A New Reference Pixel Prediction for Reversible Data Hiding with Reduced Location Map

Jeanne Chen [1], Tung-Shou Chen [1] , Wien Hong[2], Gwoboa Horng[3], Han-Yan Wu[4] and Chih-Wei Shiu[3]

[1] Department of Computer Science and Information Engineering, National Taichung University of Science and Technology
Taichung, 404 - TW
[e-mail: jeanne@nutc.edu.tw, tschen@nutc.edu.tw]
[2] Department of Information Management, Yu Da University
Miaoli, 361 - TW
[e-mail: wienhong@ydu.edu.tw]
[3] Department of Computer Science and Engineering, National Chung Hsing University
Taichung, 402 - TW
[e-mail: gbhorng@cs.nchu.edu.tw, chihwei.shiu@gmail.com]
[4] Department of Computer Science, National Chiao Tung University
Hsinchu, 300 - TW
[e-mail: hanyan.wu0414@gmail.com]
*Corresponding author: Wien Hong

Received August 30, 2013; revised January 16, 2014; accepted February 19, 2014; published March 31, 2014

Abstract

In this paper, a new reversible data hiding method based on a dual binary tree of embedding levels is proposed. Four neighborhood pixels in the upper, below, left and right of each pixel are used as reference pixels to estimate local complexity for deciding embeddable and non-embeddable pixels. The proposed method does not need to record pixels that might cause underflow, overflow or unsuitable for embedment. This can reduce the size of location map and release more space for payload. Experimental results show that the proposed method is more effective in increasing payload and improving image quality than some recently proposed methods.

Keywords: Data hiding, reference pixel, error energy, variance, location map

http://dx.doi.org/10.3837/tiis.2014.03.023

1. Introduction

Data hiding is a technique which embeds additional information into images, music, and other media for delivery to specific destination. A digital image is a suitable data hiding carrier since it is a common type of content on the internet. Image data hiding schemes can be divided into two kinds, namely, non-reversible ones [1]-[5] and reversible ones [6]-[18]. A non-reversible hiding scheme has high payload but it sacrifices image quality. A reversible hiding scheme preserves the original image but it is restricted to low payload.

Tian [9] proposed a reversible scheme to embed one bit of information by expanding the difference of two neighboring pixel pair to achieve a high payload of 0.5 bits per pixel (bpp). However, larger pixel difference expansion will increase distortion. Ni et al. [10] proposed a reversible data hiding method based on histogram-shifting. Information is embedded by shifting histogram between the peak and zero points. Since only one grayscale value per pixel is modified, the image quality is preserved. However, the payload is restricted by the histogram peak height. Thodi and Rodriguez [11] proposed the shifting of prediction error scheme by combining the pixel value prediction and histogram-shifting. Since the prediction errors in Thodi and Rodriguez's method are smaller in comparison to the difference expansion, their scheme offers higher image quality than that of Tian's method.

Tsai et al. [12] proposed a reversible method based on the modification of pixel differences. The image is divided into N blocks where every block is a group of $n \times n$ pixels. The center pixel of each block is not modified to ensure image reversibility. A total of $N \times ((n \times n) - 1)$ pixel difference values for the blocks are calculated and used to build a difference histogram. The difference values are either positive or negative making up the two peaks used for embedding secret data. Tai et al. [13] also proposed a reversible method based on histogram shifting. This method makes use of the inverse s-order to scan the image, and the previous scanned pixel is the reference pixel. The absolute difference values between the current scanned pixels and the reference pixels are calculated, and are used to generate the difference histogram for embedding data. Tai et al.'s method also makes use of a binary tree with embedding level L to offer an adjustable payload. However, Tai et al.'s method shifts the histogram of absolute prediction errors. The shifted negative prediction errors resulted in unnecessary image distortion.

Hong et al. [14] also proposed a reversible histogram shifting method similar to Tsai et al.'s method. The image was partitioned into blocks of $n \times n$ pixels, and the center pixels of blocks are not only selected as reference pixels but are also used to estimate the local variance. The variance is then used to control the payload size and image quality. This method outperforms Tsai et al.'s method in terms of payload and image quality. In 2012, Hong [15] proposed a reversible method by using the dual binary tree (DBT) and error energy estimator [19] to improve Tai et al.'s method. This method avoids unnecessary pixel modifications and uses the median edge detection (MED) [20] predictor to improve Tai et al.'s method with higher payloads and image quality. However, the choice of reference pixels in this method is not fully explored, and the size of the location map can be further reduced. Both improvements resulted in higher payloads and image quality.

In previous research on reversible data hiding using prediction error shifting, one or more reference pixels are used to calculate prediction errors [6]-[7],[11]-[15]. The to-be-predicted pixels are half enclosed by the reference pixels. For example, Hong's method [15] only uses the left, upper and upper left pixels as reference pixels to predict the current pixel, whereas Tai et al.'s method only uses previously visited pixel to predict the current pixel. However, it is known that using the full neighborhood pixel information (termed as full enclosed) is more precise than the half enclosed partial neighborhood pixels in terms of predicting pixel values. In addition, some pixels after embedding may result in underflow or overflow situations. Therefore, a location map for recovering the image is used to

keep records of the pixels. Full enclosed reference pixels provide more information to eliminate the requirement of recording the location map. Since the location map occupies space, reducing the length of the location map is also important.

