data and some overheads as well as generate the best quality of a marked image. Finally, a modified embedding order determination strategy is proposed to reduce the distortion of a marked image.

The rest of this paper is organized as follows. In Section 2, a brief review to the four state-of-the-art prediction error-based RHD algorithms is given. In Section 3, we present the proposed GEDP scheme and discuss the Laplacian distribution of the prediction errors. In Section 4, we describe the proposed RDH algorithm. Section 5 reports the experimental results to demonstrate the advantages of our RDH algorithm. Finally, concluding remarks are drawn in Section 6.

2. Previous works

Before presenting the proposed GEDP scheme and RHD algorithm, in this section, we briefly review the four state-of-the-art prediction error-based RHD algorithms proposed by Tai et al. (2009), Thodi and Rodriguez (2007), Luo et al. (2010), and Sachnev et al. (2009), respectively. The four RHD algorithms are based on the concepts of prediction error expansion and histogram modification. They utilize prediction schemes to predict the pixel values and embed hidden data in the image by modifying the corresponding prediction errors.

Tai et al.'s RHD algorithm (Tai et al., 2009) scans an image by an inverse s-order and uses the last scanned pixel value as the prediction value of the current pixel. After constructing the histogram formed by the prediction errors, the hidden data are embedded by the histogram modification. Without using the location map, Tai et al.'s algorithm directly contracts the histogram from both sides to ensure that the embedding hidden data does not cause the overflow and underflow problems. Their algorithm also utilizes a binary tree structure to resolve the problem of communicating multiple peak points to recipients, which is the major drawback in Lin et al.'s algorithm (Lin et al., 2008).

Thodi and Rodriguez (2007) first utilizes the LOCO-I predictor (Weinberger et al., 1996), which is based on the mutual relation in the neighborhood of a pixel, to predict the value of each pixel. Then, they embed the hidden data in an image by using the combination of the prediction error expansion and the histogram shifting scheme. Furthermore, instead of using the location map, a two-pass testing method with a flag bit stream is presented to resolve the overflow or underflow problem, leading to enhancing the embedding capacity.

In Luo et al.'s RHD algorithm (Luo et al., 2010), pixel values are predicted by using the interpolation scheme. Then, the pixels with prediction errors falling into the two highest peaks of the histogram are exploited to embed hidden data by using the additive error expansion. To prevent the overflow or underflow problem, Luo et al.'s RHD algorithm only uses the pixels whose values are within the range of 1 through 254 to embed the hidden data, and it utilizes a boundary map to resolve ambiguous problem when embedding a hidden bit in the pixel with value 1 or 254.

Sachnev et al.'s algorithm (Sachnev et al., 2009) predicts pixel values by averaging the gray values of the four neighboring pixels in a rhombus shape. The hidden data are embedded in the pixels whose corresponding prediction errors are within the range of the two thresholds, T_n and T_p, by using the difference expansion. Furthermore, in order to enhance the marked image quality performance, the shifting scheme by Thodi and Rodriguez (2007) is utilized and the embedding order is determined by a sorting strategy based on the local variances calculated by the differences between neighboring pixel pairs. According to our experiments, Sachnev et al.'s algorithm can embed more data with less distortion when compared with existing RDH algorithms.

Since the embedding capacity and marked image quality of the above prediction error-based RDH algorithms greatly depend on the employed prediction scheme, we can enhance the embedding and quality performance of an RHD algorithm by improving the prediction accuracy. Therefore, in what follows, we first present a more accurate GEDP scheme to predict the pixel values, and then based on the GEDP scheme, the proposed RDH algorithm with larger embedding capacity and better quality of marked images is developed.

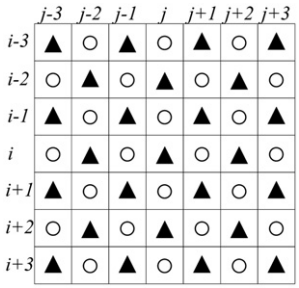

Fig. 1. Depiction of pixels in Ψ_α and Ψ_β, where ▲ denotes Ψ_α and ○ denotes Ψ_β.

3. Proposed GEDP scheme and Laplacian distribution of prediction errors

In Section 3.1, we first present the proposed GEDP scheme to calculate the prediction errors, and then in Section 3.2, we discuss the Laplacian distribution of the prediction errors and its potential embedding benefits.

3.1. Proposed GEDP scheme

For ease of explanation, the gray value of the pixel located at position (i, j) of an input image is denoted as $x_{i,j}$, and we use Fig. 1 to present the proposed GEDP scheme. As shown in Fig. 1, all the pixels are partitioned into two sets marked by symbols "▲" and "○", respectively. We denote the two sets as Ψ_α and Ψ_β, respectively. Since the pixels in Ψ_α and Ψ_β disjoint to each other, we can predict the gray values in Ψ_α by using those in Ψ_β and vice versa. Since the proposed GEDP scheme for the two sets is the same, we only discuss the case of Ψ_α.

Note that we partition the pixels into two sets due to the following reasons. Since the pixels in the two sets disjoint to each other, embedding hidden data in the pixels in one set would not affect those in the other one. Therefore, the extraction process can first reobtain the precise predicted gray values of the pixels in Ψ_α from the pixels in Ψ_β and vice versa. Then, the reobtained predicted gray values are utilized to assist in extracting the hidden data and recovering the original gray values. Furthermore, in the proposed RHD algorithm, the embedding order of the pixels in Ψ_α is determined by the pixels in Ψ_β; conversely, the embedding order of the pixels in Ψ_β is determined by the pixels in Ψ_α. Due to the mutual independence of the pixels in Ψ_α and Ψ_β, the original embedding order can be maintained after the embedding process.

From Fig. 1, it is obvious that the predicted gray value of the central pixel at position (i, j) can be determined by its four neighboring pixels with movement $\Phi = \{(i \pm 1, j), (i, j \pm 1)\}$. Note that the pixels on the boundary of the image are dealt with using the mirroring method. To better predict the gray value, instead of predicting such a value by averaging the gray values of the four neighboring pixels (Sachnev et al., 2009), we assign four proper weights based on gradient information and edge direction to the four corresponding neighboring pixels. The proposed gray value prediction method is called GEDP scheme. The proposed GEDP scheme can predict gray values more accurately, and it would result in larger embedding capacity and better marked image quality.

To keep the independence between the pixels in Ψ_α and Ψ_β, the proposed GEDP scheme adopts two hybrid mask pairs which are the combination of Sobel masks and interpolation masks (abbreviated as SI mask pairs), as shown in Fig. 2, to extract the gradient information. The detailed derivations of the two SI mask pairs are shown in Appendix A. After running the proper SI masks on a 5×5 subimage centered at position (m, n), the horizontal gradient response $\Delta h_{m,n}$ and the vertical gradient response $\Delta v_{m,n}$ can be obtained.

Considering the neighboring pixel located at position $(i-1, j)$, if the magnitude of the vertical gradient response is large, i.e., there is a horizontal edge passing through it, it indicates that this pixel should make less contribution to predict the gray value of the central pixel; otherwise, it should make more contribution. Therefore, the weight of the pixel at position $(i-1, j)$ can be determined by $w_{i-1,j} = (1/(1 + |\Delta v_{i,j}| + 2|\Delta v_{i-1,j}| + |\Delta v_{i-2,j}|))$. Similarly, the weights of the other three neighbors can be determined by $w_{i+1,j} = (1/(1 + \sum_{k=0}^{2} \delta_k |\Delta v_{i+k,j}|))$, $w_{i,j-1} = (1/(1 + \sum_{k=0}^{2} \delta_k |\Delta h_{i,j-k}|))$, and $w_{i,j+1} = (1/(1 + \sum_{k=0}^{2} \delta_k |\Delta h_{i,j+k}|))$, where $\delta_k = 2$ if $k = 1$; otherwise $\delta_k = 1$. Based on the four weights, the predicted gray value $p_{i,j}$ can be determined by

$$p_{i,j} = \Big(\sum_{(u,v)\in\Phi} w_{u,v}\Big)^{-1} \sum_{(u,v)\in\Phi} w_{u,v} x_{u,v}, \tag{1}$$

where $\Phi = \{(i \pm 1, j), (i, j \pm 1)\}$. Further, we can calculate the prediction error $e_{i,j}$ at position (i, j) by

$$e_{i,j} = x_{i,j} - p_{i,j}. \tag{2}$$

3.2. Laplacian distribution of prediction errors and its potential embedding benefits

Since $x_{i,j}$ and $p_{i,j}$ in Eq. (2) are usually very close to each other, $e_{i,j}$ is close to zero and the prediction errors with zero mean tend to follow a Laplacian distribution (Parzen, 1960):

$$P(e) = \frac{\sqrt{2}}{2\sigma_e} \exp\left(-\frac{\sqrt{2}|e|}{\sigma_e}\right), \tag{3}$$

where the random variable e denotes the prediction error ignoring the position parameter; σ_e is the standard deviation of the prediction errors. At present, we discuss the potential embedding benefits of the proposed RDH algorithm using the GEDP scheme from the Laplacian distribution. The performance of a prediction error-based RDH algorithm has a strong relation with the shape of the corresponding Laplacian distribution histogram. The sharper the Laplacian distribution histogram, i.e., more prediction errors closing zero and smaller variance, the better will be the RDH algorithm's data embedding performance (Sachnev et al., 2009).