In this paper, we propose a full enclosed prediction method that outperforms previous related works. The full enclosed prediction is based on top, below, left and right neighborhood pixels, known as reference pixels. Before a secret message is embedded, the variance of the reference pixel is calculated. The aim is to prevent embedding in complex pixels so as to maintain a high image quality. Furthermore, the reference pixel is also used to predict the current pixel for possible overflow or underflow which would then be excluded from embedding. Finally, the location map records only pixels that are used in embedding. This decreases the size of the un-compressed location map. The size of the location map is smaller than the one in Hong's MED prediction method [15] which requires the map to be the size of the image.

2. Related Work

Hong's [15] method makes use of prediction error shifting with a dual binary tree (DBT) of embedding levels for hiding data (see **Fig.1**). The tree node represents the prediction error. For example, if the embedding level is L=1, the peaks used for embedding are -2,-1, 0, 1 and 2. The prediction errors are classified into three types, namely embeddable, shiftable and non-embeddable.

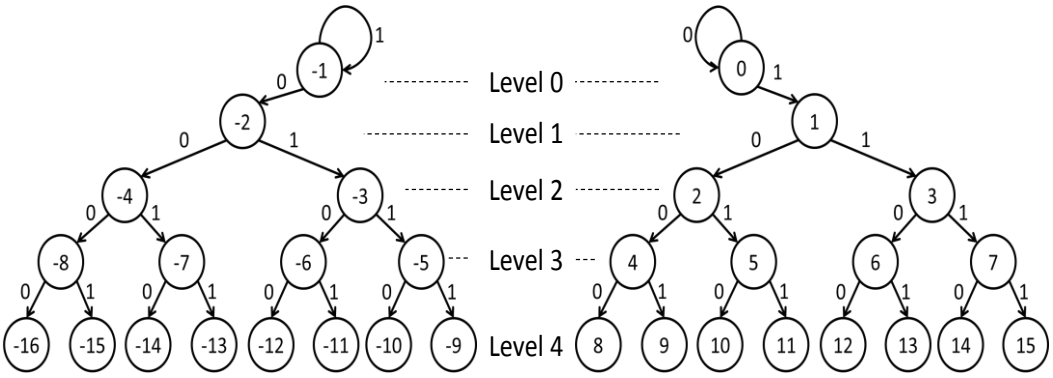


Fig. 1. Dual binary tree

The error energy estimator predicts the energy of prediction error. When the error energy is greater than a pre-defined threshold, the prediction error is considered as non-embeddable and no embedding is performed. Otherwise, the prediction error is either embeddable or shiftable. The error energy is calculated as in Eq. (1):

$$\Delta_{r,c} = d_h + d_v + 2\left|e_{r-1,c}\right| \tag{1}$$

where $\Delta_{r,c}$ is the error energy of pixel $I_{r,c}$ located at (r,c). $e_{r-1,c}$ is the prediction error of pixel at $(r-1,c)$. d_v and d_h are defined in Eq.(2) and Eq.(3), respectively.

$$d_v = \left|I_{r,c-1} - I_{r-1,c-1}\right| + \left|I_{r-1,c} - I_{r-2,c}\right| + \left|I_{r-1,c+1} - I_{r-2,c+1}\right| \tag{2}$$

$$d_h = \left|I_{r-1,c} - I_{r-2,c}\right| + \left|I_{r,c-1} - I_{r-1,c-1}\right| + \left|I_{r,c-1} - I_{r+1,c-1}\right| \tag{3}$$

Here is the detailed procedure of the embedding method. Suppose an 8-bit gray cover image *Ic* of

size $N \times N$ is to be embedded using the embedding level L and threshold TH. To avoid underflow and overflow, pixel values larger than $255-2^L$ or smaller than 2^L are shifted by 2^L units. A 2-dimensional location map B is created with 0 and 1 representing un-shifted and shifted pixel locations. The JBIG-2 coder [21] is used to compress B. The compressed location map and secret data S are concatenated to form a bitstream M. Let the shifted image be I.

We scan I using the raster scan order and use Eq.(1) to calculate the error energy Δ for each visited pixel. If $\Delta \le TH$, this means that the corresponding prediction error is either embeddable or shiftable; then MED is used to predict the current pixel values and to obtain the prediction errors e. If the prediction error e fulfills $-2^L \le e < 2^L$, then e is embeddable. In this case, e is modified to $2e+m$, where m is the message bit extracted from M. If the condition fails then e is shiftable and has to be shifted by 2^L. If $\Delta > TH$, then the corresponding prediction error is non-embeddable and is excluded from embedding. Repeat this procedures until all the bits in M are embedded. Finally, we get the stego image I'.