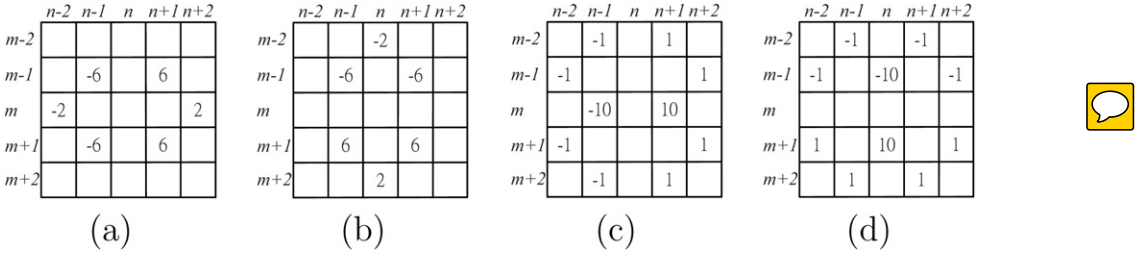

Fig. 2. Two SI mask pairs. For the pixels at position $(m, n) \in \Psi_\alpha$, (a) the horizontal mask and (b) the vertical mask. For the pixels at position $(m, n) \notin \Psi_\alpha$, (c) the horizontal mask and (d) the vertical mask.

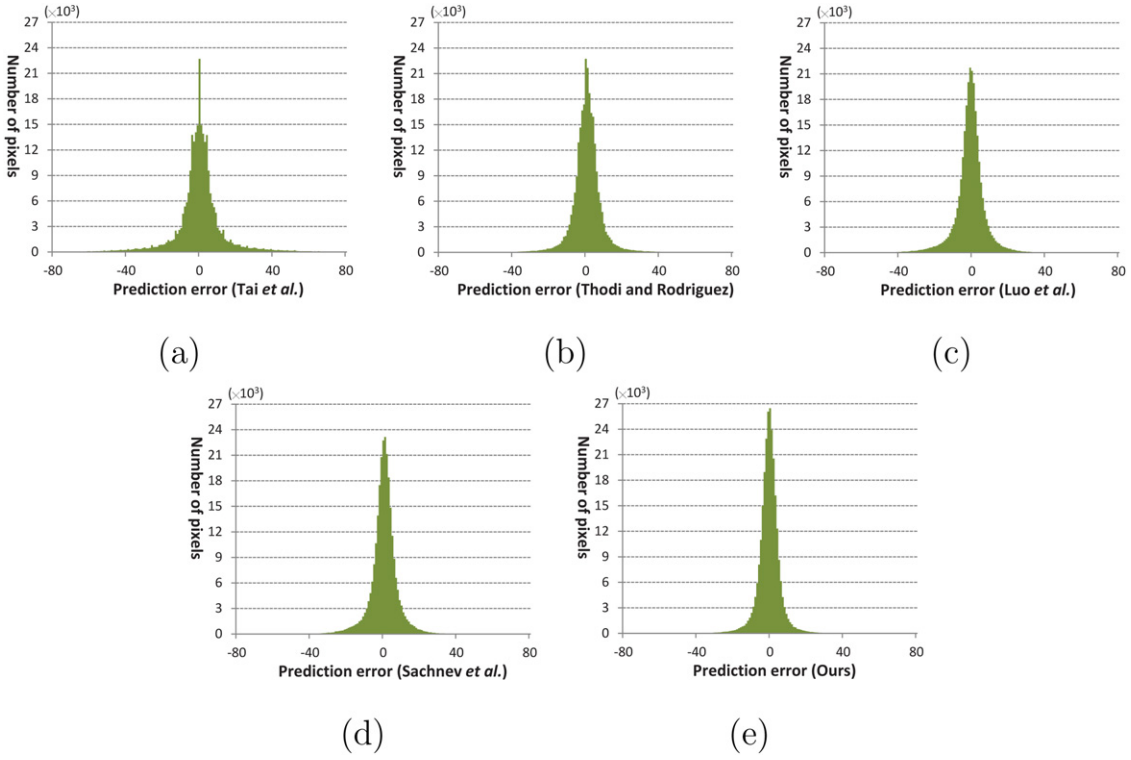

Fig. 3. For the Couple image, the corresponding histograms of the prediction errors obtained from (a) Tai et al.'s scheme, (b) Thodi and Rodriguez's scheme, (c) Luo et al.'s scheme, (d) Sachnev et al.'s scheme, and (e) the proposed scheme.

We compare our algorithm with four RDH algorithms proposed by Tai et al. (2009), Thodi and Rodriguez (2007), Luo et al. (2010), and Sachnev et al. (2009), respectively. Based on the Couple image, Fig. 3(a)–(e) illustrates the five histograms of the prediction errors obtained from the four prediction schemes employed in compared RDH algorithms and the proposed GEDP scheme, respectively. In addition, the standard deviations and peak values of the above five histograms are shown in Table 1. Clearly, the histogram in Fig. 3(e) is sharper than any one in Fig. 3(a)–(d); and Table 1 also demonstrates that the histogram obtained from the proposed GEDP scheme has the smallest standard deviation and the highest peak value. Furthermore, for the standard deviations and peak values in Table 1, the related improvements and the corresponding improvement ratios of the proposed GEDP scheme over four concerned prediction schemes are shown in the parentheses. From the table, it is obvious that the proposed GEDP scheme produces more than 16.16% standard deviation reduction ratio and 14.28% peak value increase ratio, implying the proposed GEDP scheme can achieve a more accurate prediction and result in smaller prediction errors. It leads to significant quality and embedding capacity benefits of the proposed RDH algorithm. Experimental results in Section 5 will confirm the two benefits of the proposed RDH algorithm.

4. Proposed RHD algorithm

In Section 4.1, we first describe the embedding and extraction strategies. In Section 4.2, we propose a modified version of the embedding order strategy to improve the outcome; and in Section 4.3, we design a new systematic way to determine the appropriate threshold values, that can guide the data hiding process. Finally, the whole procedure of the proposed RDH algorithm is presented in Section 4.4.

4.1. Embedding and extraction strategies

After obtaining the prediction error $e_{i,j}$ by the proposed GEDP scheme mentioned in Section 3.1, the hidden bit h can be embedded in the pixel $x_{i,j}$ via the difference expansion technique. To reduce the overall distortion, we only use the pixels whose prediction errors are within the range $[T_n, T_p]$ to embed the hidden data. Thus, the prediction error $e_{i,j}$ can be modified based on the following rule:

$$e'_{i,j} = \begin{cases} e_{i,j} + T_p + 1 & \text{if } e_{i,j} > T_p \\ e_{i,j} + T_n & \text{if } e_{i,j} < T_n \\ 2e_{i,j} + h & \text{otherwise,} \end{cases} \tag{4}$$

where $h \in \{0, 1\}$ is the current scanned hidden bit; T_p (≥ 0) and T_n (≤ 0) are two thresholds. In Section 4.3, we shall show a new systematic way to determine the appropriate values of T_p and T_n. After embedding the hidden data, the original pixel $x_{i,j}$ is perturbed to the marked pixel $x'_{i,j}$ as follows:

$$x'_{i,j} = e'_{i,j} + p_{i,j}. \tag{5}$$

We now provide a theoretical analysis of the proposed scheme in terms of the expected value of the embedding distortion. According

Table 1
The standard deviations and peak values of the five histograms in Fig. 3.

Scheme	Standard deviation	Peak value
Tai et al. (2009)	14.501 (8.228; 56.74%)	22,701 (3754; 16.54%)
Thodi and Rodriguez (2007)	8.345 (2.072; 24.83%)	22,732 (3723; 16.38%)
Luo et al. (2010)	8.443 (2.170; 25.70%)	21,731 (4724; 21.74%)
Sachnev et al. (2009)	7.482 (1.209; 16.16%)	23,149 (3306; 14.28%)
The proposed	6.273	26,455

to the previous explanation in Section 3.2, we model the prediction errors as a Laplacian distribution with zero mean. The probability of a specific prediction error can be calculated by Eq. (3). From Eq. (4) and (5), the embedding distortion $\mathfrak{D}(e)$ can be obtained by

$$\mathfrak{D}(e) = \begin{cases} T_p + 1 & \text{if } e > T_p \\ -T_n & \text{if } e < T_n \\ |e + h| & \text{otherwise,} \end{cases} \tag{6}$$

where $h \in \{0, 1\}$ and e denotes the prediction error ignoring the position parameter. We assume that the two events, $h = 0$ and $h = 1$, are equiprobable. Consequently, the expected value of the embedding distortion $E(\mathfrak{D}(e))$ can be calculated by

$$E(\mathfrak{D}(e)) = (T_p + 1)\sum_{e=T_p+1}^{255} P(e) - T_n \sum_{e=-255}^{T_n-1} P(e) + \frac{1}{2}\left[\sum_{e=T_n}^{T_p} P(e)|e| + \sum_{e=T_n}^{T_p} P(e)|e+1|\right], \tag{7}$$

where the probability density function $P(e)$ has been defined in Eq. (3). From Eq. (7), it is clear that the embedding distortion is strongly dependent on the distribution of prediction errors. Table 1 has shown that the standard deviation of the Laplacian distribution in our scheme is smallest and it implies that the Laplacian distribution obtained by our scheme is the sharpest. Accordingly, for a fixed threshold pair, $(T_p; T_n)$, the value of $E(\mathfrak{D}(e))$ in our proposed scheme is the smallest. Therefore, the proposed RDH algorithm may achieve the best quality of marked images. Experimental results will confirm this argument.

The extracting strategy is the inverse of the embedding strategy. Since the pixels in Ψ_α and Ψ_β disjoint to each other, embedding hidden data in the pixels in Ψ_α would not affect those in Ψ_β and vice versa. From Fig. 1 and Eq. (1), it is clear that the central pixel $x_{i,j}$ is in Ψ_α, but the predicted gray value $p_{i,j}$ is determined by the pixels in Ψ_β. Therefore, we can reobtain the precise predicted gray value $p_{i,j}$ by Eq. (1) at the extraction side.