To extract the secret data and to recover the cover image Ic from I', we scan the stego image in raster-scan order and calculate the error energy Δ as well as prediction error e. If $\Delta \le TH$ and $-2^{L+1} \le e'_{r,c} < 2^{L+1}$, a message bit m can be extracted by calculating $m = \mathrm{mod}(e, 2)$. Once the message bits are all extracted, pixels previously modified in the embedding stage are shifted to obtain the pre-shifted image I. Based on the lengths of M and S, we extract the compressed location map from I' and then decompress it into B. Based on B, we transform I back to Ic

3. The Proposed Method

Hong [15] proposed the method to use DBT with embedding level L and MED to predict pixel values. The MED prediction is taken from half enclosed pixels, namely, the left side, upper and left upper pixels. The half enclosed prediction provides less accurate results in comparison to full enclosed one. Error energy was used to reject non-embeddable prediction error from unnecessary modification. Similar to MED, the estimation used in [15] of error energy is also half enclosed. Moreover, to avoid underflow and overflow, a 2-dimensional array is used to record the location map. The location map is compressed and hidden together with the secret data. The size of the location map before compression is the size of the image. However, for most natural images, underflow and overflow occurred in few pixels with values less than 2^L and greater than $255-2^L$. As a result, it is inefficient to record these pixels in the location map.

Our proposed method is an improvement of [15] which also uses DBT with embedding level L to hide data. However, the proposed method is a full enclosed prediction (FEP) which makes use of the upper, below, left and right pixels as reference pixels. The largest and smallest pixels are discarded and the medium two pixels are averaged and used in the prediction. Both FEP and MED performed well when predicting pixels near the edge. However, FEP has a higher accurate prediction since the full enclosed reference pixels are used to decide whether a prediction error is non-embeddable or embeddable. Furthermore, the reference pixel can be used to accurately predict pixels that might underflow or overflow, and these pixels will be skipped in embedding stage and will not be recorded in the location map. Therefore, the location map is smaller for not having to record unnecessary pixels whereby leaving more space for payload.

3.1 Full Enclosed Predictor

The full enclosed predictor (FEP) uses by the upper, below, right and left reference pixels. Suppose $I_{r,c}$ is the pixel located at r-th row and c-th column of the image I. The reference pixels are $I_{r-1,c}$, $I_{r+1,c}$, $I_{r,c-1}$ and $I_{r,c+1}$. The four reference pixels are sorted in descending order to obtain the sorted

result $D_{r,c}$, as in Eq. (4).

$$D_{r,c} = \text{sort}\left(I_{r-1,c}, I_{r+1,c}, I_{r,c-1}, I_{r,c+1}\right) = \left(D_{r,c}^{(1)}, D_{r,c}^{(2)}, D_{r,c}^{(3)}, D_{r,c}^{(4)}\right), \tag{4}$$

where $D_{r,c}^{i} \geq D_{r,c}^{i+1}$, $1 \leq i \leq 4$. Next, we ignore the largest and smallest reference pixels and use the average of second and third largest pixels. Eq. (5) calculates the predicted value $p_{r,c}$ of $D_{r,c}$.

$$p_{r,c} = \text{round}\left(\frac{D_{r,c}^{(2)} + D_{r,c}^{(3)}}{2}\right), \tag{5}$$

where round(x) function rounds x to the nearest integer. The prediction error for $I_{r,c}$ is $e_{r,c} = I_{r,c} - p_{r,c}$.

3.2 Variance of Reference Pixel

Variances of the full enclosed reference pixels are used to estimate whether a pixel is embeddable or non-embeddable. The variance is calculated by

$$v_{r,c} = \sum_{i=1}^{4}\left(\left(D_{r,c}^{(i)} - \bar{I}_{r,c}\right)^2\right) / 4, \tag{6}$$

where $\bar{I}_{r,c}$ is average value of reference pixels of $I_{r,c}$ defined by

$$\bar{I}_{r,c} = \sum_{i=1}^{4}\left(D_{r,c}^{(i)}\right) / 4. \tag{7}$$

Suppose $v_{r,c}$ is greater than a predefined threshold *TH*, then $I_{r,c}$ is excluded for data embedding because it is likely to carry no data but has to be shifted.

3.3 Predicting Underflow and Overflow (PUO)

Empirically, there are few pixels with possible underflow or overflow after embedment. Therefore, we use the reference pixels to predict pixels that might result in underflow or overflow. These pixels will then be excluded from embedment to prevent the downgrading of image quality. Suppose $\max(D_{r,c}) \geq 255 - 2^L$ or $\min(D_{r,c}) \leq 2^L$, then we assume that $I_{r,c}$ could result in either underflow or overflow. Therefore, no embedment is done for $I_{r,c}$ and will, therefore, be excluded from the location map.

3.4 Prevent Underflow and Overflow with Location Map

In order to avoid underflow and overflow after data embedment, one possibility is to modify these pixels and record their locations in the location map. The location map will be compressed and then embedded with the secret information.

Our proposed method do not record the overflow and underflow locations but instead records the decisions made based on reference pixels. We need only to sequentially record the properties of the processed pixels as in Eq.(8).

$$B_b = \begin{cases} 0, & \left(I_{r,c} < 255 - 2^L \text{ and } I_{r,c} \geq 255 - 2^{L+1}\right) \text{ or } \left(I_{r,c} > 2^L \text{ and } I_{r,c} \leq 2^{L+1}\right), \\ 1, & I_{r,c} \geq 255 - 2^L \text{ or } I_{r,c} \leq 2^L. \end{cases} \tag{8}$$

In Eq.(8) B_b represents the bth value of location map B starting from 0. Every time $I_{r,c}$ satisfies Eq. (8), then b is increased by 1. After being recorded the location map, pixel values are shifted by 2^L unit to prevent underflow and overflow. After modifying all pixels which might result in either underflow or overflow, B is compressed to be B' using JBIG-2 coder [21] and concatenated to the secret data to be embedded in the image. The size of the uncompressed location map of the proposed method is equal to the number of embeddable and shiftable prediction errors, which is always smaller than the location map used in [15], where the size of the uncompressed location map is equal to the size of the cover image.

3.5 Embedding Procedure

Let the cover image Ic be an 8-bit gray image of size $N \times N$. The stego image I' is the cover image with embedded secret data S and B is the location map. First we divide Ic into two disjoint sets, S_1 and S_2. Suppose (r,c) is the location of pixel where $2 \leq r \leq N-1$ and $2 \leq c \leq N-1$. If $\text{mod}(r,2) = \text{mod}(c,2)$ then $(r,c) \in S_1$, otherwise $(r,c) \in S_2$. The embedding process for S_1 and S_2 are similar and thus we describe only the embedding process for set S_1. If more message bits are to be embedded, then perform the embedding for set S_1 followed by S_2.

Input: Cover image Ic, embedding level L, threshold TH, secret data S.
Output: Stego image I', length of message bit $|M|$, length of secret data $|S|$.

Step 1. In raster-scan order, scan Ic where $Ic_{r,c}$ is the pixel located at (r,c) for $2 \leq r \leq N-1$ and $2 \leq c \leq N-1$. If $(r,c) \in S_1$, then goto Step 2; otherwise $I_{r,c} = Ic_{r,c}$, and go to Step 5.

Step 2. Suppose E is the embeddable prediction error in the set. Use Eq.(4) to get the reference pixel $D_{r,c}$ of $Ic_{r,c}$. Next, Eq.(6) is used to calculate the variance $v_{r,c}$ of $Ic_{r,c}$. If $v_{r,c} \leq TH$ and $\max(D_{r,c}) < 255 - 2^L$ and $\min(D_{r,c}) > 2^L$ then the corresponding prediction error $p_{r,c}$ is embeddable. Let $(r,c) \in E$ and goto Step 3; otherwise $I_{r,c} = Ic_{r,c}$ and go to Step 5.

Step 3. Use Eq.(8) to determine if $Ic_{r,c}$ need to be recorded. At the same time, determine if $Ic_{r,c}$ is to be modified to avoid underflow or overflow. Let the modified $Ic_{r,c}$ be $I_{r,c}$. The rule for modification is as follows:

$$I_{r,c} = \begin{cases} Ic_{r,c} - 2^L, & Ic_{r,c} \geq 255 - 2^L, \\ Ic_{r,c} + 2^L, & Ic_{r,c} \leq 2^L, \\ Ic_{r,c}, & \text{otherwise.} \end{cases} \tag{9}$$

Step 4. Use Eq.(5) to get the predicted value $p_{r,c}$ of $I_{r,c}$. The prediction error for $I_{r,c}$ is $e_{r,c} = I_{r,c} - p_{r,c}$.

Step 5. Repeat 0 thru' Step 4 to complete the location map and to check underflow and overflow.

Step 6. Use JBIG-2 coder [21] to compress B to B'. B' and secret data S are concatenated to form a bitstream M.

Step 7. Scan $I_{r,c}$. If $(r,c) \in E$ then use the following equation to modify $I_{r,c}$:

$$I'_{r,c} = \begin{cases} I_{r,c} + 2^L, & e_{r,c} \geq 2^L, \\ I_{r,c} - 2^L, & e_{r,c} < -2^L, \\ I_{r,c} + e_{r,c} + m, & \text{otherwise.} \end{cases} \tag{10}$$

Let $I'_{r,c}$ be the modified pixel of $I_{r,c}$. If $(r,c) \notin E$, then $I'_{r,c} = I_{r,c}$.

Step 8. Repeat Step 7.until all of M are embedded.