Given a marked pixel $x'_{i,j}$, the modified can be calculated prediction error $e'_{i,j}$ by

$$e'_{i,j} = x'_{i,j} - p_{i,j}, \tag{8}$$

From $e'_{i,j}$, the hidden bit h can be extracted by

$$h = e'_{i,j} \bmod 2 \quad \text{if } e'_{i,j} \in [2T_n, 2T_p + 1], \tag{9}$$

and the original prediction error $e_{i,j}$ can be obtained by

$$e_{i,j} = \begin{cases} e'_{i,j} - T_p - 1 & \text{if } e'_{i,j} > 2T_p + 1 \\ e'_{i,j} - T_n & \text{if } e'_{i,j} < 2T_n \\ \lfloor e'_{i,j}/2 \rfloor & \text{otherwise.} \end{cases} \tag{10}$$

Finally, recovery the original gray value $x_{i,j}$ by

$$x_{i,j} = e_{i,j} + p_{i,j}. \tag{11}$$

Note that using prediction errors to embed hidden data may cause an overflow or underflow problem for some pixels. We must resolve the problem before executing the embedding or extraction processes. The two-pass testing method utilized in previous algorithms (Thodi and Rodriguez, 2007; Sachnev et al., 2009) with an extra flag bit stream can be used to resolve the problem.

4.2. Modified embedding order determination strategy

Instead of using embedding order determination strategy by Sachnev et al. (2009), in this sub-section, we present a modified embedding order determination strategy, which directly uses local variances formed by the neighboring gray values of targeted pixels, to determine the embedding order. The proposed modified embedding order determination strategy can achieve less distortion of a marked image.

It is believed that embedding hidden data in pixels which have smaller prediction errors would cause less distortion. To reduce the distortion of a marked image, the embedding order of the targeted pixels needs be changed (Kamstra and Heijmans, 2005; Sachnev et al., 2009). Since a pixel located in a homogeneous area usually implies that its prediction error is small, we let such a pixel have a higher priority for embedding a hidden bit. In Sachnev et al.'s algorithm (Sachnev et al., 2009), the embedding order is determined by using sorting strategy according to their corresponding local variances formed by the differences between the neighboring pixel pairs. However, when a pixel is located in a nonhomogeneous area, i.e., the four differences between neighboring pixel pairs are large, its corresponding local variance may be small. Thus, the pixel would have a higher priority for embedding a hidden bit, but it would cause larger distortion.

To resolve the above priority mistake problem, we directly use the local variances formed by the neighboring gray values of targeted pixels in Ψ_α to determine the embedding order. Given a pixel $x_{i,j}$, its local variance $\sigma^2_{i,j}$ can be calculated by

$$\sigma^2_{i,j} = \frac{1}{4}\sum_{(u,v)\in\Phi} [x_{u,v} - \bar{x}_{i,j}]^2, \tag{12}$$

where $\bar{x}_{i,j} = (1/4)\sum_{(u,v)\in\Phi} x_{u,v}$ and $\Phi = \{(i \pm 1, j), (i, j \pm 1)\}$. Consequently, a pixel with small local variance indicates that it is located in a homogeneous area, so the embedding order of the pixels in Ψ_α is based on their local variances sorted in ascending order.

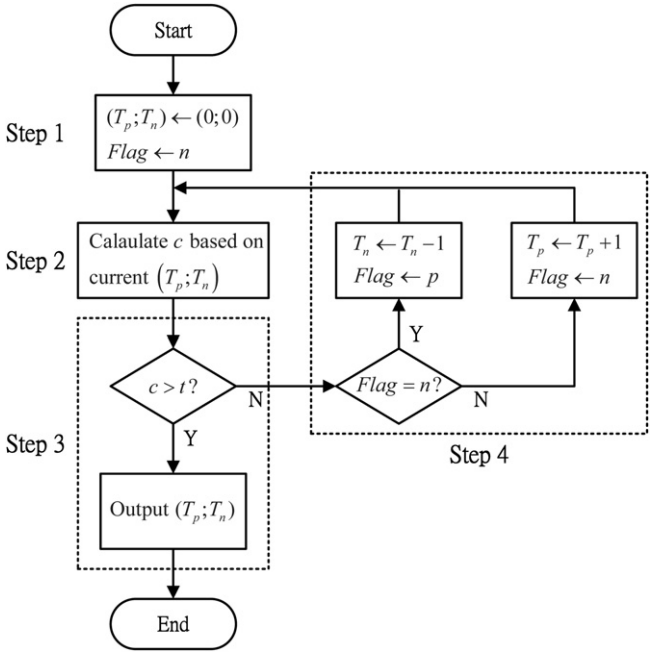

Fig. 4. The flowchart of the $(T_p; T_n)$-determination process.

Table 2
The expected value of the distortion caused by embedding a hidden bit under different prediction errors.

Prediction error e	0	−1	1	−2	2	−3	3	−4	4	−5	5	⋯
Expected value	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{3}{2}$	$\frac{3}{2}$	$\frac{5}{2}$	$\frac{5}{2}$	$\frac{7}{2}$	$\frac{7}{2}$	$\frac{9}{2}$	$\frac{9}{2}$	$\frac{11}{2}$	⋯

One thing to be noted is that embedding data in the pixels in Ψ_α would not affect the ones in Ψ_β because the pixels in Ψ_α and Ψ_β disjoint to each other. Further, the embedding order of the pixels in Ψ_α is determined by the pixels in Ψ_β, and vice versa. Therefore, the original sorted order can be maintained after the embedding process.

4.3. New systematic way to determine the appropriate threshold values

The embedding capacity and the quality of marked images are affected by two thresholds, T_p and T_n. Thus, it is important to determine the related values of T_p and T_n that can provide

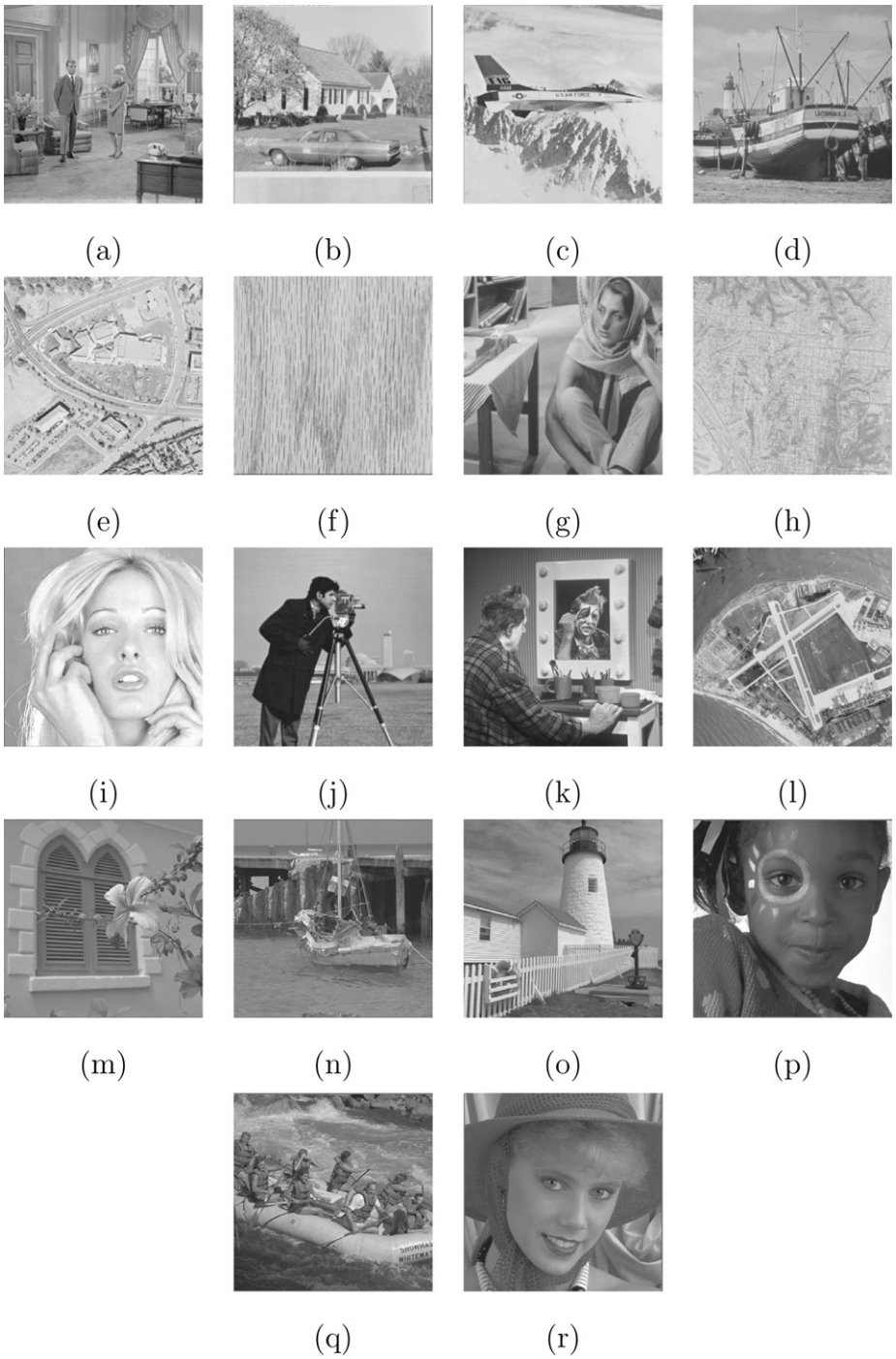

Fig. 5. Eighteen test images: (a) Couple, (b) House, (c) Jet, (d) Boat, (e) Street, (f) Wood, (g) Barbara, (h) Map, (i) Tiffany, (j) Cameraman, (k) Payaso, (l) Harbor, (m) Window, (n)Sailboat, (o) Lighthouse, (p) Child, (q) Boating, and (r) Woman.

enough usable capacity to embed the intended data as well as produce the best quality of a marked image. In Sachnev et al.'s algorithm (Sachnev et al., 2009), the precise approach to determine the appropriate threshold values is not discussed. At present, we propose a systematic process to determine T_p and T_n. Before determining the appropriate thresholds, the total number of bits of intended data should be known in advance. When embedding hidden data $\mathcal{P}$ in an image, besides $\mathcal{P}$, other overheads, such as a flag bit stream $\mathcal{F}$ and a header stream $\mathcal{H}$, should also be embedded. Thus, the total number of bits of intended data t can be calculated by $t = |\mathcal{P}| + |\mathcal{H}| + |\mathcal{F}|$, where $|\mathcal{P}|$, $|\mathcal{H}|$, and $|\mathcal{F}|$ denote the bit numbers of $\mathcal{P}$, $\mathcal{H}$, and $\mathcal{F}$, respectively. In addition, the header stream, which records size of the hidden data, the size of flag bit stream, and the values of two thresholds, is embedded in the first $|\mathcal{H}|$ pixels of an image by using the LSB replacement method (Tian, 2003). Under this circumstance, an extra correction bit stream with the length $|\mathcal{H}|$ is needed to record the LSB values replaced by the header in order to achieve the purpose of the reversible data hiding. Therefore, the embedding process starts from the $(\mathcal{H}+1)$th pixel.