3.6 Extraction and Recovery Procedure

In our proposed method, the image is divided into two groups for embedding. In the extraction and recovery process, the group embedded last will be the first to be extracted. For example, if S_1 is embedded first followed by S_2 then extraction and recovery starts from S_2 followed by S_1. Both S_1 and S_2 require the same procedure to extract and recover. The procedure for extraction and recover of S_1 is as the following:

Input: Stego image I', threshold TH, length of message bits $|M|$, length of secret data $|S|$.

Output: Original cover image Ic, secret data S.

Step 1. In raster-scan order, scan I' and $I'_{r,c}$ is the pixel at location (r,c) for $2 \leq r \leq N-1$ and $2 \leq c \leq N-1$. If $(r,c) \in S_1$, goto Step 2. If $(r,c) \notin S_1$ and $I_{r,c} = I'_{r,c}$ then go to Step 6.

Step 2. Use Eq.(4) to get the reference pixel $D^{r,c}$ of $I'_{r,c}$. Next, Eq.(6) is used to calculate the variance $v_{r,c}$ of $I'_{r,c}$. If $v_{r,c} \leq TH$ and $\max(D_{r,c}) < 255 - 2^L$ and $\min(D_{r,c}) > 2^L$ then $I'_{r,c}$ is embeddable. Let $(r,c) \in E$ and perform Step 3 through Step 5; otherwise $I_{r,c} = I'_{r,c}$, go to Step 6.

Step 3. Use Eq. (5) to get the predicted value $p'_{r,c}$ of $I'_{r,c}$. The prediction error for $I'_{r,c}$ is $e'_{r,c} = I'_{r,c} - p'_{r,c}$.

Step 4. If $-2^{L+1} \leq e'_{r,c} < 2^{L+1}$ then a bit message m is extracted, where $m = \text{mod}\left(e'_{r,c}, 2\right)$.

Step 5. Use the following equation to recover $I_{r,c}$ from $I'_{r,c}$:

$$I_{r,c} = \begin{cases} I'_{r,c} - 2^L, & e'_{r,c} \geq 2^{L+1}, \\ I'_{r,c} + 2^L, & e'_{r,c} < -2^{L+1}, \\ I'_{r,c} - \left\lceil \dfrac{e'_{r,c}}{2} \right\rceil, & \text{otherwise.} \end{cases} \tag{11}$$

Step 6. Repeat Step 1 through Step 5, until all embedded secret bits are extracted.

Step 7. According to $|M|$ and $|S|$, extract B' which is decompress into B.

Step 8. Scan $I_{r,c}$. If $(r,c) \in E$ then extract from B one bit b, then base on following equation to recover $Ic_{r,c}$ from $I_{r,c}$..

$$Ic_{r,c} = \begin{cases} I_{r,c} + 2^L, & (b=1) \text{ and } (I_{r,c} \geq 255 - 2^{L+1}), \\ I_{r,c} - 2^L, & (b=1) \text{ and } (I_{r,c} \leq 2^{L+1}), \\ I_{r,c}, & \text{otherwise.} \end{cases} \quad (12)$$

Step 9. Repeat Step 8 until all $Ic_{r,c}$ are recovered from $I_{r,c}$.

4. Experimental Results

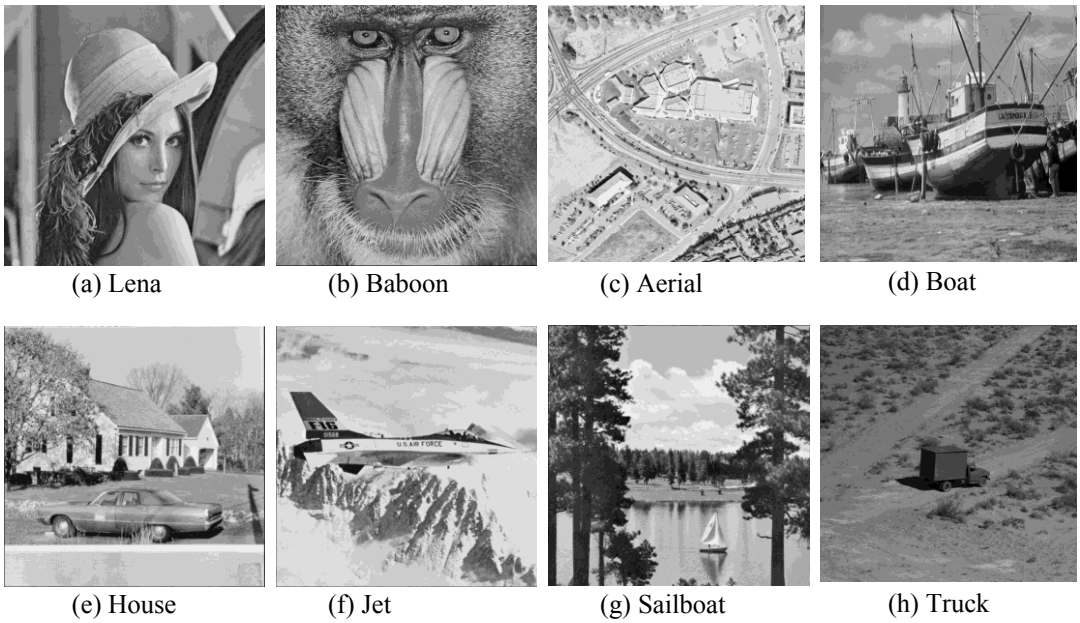
In this section, experimental tests will be performed using eight 512×512 gray-scale images, namely, Lena, Baboon, Aerial, Boat, House, Jet, Sailboat, and Truck (refer to **Fig. 2**). These images were downloaded from the USC-SIPI image database [22]. The peak-to-noise ratio (PSNR) is used to evaluate the quality of an image, and the embedded information will be generated by a pseudo random number generator (PRNG). In our proposed embedment we may embed in one group (either S_1 or S_2) or in both groups. Unless specifically mentioned, all experiments will embed in both groups.