To guarantee the best quality of a marked image, we analyze the embedding distortion caused by embedding a hidden bit h in a pixel.

Theorem 1. *The expected value of the distortion caused by embedding one hidden bit h in a pixel with the prediction error ε (≥ 0) or the one caused by embedding h in a pixel with the prediction error $-\varepsilon - 1$ is $\varepsilon + (1/2)$ when the two events, $h = 0$ and $h = 1$, are equiprobable.*

Proof. See Appendix B.

Based on Theorem 1, Table 2 shows the expected value of the distortion caused by embedding one hidden bit under different prediction errors. From the table, it is clear that using pixels with larger positive prediction errors and larger negative prediction errors to embed hidden data would result in serious quality

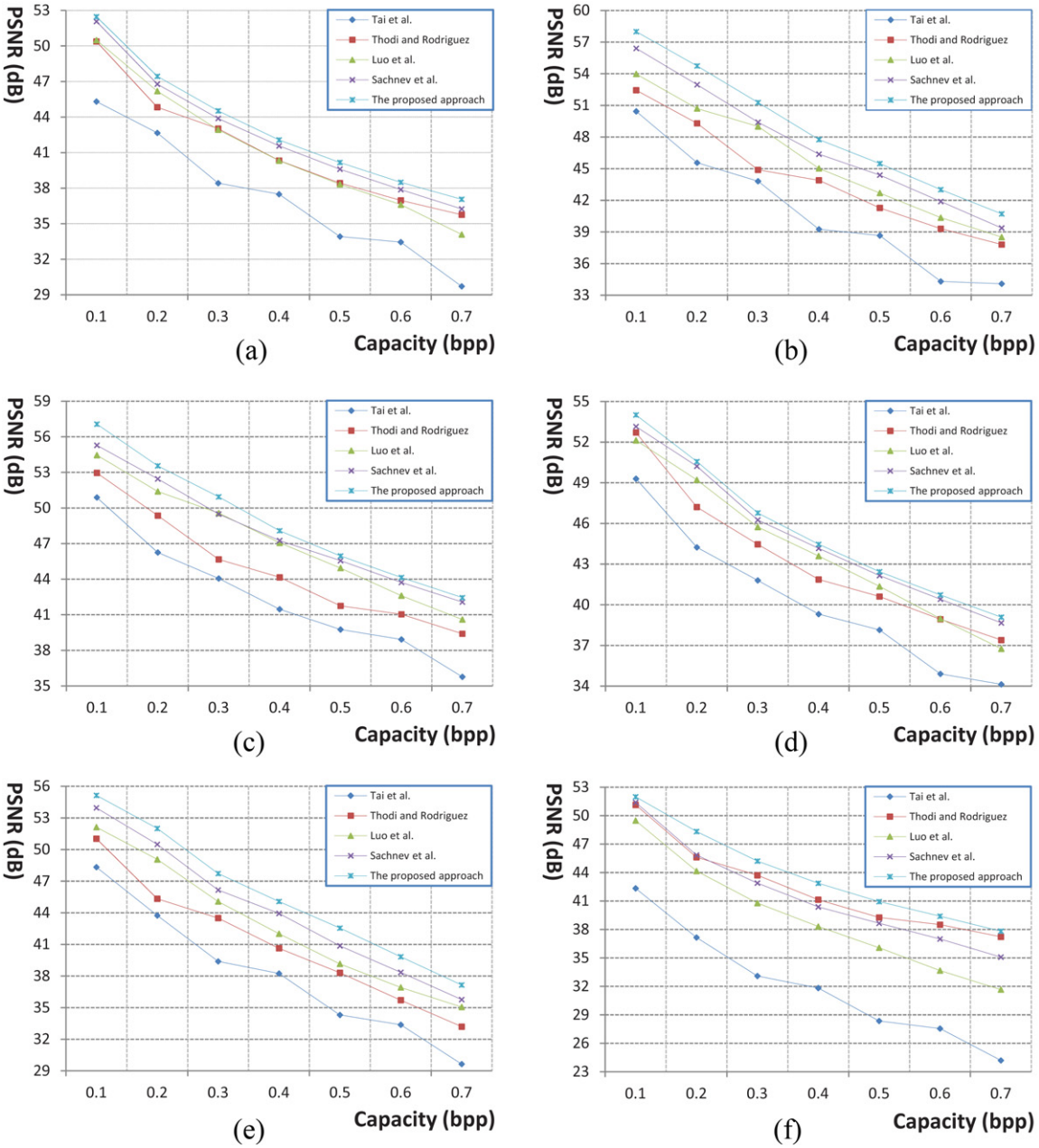

Fig. 6. The PSNR versus the Capacity curves of the five compared RDH algorithms for the test images: (a) Couple, (b) House, (c) Jet, (d) Boat, (e) Street, (f) Wood, (g) Barbara, (h) Map, (i) Tiffany, (j) Cameraman, (k) Payaso, (l) Harbor, (m) Window, (n) Sailboat, (o) Lighthouse, (p) Child, (q) Boating, and (r) Woman.

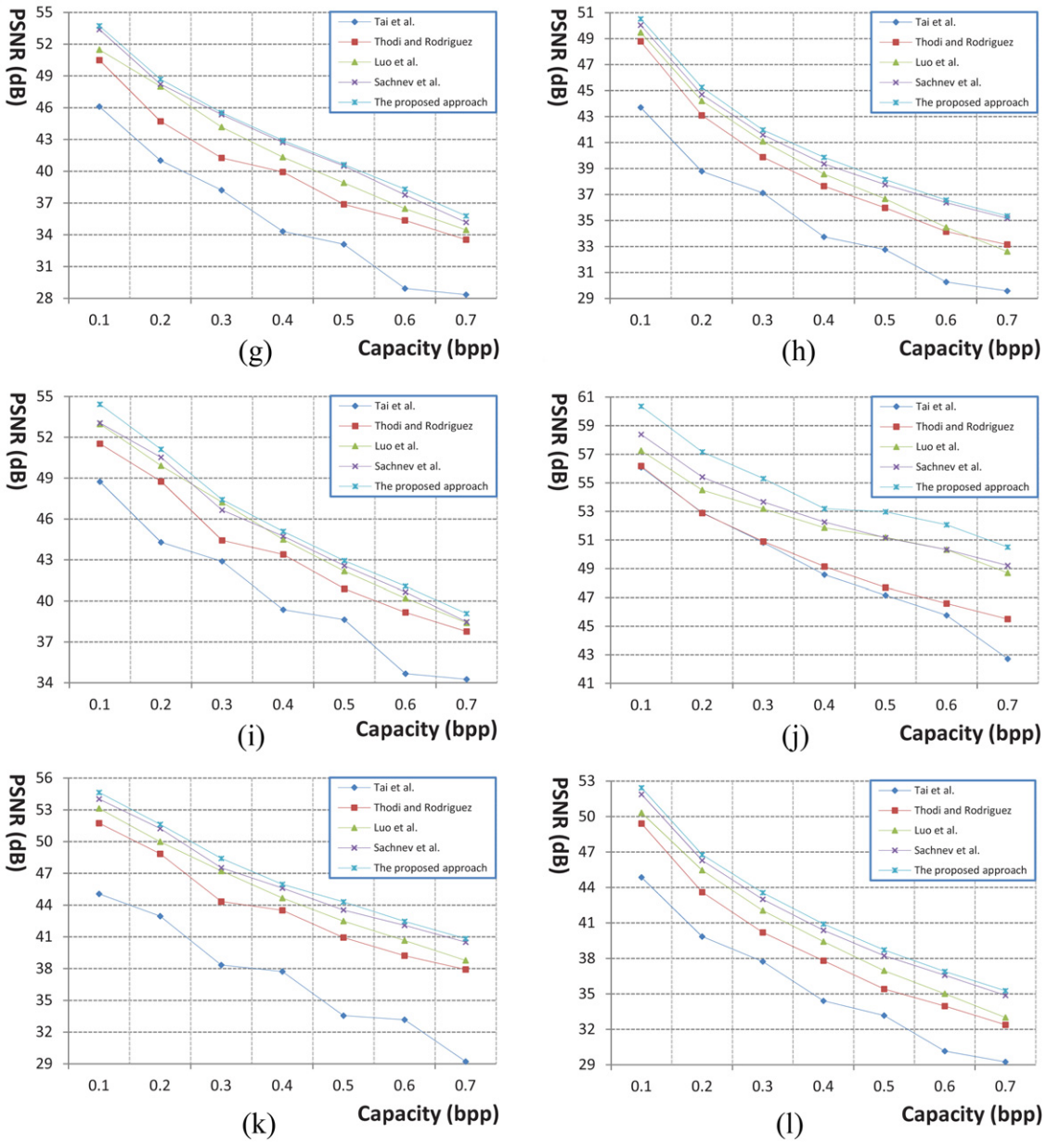

Fig. 6. (*Continued*).

degradation of marked images. Thus, the absolute values of the two thresholds should be as small as possible. In addition, when two prediction errors have the same absolute values, the distortion resulting from difference expansion embedding based on the negative one would be less than that based on the positive one. Hence, the negative prediction error has higher priority for embedding the hidden bit. When the threshold pair $(T_p; T_n)$, where $|T_n| = |T_p|$, cannot provide enough usable capacity, the new threshold pair $(T_p; T_n - 1)$ is prior to be selected.