4.1 Payload and Quality Analysis

First, we will compare our method with Hong's method [15] using all eight images for the highest payload at different embedding levels. The results are as shown in **Table 1** for embedding levels $L = 0$ to $L = 4$. Results show that the proposed method has higher payloads than that of [15]. The reason is that the FEP predictor used in the proposed method provides better performance in comparison to MED predictor used in [15].

Table 1. Maximum payload comparison between the proposed method and [13]

Cover image	$L=0$		$L=1$		$L=2$		$L=3$		$L=4$	
	Hong	Proposed	Hong	Proposed	Hong	Proposed	Hong	Proposed	Hong	Proposed
Lena	53424	63875	98853	113964	166335	183048	224946	236929	250580	256229
Baboon	17636	21792	34977	41840	67896	77462	119601	131148	180430	190148
Aerial	50141	63692	89263	110081	136696	161203	182770	205314	221894	237273
Boat	33128	39214	63847	72922	117419	127285	189890	197292	243460	245623
House	76365	84929	124514	136004	173537	186509	214811	225621	243384	249629
Jet	79373	90360	136672	151418	194428	209481	231960	242541	250734	256038
Sailboat	29625	36102	57410	66420	104377	114736	170587	179711	230432	236716
Truck	46969	54027	84700	99841	147780	165338	213807	227243	251951	254998
Avg.	48333	56749	86280	99061	138559	153133	193547	205725	234108	240832
Gain in payload	8416		12781		14574		12178		6724	

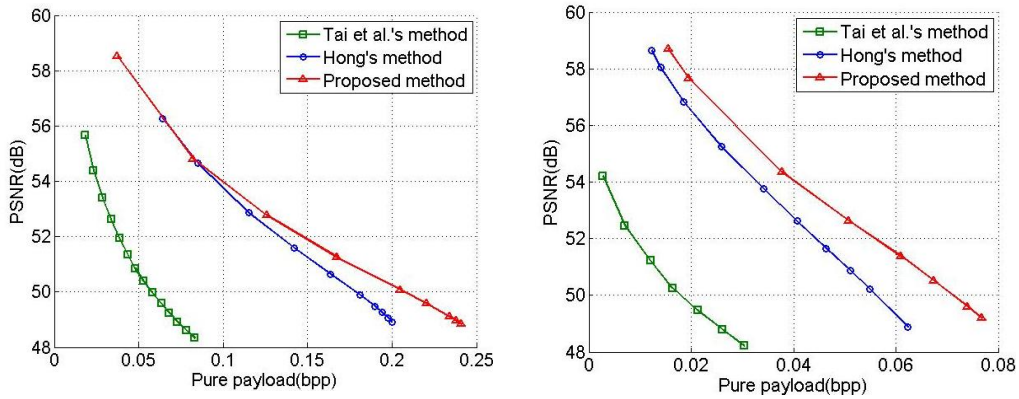


(a) Lena (b) Baboon (c) Aerial (d) Boat

(e) House (f) Jet (g) Sailboat (h) Truck

Fig. 2. Eight test images.

Fig. 3 shows comparison results between the proposed method and two other research, namely, Hong [15] and Tai et al. [13], in terms of payload size and image quality. The experimental images Lena and Baboon are used. As illustrated, Tai et al.'s method has low payload and low image quality. The proposed method has the highest payload capacity while maintaining high image quality. In Lena, Hong's method and the proposed method maintain the same image quality for the payloads between 0.035 bpp and 0.085 bpp. On the other hand, the proposed method has the highest payload and image quality for image Baboon. The reason for the differences is that Lena has a smoother gradient than Baboon. The proposed method maintains high image quality for high payload due to the high prediction precision of FEP. And smaller location map. Overall the proposed method has better capacity and image quality in comparison to both Hong's and Tai et al.'s methods.