According to the above analysis, the $(T_p; T_n)$-determination process consists of four steps:

Step 1: Set $(T_p; T_n) \leftarrow (0; 0)$ and $Flag \leftarrow n$.

Step 2: Based on the current $(T_p; T_n)$, calculate the usable capacity c provided by the input image.

Step 3: If the condition $c \geq t$ holds, where t is the total number of bits of intended data, then output $(T_p; T_n)$ as the appropriate threshold pair and stop; otherwise, go to Step 4.

Step 4: If the condition $Flag = n$ holds, perform the operation $T_n \leftarrow T_n - 1$ and set $Flag \leftarrow p$; otherwise, perform the operation $T_p \leftarrow T_p + 1$ and set $Flag \leftarrow n$. Then, go to Step 2.

Following the above process, the threshold pairs $(T_p; T_n)$'s will be set to $(0; 0), (0; -1), (1; -1), (1; -2)$, and so on. This process will continue on until the intended data can be embedded. The flowchart of the $(T_p; T_n)$-determination process is shown in Fig. 4.

For illustration, consider the case where the total number of bits of intended data is $t = 1024$ and the usable capacities based on $(T_p; T_n) = (0; 0)$ and $(T_p; T_n) = (0; -1)$ are $c_{(0;0)} = 754$ and $c_{(0;-1)} = 1278$, respectively. Following *Step 1*, we, respectively, set $(T_p; T_n)$ and *Flag* to be $(0; 0)$ and n, initially. The rationale behind $(T_p; T_n)$ and *Flag* is that the absolute values of the two thresholds should be as small as possible and the new threshold pair $(T_p; T_n - 1)$ is prior to be selected when $|T_n| = |T_p|$. According to *Step 2*, based on $(T_p; T_n) = (0; 0)$, we obtain $c_{(0;0)} = 654$.

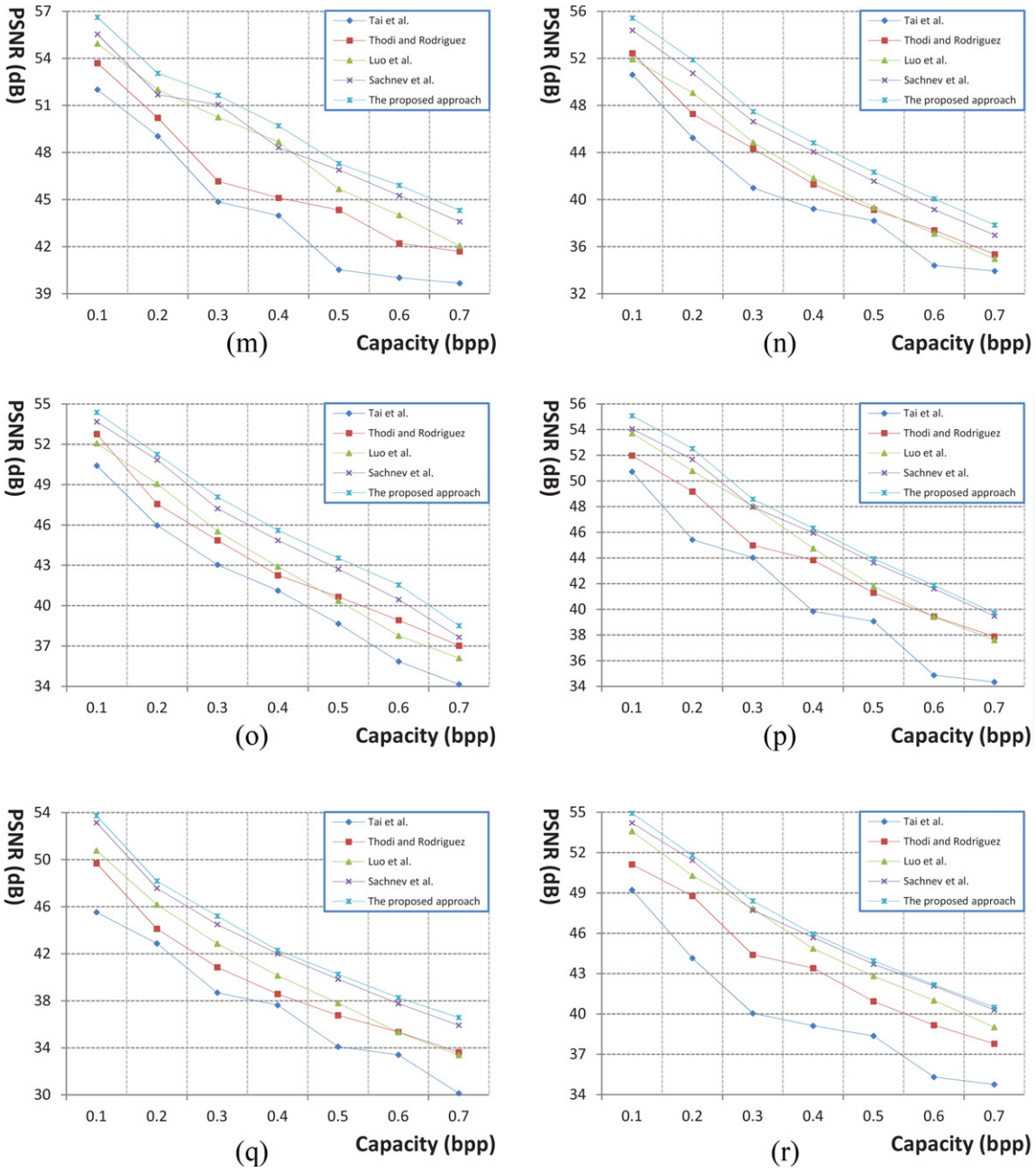

Fig. 6 (*Continued*).

Following *Step 3*, for $c_{(0;0)} < t$ which implies the usable capacity is insufficient for embedding the intended data, we thus cannot set $(0;0)$ as the appropriate threshold pair and should adjust $(T_p;T_n)$. From *Step 4*, for Flag $= n$, we perform $T_n \leftarrow T_n - 1 = -1$ and obtain $(T_p;T_n) = (0;-1)$; then we set Flag $\leftarrow p$. After adjusting $(T_p;T_n)$, we repeat *Step 2* and *Step 3* to examine whether the new $(T_p;T_n) = (0;-1)$ can provide enough usable capacity to embed the intended data. Thus, based on $(T_p;T_n) = (0;-1)$, we obtain $c_{(0;0)} = 1278$, and then for $c_{(0;-1)} \geq t$ which implies the usable capacity is sufficient for embedding the intended data, we output $(0;-1)$ as the appropriate threshold pair and stop.

4.4. Complete procedure of the proposed RDH algorithm

We now describe how the proposed RDH algorithm operates. Given an input image, we first divide all the pixels into two sets, Ψ_α and Ψ_β. Then, the hidden data are evenly separated into two parts. The embedding process requires that we first embed one part of hidden data in the pixels in Ψ_α, and then repeat the embedding process to embed the other part in the pixels in Ψ_β. The extraction process is the inverse of the embedding process. Since the embedding and extraction processes for the pixels in Ψ_α and Ψ_β are the same, we only show the case related to Ψ_α. Note that the size of the header stream is known in advance for both the embedding and extraction parts.

The embedding process for the pixels in Ψ_α involves six steps.

Step 1: Preserve the first $|\mathcal{H}|$ pixels in Ψ_α.
Step 2: From the $(|\mathcal{H}|+1)$th pixel to the last pixel, calculate the prediction error for each pixel by using the proposed GEDP scheme described in Section 3.1.

Table 3
Under the constraints of PSNR ≃ 50 dB, the comparison results in terms of the SHD (bit) for the eighteen test images.