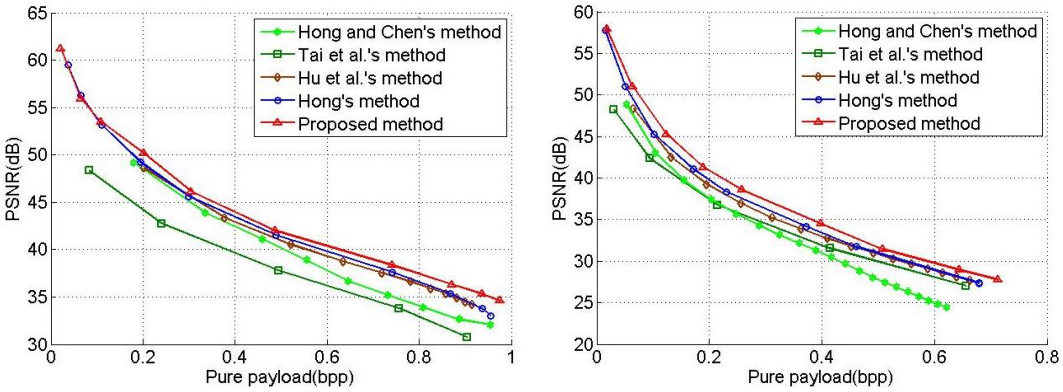


(a) Lena (b) Baboon

Fig. 3. Compare payload and image quality

Next, comparisons will be made between Tai et al.'s [13], Hong's [15], Hong and Chen's[14], and Hu et al.'s [18] methods for pure payload and image quality. The experimental images used are Lena and Baboon. For [13], [15] and the proposed method, the embedding levels are set at $L = 0$ to 4. The

block size used in [11] is set to 3×3 with repeated embedding to achieve high payload. **Fig. 4** shows that for the same payload, our proposed method outperformed the other four methods. For example, when Lena's payload is 0.6 bpp, the image quality of our proposed method is 41 dB while methods [15] and [18] are less than 40 dB, method [14] is 37.5 dB, and method [13] is only 36 dB. For Baboon at payload 0.6 bpp, our proposed method is 30 dB, methods [15] and [18] are around 29 dB, while method [13] is 28 dB, and method [14] is only 25 dB.



(a) Lena **Fig. 4.** High payload comparison (b) Baboon

4.2 Comparing the Location Map

Table 3 shows the comparison results for the size of the compressed location map between Hong's [15] and the proposed method. The results show that when the embedding level increases, more pixels will result in underflow or overflow. This will increase the size of the location map in both methods. Note that in Table 3 when $L=3$ and $L=4$, the sizes (marked in *) of the location map for three experimental images of the proposed method are slightly larger than those of Hong's. This is a natural phenomenon in smooth images where fewer pixels are excluded for underflow or overflow. However, on average, the proposed method has smaller location maps than that of Hong's. For example, at $L=5$, the size of the averaged location map of the proposed method is only 6,696 bits, but Hong's method requires 18,267 bits.

Table 3. Comparison of the size of the compressed location map

Cover image	$L=3$		$L=4$		$L=5$	
	Hong	Proposed	Hong	Proposed	Hong	Proposed
Lena	312	***432**	360	***432**	6480	3464
Baboon	632	576	1968	1,776	11448	6576
Aerial	5464	2776	11288	5888	43760	11,824
Boat	2464	2112	7,896	3248	24336	8336
House	344	***432**	992	752	13864	2192
Jet	312	***504**	440	***568**	840	664
Sailboat	720	432	1648	1,496	39,960	18312
Truck	952	672	2520	1,056	5448	2200
Avg. bits	1400	1314	3389	2,369	18267	6696
Gain in payload	86		1,020		11,571	

Table 4 shows the size of the compressed location map with and without excluding those pixels that might underflow or overflow. The threshold is set to $TH=\infty$ to maximize the payload. When embedding level increases, pixels that might underflow or overflow also increase which means the size

of location map will also increases. From the table, Aerial and Boat show obvious increase in the size of the location map. However, by excluding pixels that might overflow and underflow can effectively decrease the location map size and thus increase the payload.

Table 4. Location map comparison with and without excluding those pixels that might underflow or overflow

Cover image	$L=3$		$L=4$		$L=5$	
	without excluding	excluding	without excluding	excluding	without excluding	excluding
Lena	432	432	568	432	14344	3464
Baboon	680	576	2520	1776	19528	6576
Aerial	5056	2776	14752	5888	129136	11824
Boat	3624	2112	13824	3248	72880	8336
House	432	432	1688	752	55632	2192
Jet	504	504	568	568	1352	664
Sailboat	432	432	2288	1496	111264	18312
Truck	1480	672	4040	1056	11944	2200
Avg. bits	1580	1314	5031	2369	52010	6696
Gain in payload	266		2,662		45,314	

5. Conclusion

The proposed reversible data hiding method makes use of the predictive FEP, variance control, and a modified location map recording techniques. FEP makes use of four neighborhood pixels as reference pixels. The largest and smallest ones are discarded while the medium two pixels are averaged to be used for prediction. This method eliminates predicting pixels on the edge and avoids the bigger predictive error between the biggest and the smallest reference pixel, and offers precise predictive error to increase payload. The pixel variance is used to predict non-embeddable pixel, and to exclude pixels of underflow or overflow. The modified location map only records pixels that are used in embedding. This decreases the length of the location map. Experimental results show that our proposed method is effective in increasing image payload and image quality, and has better results compared to four other similar methods.