Algorithm	Couple image		House image		Jet image	
	SHD	PSNR	SHD	PSNR	SHD	PSNR
Tai et al. (2009)	14,391	49.998	28,442	50.000	30,854	49.999
Thodi and Rodriguez (2007)	28,573	49.998	43,384	50.001	45,788	50.008
Luo et al. (2010)	29,108	50.004	64,230	50.000	72,172	50.005
Sachnev et al. (2009)	37,756	49.997	75,028	50.003	76,130	50.001
The proposed	41,346	50.001	86,696	50.001	86,880	50.001

Algorithm	Boat image		Street image		Wood image	
	SHD	PSNR	SHD	PSNR	SHD	PSNR
Tai et al. (2009)	17,642	50.003	16,462	50.004	5819	49.996
Thodi and Rodriguez (2007)	38,902	50.000	33,135	50.001	33,213	50.002
Luo et al. (2010)	46,144	49.997	42,788	50.004	22,942	50.003
Sachnev et al. (2009)	53,588	50.005	54,426	50.003	31,150	49.996
The proposed	55,896	50.002	62,526	50.001	44,046	50.002

Algorithm	Barbara image		Map image		Tiffany image	
	SHD	PSNR	SHD	PSNR	SHD	PSNR
Tai et al. (2009)	11,691	50.003	7885	49.999	17,904	49.999
Thodi and Rodriguez (2007)	28,023	50.001	19,870	50.002	37,879	50.003
Luo et al. (2010)	36,446	49.995	23,100	50.003	51,386	50.003
Sachnev et al. (2009)	44,728	50.000	26,406	49.999	56,892	50.003
The proposed	46,628	50.004	28,846	50.001	61,610	49.997

Algorithm	Cameraman image		Payaso image		Harbor image	
	SHD	PSNR	SHD	PSNR	SHD	PSNR
Tai et al. (2009)	89,653	49.999	11,324	49.998	10,171	49.999
Thodi and Rodriguez (2007)	93,768	50.002	37,172	50.003	23,383	49.998
Luo et al. (2010)	175,636	50.000	51,910	50.002	28,346	50.005
Sachnev et al. (2009)	166,200	49.997	62,920	49.996	34,794	50.004
The proposed	192,152	49.998	68,502	50.004	37,808	50.003

Algorithm	Window image		Sailboat image		Lighthouse image	
	SHD	PSNR	SHD	PSNR	SHD	PSNR
Tai et al. (2009)	38,875	50.000	28,232	50.003	27,433	49.998
Thodi and Rodriguez (2007)	55,705	50.002	37,355	49.996	41,051	50.004
Luo et al. (2010)	84,468	50.005	44,308	49.999	44,572	49.999
Sachnev et al. (2009)	87,850	50.000	56,630	50.004	53,634	49.993
The proposed	101,190	50.004	62,920	50.002	61,866	50.001

Algorithm	Child image		Boating image		Woman image	
	SHD	PSNR	SHD	PSNR	SHD	PSNR
Tai et al. (2009)	32,047	49.997	14,994	50.002	21,023	49.998
Thodi and Rodriguez (2007)	42,545	50.003	23,592	49.997	36,883	50.001
Luo et al. (2010)	61,688	50.004	29,998	50.002	55,346	50.004
Sachnev et al. (2009)	64,650	50.003	41,084	50.002	64,518	50.000
The proposed	72,120	49.996	45,384	50.004	68,450	50.003

Algorithm	Average SHD	SHD improvement ratio
Tai et al. (2009)	23,602.33	188.31%
Thodi and Rodriguez (2007)	38,901.17	74.93%
Luo et al. (2010)	53,588.22	26.98%
Sachnev et al. (2009)	60,465.78	12.54%
The proposed	680,48.11	

Step 3: Determine the embedding order by the modified embedding order determination strategy described in Section 4.2.

Step 4: Determine two appropriate thresholds by the new systematic way described in Section 4.3, and then get the flag bit stream.

Step 5: Use the LSB replacement method to embed the header stream in the first $|\mathcal{H}|$ pixels, and then obtain the correction bit stream.

Step 6: According to the embedding order of the pixels, embed the correction bit stream, the flag bit stream, and the hidden data in the pixels

Table 4
Under the constraints of PSNR ≃ 45 dB, the comparison results in terms of the SHD (bit) for the eighteen test images.

Algorithm	Couple image		House image		Jet image	
	SHD	PSNR	SHD	PSNR	SHD	PSNR
Tai et al. (2009)	27,918	45.005	58,720	44.999	66,319	44.990
Thodi and Rodriguez (2007)	50,478	45.007	75,864	44.999	87,372	44.999
Luo et al. (2010)	59,774	44.998	105,124	44.998	130,050	45.004
Sachnev et al. (2009)	69,026	45.005	121,792	45.003	137,890	44.998
The proposed	73,666	44.999	135,322	45.001	146,016	45.002

Algorithm	Boat image		Street image		Wood image	
	SHD	PSNR	SHD	PSNR	SHD	PSNR
Tai et al. (2009)	47,316	45.001	38,102	44.993	14,391	44.997
Thodi and Rodriguez (2007)	71,172	45.004	56,360	45.001	59,506	44.999
Luo et al. (2010)	87,038	45.004	79,330	45.004	46,930	45.003
Sachnev et al. (2009)	96,210	45.004	93,144	45.005	60,088	45.002
The proposed	98,858	45.002	105,332	45.002	80,484	45.002

Algorithm	Barbara image		Map image		Tiffany image	
	SHD	PSNR	SHD	PSNR	SHD	PSNR
Tai et al. (2009)	31,247	45.003	19,660	44.996	43,725	44.999
Thodi and Rodriguez (2007)	50,672	45.005	33,816	45.000	67,816	45.001
Luo et al. (2010)	70,888	45.002	46,956	45.005	98,832	44.997
Sachnev et al. (2009)	81,164	45.004	49,840	45.004	101,714	45.002
The proposed	82,528	44.999	55,058	45.000	105,700	44.997

Algorithm	Cameraman image		Payaso image		Harbor image	
	SHD	PSNR	SHD	PSNR	SHD	PSNR
Tai et al. (2009)	164,993	44.999	26,581	45.003	25,559	45.002
Thodi and Rodriguez (2007)	192,597	45.001	65,536	44.999	40,134	45.002
Luo et al. (2010)	237,498	45.001	101,452	44.999	55,660	45.005
Sachnev et al. (2009)	235,556	45.004	114,298	44.999	59,512	44.999
The proposed	255,192	45.004	120,064	44.998	65,698	45.002

Algorithm	Window image		Sailboat image		Lighthouse image	
	SHD	PSNR	SHD	PSNR	SHD	PSNR
Tai et al. (2009)	75,759	44.997	53,739	44.998	60,843	45.005
Thodi and Rodriguez (2007)	108,265	44.999	69,337	45.005	77,070	45.000
Luo et al. (2010)	140,458	45.002	77,336	45.001	82,948	44.999
Sachnev et al. (2009)	162,268	44.998	96,288	45.001	102,764	44.999
The proposed	172,516	45.004	103,550	45.004	113,508	45.004

Algorithm	Child image		Boating image		Woman image	
	SHD	PSNR	SHD	PSNR	SHD	PSNR
Tai et al. (2009)	58,484	45.004	29,097	44.997	40,632	45.002
Thodi and Rodriguez (2007)	78,381	44.996	39,845	44.998	65,536	44.999
Luo et al. (2010)	102,238	44.995	59,800	45.003	102,790	45.005
Sachnev et al. (2009)	115,554	44.998	74,844	45.000	116,052	44.999
The proposed	119,802	44.997	80,562	44.999	117,968	44.996

Algorithm	Average SHD	SHD improvement ratio
Tai et al. (2009)	49,060.28	130.08%
Thodi and Rodriguez (2007)	71,653.17	57.54%
Luo et al. (2010)	93,616.78	20.58%
Sachnev et al. (2009)	104,889.11	7.62%
The proposed	112,879.11	

in Ψ_α by the embedding strategy described in Section 4.1.

The extraction process for the pixels in Ψ_α is comprised of five steps:

Step 1: Extract the header stream from the first $|\mathcal{H}|$ pixels to get the size of the hidden data, the size of flag bit stream, and the values of two thresholds.

Step 2: From the $(\mathcal{H}+1)$th pixel to the last pixel, calculate the modified prediction error for each pixel

Table 5
The average execution-time (seconds) comparison among the concerned five RHD algorithms.

	Capacity (bpp)						
	0.1	0.2	0.3	0.4	0.5	0.6	0.7
Tai et al. (2009)	0.0110	0.0196	0.0235	0.0280	0.0319	0.0391	0.0422
Thodi and Rodriguez (2007)	0.0555	0.1098	0.1617	0.2140	0.2659	0.3195	0.3711
Luo et al. (2010)	0.1024	0.1848	0.2847	0.4018	0.5371	0.6923	0.8773
Sachnev et al. (2009)	0.0962	0.1413	0.1874	0.2419	0.3062	0.3809	0.4946
The proposed	0.3107	0.4174	0.5076	0.6136	0.7332	0.9076	1.1523

Table 6
The average marked image quality performance comparison between our RDH algorithm with the embedding order determination strategy by Sachnev et al. (2009) and the one with the proposed modified embedding order determination strategy.

Capacity (bpp)						
0.1	0.2	0.3	0.4	0.5	0.6	0.7
With the embedding order determination strategy by Sachnev et al. (2009)						
54.1488	50.3938	47.2909	44.8412	42.8878	41.0570	39.2666
With the proposed modified embedding order determination strategy						
54.5575	50.6542	47.5471	45.0639	43.0663	41.1983	39.3204

by using the proposed GEDP scheme described in Section 3.1.

Step 3: Determine the extraction order based on the modified embedding order determination strategy described in Section 4.2.

Step 4: According to the extraction order of the pixels, extract the correction bit stream, the flag bit stream, and the hidden data from the pixels in Ψ_α by the extracting strategy described in Section 4.1.

Step 5: Recover the LSB values of the first $|\mathcal{H}|$ pixels by the correction bit stream.

5. Experimental results

To test the effectiveness of the proposed RDH algorithm, we used eighteen test images shown in Fig. 5 to conduct experiments. We compared our proposed RDH algorithm with four existing RDH algorithms, namely Tai et al.'s algorithm (Tai et al., 2009), Thodi and Rodriguez's algorithm (Thodi and Rodriguez, 2007), Luo et al.'s algorithm (Luo et al., 2010), and Sachnev et al.'s algorithm (Sachnev et al., 2009). The size of each test image was 512×512. All the five algorithms were implemented on an IBM compatible computer with an Intel Core 2 Duo CPU T9600@2.8 GHz and a 2.96 GB RAM. The operating system was MS-Windows XP; the program development environment was Visual Studio C++2005; and the hidden data were generated by the function *rand*() in C++language.

The comparisons were based on two performance measures, the peak signal-to-noise ratio (PSNR) and the Capacity. The PSNR measures the quality of the marked image whereas the Capacity, measured in bits per pixel (bpp), represents the amount of hidden data. The PSNR of a marked image of size $M \times N$ is defined as

$$\mathrm{PSNR} = 10\log_{10}\frac{255^2}{(1/MN)\sum_{i=0}^{M-1}\sum_{j=0}^{N-1}[x_{i,j}-x'_{i,j}]^2}, \tag{13}$$

where $x_{i,j}$ and $x'_{i,j}$ denote the gray values of the pixels at position (i,j) in an original image and a marked image, respectively. The larger the PSNR, the better will be the image quality. The Capacity of a marked image of size $M \times N$ is defined as

$$\mathrm{Capacity} = \frac{\#\{\text{hidden bits}\}}{MN}, \tag{14}$$

where #{hidden bits} denotes the number of hidden bits. The larger the Capacity, the higher will be the embedding capacity.

Fig. 6 illustrates the PSNR versus the Capacity curves generated by the five RDH algorithms. From the figure, it is obvious that with the same Capacity, the proposed RDH algorithm yields the best quality of marked images among the five RDH algorithms. Next, we demonstrate capacity advantage of the proposed RDH algorithm. Tables 3 and 4 show the comparison results in terms of the size of hidden data (SHD) under the constraints of PSNR $\simeq$ 50 dB and PSNR $\simeq$ 45 dB, respectively. From the two tables, it is clear that based on the eighteen test images, the proposed RDH algorithm can embed the most amount of hidden data under the same marked image quality. On average, under the constraint of PSNR $\simeq$ 50 dB, the SHD improvement ratio of the proposed algorithm over Tai et al.'s algorithm, Thodi and Rodriguez's algorithm, Luo et al.'s algorithm, and Sachnev et al.'s algorithm can, respectively, achieve 188.31%, 74.93%, 26.98%, and 12.54%; and under the constraint of PSNR $\simeq$ 45 dB, the SHD improvement ratio of the proposed algorithm over Tai et al.'s algorithm, Thodi and Rodriguez's algorithm, Luo et al.'s algorithm, and Sachnev et al.'s algorithm can, respectively, achieve 130.08%, 57.54%, 20.58%, and 7.62%, implying that the improvement of the proposed algorithm is significant.

Based on the eighteen test images, Table 5 shows the average execution-time performance comparison among the five concerned RHD algorithms. Although the proposed RHD algorithm has some execution-time degradation, the above results demonstrate the proposed RHD algorithm can achieve the best embedding capacity and marked image quality. Furthermore, even though for the case of Capacity = 0.7 bpp, the average execution time of the proposed RHD algorithm is only 1.1523 seconds, indicating that the proposed RHD algorithm is effective and applicable.

Finally, we discuss the influence of the proposed modified embedding order determination strategy discussed in Section 4.2 upon the marked image quality performance in terms of the average PSNR. Based on the same eighteen test images, Table 6 demonstrates that under the same Capacity, our RHD algorithm with the proposed modified embedding order determination strategy has better average quality of marked images than the one with the embedding order determination strategy by Sachnev et al. (2009), especially in the case of low Capacity.

6. Conclusions

We have presented an improved RDH algorithm based on the proposed GEDP scheme. The contribution of this work is threefold. First, because the embedding capacity and marked image quality are dependent on the prediction accuracy in RDH algorithm, we develop a new GEDP scheme to generate more accurate prediction results. The prediction errors derived by the proposed GEDP

tend to form a sharper Laplacian distribution, and it implies that the proposed RDH algorithm can yield larger embedding capacity and produce better quality of marked images. Second, it is a critical issue to determine the appropriate thresholds, T_P and T_n, which can provide enough usable capacity to embed the hidden data and some overheads as well as generate the best quality of a marked image, so a new systematic way is developed to tackle this problem. Finally, we present a modified embedding order determination strategy to reduce the distortion of a marked image. By experimenting on eighteen test images, the results demonstrate the superior properties of the proposed algorithm in terms of embedding capacity and image quality when compared with four existing state-of-the-art algorithms (Luo et al., 2010; Sachnev et al., 2009; Tai et al., 2009; Thodi and Rodriguez, 2007). The proposed RDH algorithm can be particularly applied to the field of sensitive images, such as military, medical, and artwork images, where the total reconstruction of the original images is imperative.

Appendix A. Derivations of SI masks

In this appendix, we introduce the detailed derivations of the SI masks, which combine Sobel masks (Gonzalez and Woods, 1992) and bilinear interpolation masks. Since derivation for the pixels in Ψ_α is the same as that for the pixels in Ψ_β, we only consider the case of Ψ_α.

Fig. A.1(a) and (b) illustrates the 3×3 horizontal and vertical masks, respectively. After running the horizontal and vertical masks on a 3×3 subimage centered at position (m, n), the horizontal gradient response $\Delta h_{m,n}$ and the vertical gradient response $\Delta v_{m,n}$ can be calculated by

$$\begin{aligned}\Delta h_{m,n} &= \sum_{k=-1}^{1}\rho_k[x_{m+k,n+1}-x_{m+k,n-1}]\\ \Delta v_{m,n} &= \sum_{k=-1}^{1}\rho_k[x_{m+1,n+k}-x_{m-1,n+k}],\end{aligned} \tag{A.1}$$

where $\rho_k = 2$ if $k = 0$; $\rho_k = 1$, otherwise. To make Sobel masks workable on the pixels in Ψ_α, bilinear interpolation mask is used to obtain missing pixels. Given all the pixels in Ψ_α, a full image can be reconstructed by the following rule:

$$x_{m,n} = \begin{cases} x_{m,n} & \text{if } x_{m,n}\in\Psi_\alpha \\ \dfrac{1}{4}\displaystyle\sum_{(m',n')\in\xi} x_{m',n'} & \text{otherwise,}\end{cases} \tag{A.2}$$

where $\xi = \{(m\pm 1, n), (m, n\pm 1)\}$. Then, combining Eqs. (A.1) and (A.2), the following equations are derived:

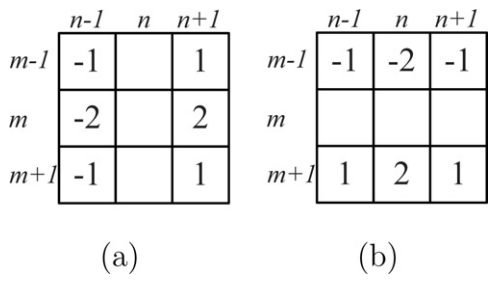

Fig. A.1. Two Sobel masks. (a) The horizontal mask. (b) The vertical mask.

if $x_{m,n}\in\Psi_\alpha$, it yields

$$\begin{aligned}\Delta h_{m,n} &= \frac{1}{4}\left\{\begin{array}{l}6\left[\displaystyle\sum_{k\in\{\pm1\}} x_{m+k,n+1}-x_{m+k,n-1}\right]\\ +2\left[x_{m,n+2}-x_{m,n-2}\right]\end{array}\right\}\\ \Delta v_{m,n} &= \frac{1}{4}\left\{\begin{array}{l}6\left[\displaystyle\sum_{k\in\{\pm1\}} x_{m+1,n+k}-x_{m-1,n+k}\right]\\ +2\left[x_{m+2,n}-x_{m-2,n}\right]\end{array}\right\};\end{aligned} \tag{A.3}$$

otherwise,

$$\begin{aligned}\Delta h_{m,n} &= \frac{1}{4}\left\{\begin{array}{l}\left[\displaystyle\sum_{k_1\in\{\pm2\}} x_{m+k_1,n+1}-x_{m+k_1,n-1}\right.\\ \left.+\displaystyle\sum_{k_2\in\{\pm1\}} x_{m+k_2,n+2}-x_{m+k_2,n-2}\right]\\ +10\left[x_{m,n+1}-x_{m,n-1}\right]\end{array}\right\}\\ \Delta v_{m,n} &= \frac{1}{4}\left\{\begin{array}{l}\left[\displaystyle\sum_{k_1\in\{\pm2\}} x_{m+1,n+k_1}-x_{m-1,n+k_1}\right.\\ \left.+\displaystyle\sum_{k_2\in\{\pm1\}} x_{m+2,n+k_2}-x_{m-2,n+k_2}\right]\\ +10\left[x_{m+1,n}-x_{m-1,n}\right]\end{array}\right\}.\end{aligned} \tag{A.4}$$

From Eqs. (A.3) and (A.4), two SI mask pairs can be obtained. Note that the coefficients of SI masks are normalized into integers for avoiding floating point computation.

Appendix B. Proof of Theorem 1

From Eq. (6), we know that when the hidden bit h is embed in a pixel, the embedding distortion $\mathfrak{D}(e)$ is $|e+h|$. Since $h\in\{0, 1\}$ and e is within the range $[T_n, T_p]$, we have the following equation:

$$\mathfrak{D}(e) = \begin{cases} e & \text{if } 0\le e\le T_p \text{ and } h=0\\ -e & \text{if } T_n\le e<0 \text{ and } h=0\\ e+1 & \text{if } 0\le e\le T_p \text{ and } h=1\\ -e-1 & \text{if } T_n\le e<0 \text{ and } h=1.\end{cases} \tag{B.1}$$

Because the two events, $h = 0$ and $h = 1$, are equiprobable, we have $Pro(h = 0) = Pro(h = 1) = (1/2)$. Thus, the expected value of the distortion caused by embedding h in a pixel with the prediction error ε is $\varepsilon+(1/2)$ $(=(1/2)\varepsilon+(1/2)(\varepsilon+1))$; for pixel with the prediction error $-\varepsilon-1$, the embedding distortion is also $\varepsilon+(1/2)$ $(=(1/2)[-(-\varepsilon-1)]+(1/2)[-(-\varepsilon-1)-1])$. This completes the proof.

References

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

Chang, C.C., Tai, W.L., Lin, C.C., 2006. A reversible data hiding scheme based on side match vector quantization. IEEE Transactions on Circuits and Systems for Video Technology 16 (10), 1301–1308.

Chang, C.C., Lu, T.C., 2006. Reversible index-domain information hiding scheme based on side-match vector quantization. The Journal of Systems and Software 79 (8), 1120–1129.

Chang, C.C., Lin, C.Y., 2007. Reversible steganographic method using SMVQ approach based on declustering. Information Sciences 177 (8), 1796–1805.

Chang, C.C., Lin, C.C., Tseng, C.S., Tai, W.L., 2007. Reversible hiding in DCT-based compressed images. Information Sciences 177 (13), 2768–2786.

Chang, C.C., Lin, C.Y., Fan, Y.H., 2008. Lossless data hiding for color images based on block truncation coding. Pattern Recognition 41 (7), 2347–2357.

Chang, C.C., Kieu, T.D., Wu, W.C., 2009. A lossless data embedding technique by joint neighboring coding. Pattern Recognition 42 (7), 1597–1603.

Chung, K.L., Yang, W.J., Yan, W.M., Wang, C.C., 2008. Demosaicing of color filter array captured images using gradient edge detection masks and adaptive heterogeneity-projection. IEEE Transactions on Image Processing 17 (12), 2356–2367.

Gonzalez, R., Woods, R., 1992. Digital Image Processing. Addison Wesley, Upper Saddle River, NJ.
Honsinger, C.W., Jones, P., Rabbani, M., Stoffel, J.C., 2001. Lossless recovery of an original image containing embedded data. US Patent # 6,278,791.
Hwang, H.J., Kim, H.J., Sachnev, V., Joo, S.H., 2010. Reversible watermarking method using optimal histogram pair shifting based on prediction and sorting. KSII Transactions on Internet and Information Systems 4 (4), 655–670.
Kamstra, L.H.J., Heijmans, A.M., 2005. Reversible data embedding into images using wavelet techniques and sorting. IEEE Transactions on Image Processing 14 (12), 2082–2090.
Kim, H.J., Sachnev, V., Shi, Y.Q., Nam, J., Choo, H.G., 2008. A novel difference expansion transform for reversible data embedding. IEEE Transactions on Information Forensics and Security 3 (3), 456–465.
Lin, C.C., Tai, W.L., Chang, C.C., 2008. Multilevel reversible data hiding based on histogram modification of difference images. Pattern Recognition 41 (12), 3582–3591.
Lin, C.N., Buehrera, D.J., Chang, C.C., Lu, T.C., 2010. Using quad smoothness to efficiently control capacity-distortion of reversible data hiding. The Journal of Systems and Software 83 (10), 1805–1812.
Luo, L., Chen, Z., Chen, M., Zeng, X., Xiong, Z., 2010. Reversible image watermarking using interpolation technique. IEEE Transactions on Information Forensics and Security 5 (1), 187–193.
Mallat, S., 1999. A Wavelet Tour of Signal Processing, 2nd ed. Academic Press, Orlando, FL.
Ni, Y.Q., Shi, N.A., Su, W., 2006. Reversible data hiding. IEEE Transactions on Circuits and Systems for Video Technology 16 (3), 354–362.
Parzen, E., 1960. Modern Probability Theory and its Applications. John Wiley and Sons, New York.
Pei, S.C., Tam, I.K., 2003. Effective color interpolation in CCD color filter arrays using signal correlation. IEEE Transactions on Circuits and Systems for Video Technology 13 (6), 503–513.
Sachnev, V., Kim, H.J., Xiang, S., Nam, J., 2007. An improved reversible difference expansion watermarking algorithm. In: The 6th International Workshop on Digital Watermarking (IWDW 2007), Guangzhou, China, pp. 254–263.
Sachnev, V., Kim, H.J., Nam, J., Suresh, S., Shi, Y.Q., 2009. Reversible watermarking algorithm using sorting and prediction. IEEE Transactions on Circuits and Systems for Video Technology 19 (7), 989–999.
Tai, W.L., Yeh, C.M., Chang, C.C., 2009. Reversible data hiding based on histogram modification of pixel differences. IEEE Transactions on Circuits and Systems for Video Technology 19 (6), 906–910.
Tsai, C.L., Chiang, H.F., Fan, K.C., Chung, C.D., 2005. Reversible data hiding and lossless reconstruction of binary images using pairwise logical computation mechanism. Pattern Recognition 38 (11), 1993–2006.
Thodi, D.M., Rodriguez, J.J., 2007. Expansion embedding techniques for reversible watermarking. IEEE Transactions on Image Processing 16 (3), 721–730.
Tian, J., 2003. Reversible data embedding using a difference expansion. IEEE Transactions on Circuits and Systems for Video Technology 13 (8), 890–896.
Vleeschouwer, C.D., Delaigle, J.F., Macq, B., 2003. Circular interpretation of bijective transformations in lossless watermarking for media asset management. IEEE Transactions on Multimedia 5 (1), 97–105.
Wang, Z.H., Chang, C.C., Lin, C.C., Li, M.C., 2009. A reversible information hiding scheme using left-right and up-down Chinese character representation. The Journal of Systems and Software 82 (8), 1362–1369.
Weinberger, M., Seroussi, G., Sapiro, S., 1996. LOCO-I: a low complexity, context-based, lossless image compression algorithm. In: Proc. of IEEE Data Compression Conf, pp. 140–149.
Yang, W.J., Chung, K.L., Liao, H.Y.M., 2012. Efficient reversible data hiding for color filter array images. Information Sciences 190, 208–226.

Wei-Jen Yang received the B.S. degree in Computer Science and Information Engineering from National Taiwan University of Science and Technology, Taipei, Taiwan, in 2004 and the Ph.D. degree in Computer Science and Information Engineering from National Taiwan University, Taipei, Taiwan, in 2009. He is now a postdoctoral researcher in Department of Computer Science and Information Engineering at National Taiwan University of Science and Technology. His current research interests include color image processing, digital camera image processing, data hiding, image/video compression, computer vision, pattern recognition, and algorithms. In 2007, he received the best paper award from the Society of Computer Vision, Graphics, and Image Processing in Taiwan; and in 2010, he received Ph.D. Dissertation Award from Institute of Information & Computing Machinery in Taiwan.

Kuo-Liang Chung is a Professor in the Department of Computer Science and Information Engineering at National Taiwan University of Science and Technology (NTUST), Taipei, Taiwan, since 1995. He received the B.S., M.S., and Ph.D. degrees from National Taiwan University, Taipei, Taiwan, in 1982, 1984, and 1990, respectively. After two years of obligatory military services (1984–1986), he was enrolled as a research assistant in the Institute of Information Science at Academia Sinica, Taiwan from 1986 to 1987. He was a Visiting Scholar at University of Washington in summer 1999. From 2003 to 2006, he served as the chair of the Department of Computer Science and Information Engineering at NTUST. He was a Managing Editor of Journal of Chinese Institute of Engineers from Jan. 1996 to Dec. 1998. In August 2000, he served as Program Co-Chair of 2000 Conference on Computer Vision, Graphics, and Image Processing, Taiwan. Dr. Chung was the recipient of the Distinguished Research Award from National Science Council of Taiwan in 2004; the recipient of the Best Paper Award from the Image Processing and Pattern Recognition Society of Taiwan in 2007; the recipient of Distinguished Teaching Award of NTUST in 2009. Since 2009, he has been a University Chair Professor at NTUST. His current research interests include image/video compression, image/video processing, pattern recognition, 3D video processing, and shape analysis in computer vision. He now serves as an Associate Editor of Journal of Visual Communication and Image Representation. Dr. Chung is a senior member of IEEE and a fellow of IET.

Hong-Yuan Mark Liao received a BS degree in physics from National Tsing-Hua University, Hsin-Chu, Taiwan, in 1981, and an MS and Ph.D. degree in electrical engineering from Northwestern University in 1985 and 1990, respectively. In July 1991, he joined the Institute of Information Science, Academia Sinica, Taiwan. He is a Distinguished Research Fellow now. During 2009-2011, he was the division chair of the computer science and information engineering division II, National Science Council of Taiwan. He is jointly appointed as a professor of the Computer Science and Information Engineering Department of National Chiao-Tung University and the Department of Electrical Engineering of National Cheng Kung University. During 2009–2012, he was jointly appointed as the Multimedia Information Chair Professor of National Chung Hsing University. From August 2010, he has been appointed as an Adjunct Chair Professor of Chung Yuan Christian University. His current research interests include multimedia signal processing, video-based Surveillance Systems, video forensics, and multimedia protection.

Dr. Liao was a recipient of the Young Investigators' award from Academia Sinica in 1998. He received the distinguished research award from the National Science Council of Taiwan in 2003 and 2010, and the National Invention Award of Taiwan in 2004. In 2008, he received a Distinguished Scholar Research Project Award from National Science Council of Taiwan. In 2010, he received the Academia Sinica Investigator Award. In June 2004, he served as the conference co-chair of the 5th International Conference on Multimedia and Exposition (ICME) and technical co-chair of the 8th ICME held at Beijing. In Jan. 2011, Dr. Liao served as General co-chair of the 17th International Conference on Multimedia Modeling. From 2006–2008, Dr. Liao was the president of the Image Processing and Pattern Recognition Society of Taiwan.

Dr. Liao is on the editorial boards of the IEEE Signal Processing Magazine, the IEEE Transactions on Image Processing, and the IEEE Transactions on Information Forensics and Security. He was an associate editor of IEEE Transactions on Multimedia during 1998–2001.

Wen-Kuang Yu received the B.S. degree in Electronic Engineering from National Taiwan University of Science and Technology, Taipei, Taiwan, in 1998 and the M.S. degree in Computer Science and Information Engineering from National Taiwan University of Science and Technology, Taipei, Taiwan, in 2011. His currently research interests include image/video processing, reversible data hiding, pattern recognition, and video de-interlacing.