References

[1] J. Mielikainen, "LSB matching revisited," *IEEE Signal Processing Letters*, 13 (2006), pp. 285-287. DOI. 10.1109/LSP.2006.870357

[2] X. Zhang, S. Wang, "Efficient steganographic embedding by exploiting modification direction," *IEEE Communications Letters*, 10 (2006), pp. 781-783. DOI. 10.1109/LCOMM.2006.060863

[3] J.M. Guo, "Improved data hiding in halftone images with cooperating pair toggling human visual system," *International Journal of Imaging Systems and Technology*, 17 (2008), pp. 328-332. DOI. 10.1109/ICASSP.2007.366228

[4] W.B. Lee, J.H. Li, S.C. Chen, T.H. Chen, "An authenticated secure image hiding scheme,"

Imaging Science Journal, 57(2009), pp. 109-117.

[5] T.H. Chen, "Remarks on some watermarking schemes based on DWT," *Journal of Computational Information Systems*, 6(2010), pp. 1571-1575.

[6] S.L. Lin, C.F. Huang, M.H. Liou, C.Y. Chen, "Improving histogram-based reversible information hiding by an optimal weight-based prediction scheme," *Journal of Information Hiding and Multimedia Signal Processing*, 4 (2013), pp. 19-33. DOI. 10.1109/ICGEC.2010.170

[7] S. Weng, J.S. Pan, X. Gao, "Reversible watermark combining pre-processing operation and histogram shifting," *Journal of Information Hiding and Multimedia Signal Processing*, 3 (2012), pp. 320-326.

[8] C.Y. Yang, C.H. Lin, W.C. Hu, "Reversible data hiding for high-quality images based on integer wavelet transform," *Journal of Information Hiding and Multimedia Signal Processing*, 3 (2012), pp. 142-150.

[9] J. Tian, "Reversible data embedding using a difference expansion," *IEEE Transactions on Circuits and Systems for Video Technology*, 13 (2003), pp. 890-896. DOI. 10.1109/TCSVT.2003.815962

[10] Z. Ni, Y.Q. Shi, N. Ansari, W. Su, "Reversible data hiding," *IEEE Transactions on Circuits and Systems for Video Technology*, 16 (2006), pp. 354-362. DOI. 10.1109/TCSVT.2006.869964

[11] D.M. Thodi, J.J. Rodriguez, "Expansion embedding techniques for reversible watermarking," *IEEE Transactions on Image Processing*, 16 (2007), pp. 721-730. DOI. 10.1109/TIP.2006.891046

[12] P.Y. Tsai, Y.C. Hu, H.L. Yeh, "Reversible image hiding scheme using predictive coding and histogram shifting," *Signal Processing*, 89 (2009), pp. 1129-1143. DOI. 10.1016/j.sigpro.2008.12.017

[13] W.L. Tai, C.M. Yeh, C.C. Chang, "Reversible data hiding based on histogram modification of pixel differences," *IEEE Transactions on Circuits and Systems for Video Technology*, 19 (2009), pp. 906-910. DOI. 10.1109/TCSVT.2009.2017409

[14] W. Hong, T.S. Chen, "A local variance-controlled reversible data hiding method using prediction and histogram-shifting," *Journal of Systems and Software*, 83 (2010), pp. 2653-2663. DOI. 10.1016/j.jss.2010.08.047

[15] W. Hong, "Adaptive reversible data hiding method based on error energy control and histogram shifting," *Optics Communications*, 285 (2012), pp.101-108. DOI. 10.1016/j.optcom.2011.09.005

[16] D.C. Lou, C.H. Hu, "LSB steganographic method based on reversible histogram transformation function for resisting statistical steganalysis," *Information Sciences*, 188(2012), pp. 346-358. DOI. 10.1016/j.ins.2011.06.003

[17] D.C. Lou, C.L. Chou, H.Y. Wei, H.F. Huang, "Active steganalysis for interpolation-error based reversible data hiding," *Pattern Recognition Letters*, 34(2013), pp. 1032-1036. DOI. 10.1016/j.patrec.2013.01.023

[18] Y. Hu, H.K. Lee, J. Li, "DE-based reversible data hiding with improved overflow location map," *IEEE Transactions on Circuits and Systems for Video Technology*, 19 (2009), pp. 250-260. DOI. 10.1109/TCSVT.2008.2009252

[19] X. Wu, N. Memon, "Context-based lossless interband compression-extending CALIC," *IEEE Transactions on Image Processing*, 9 (2000), pp.994-1001. DOI. 10.1109/83.846242

[20] M. Weinberger, G. Seroussi, G. Sapiro, "The LOCO-I lossless image compression algorithm: principles and standardization into JPEG-LS," *IEEE Transactions on Image Processing*, 9 (2000), pp. 1309- 1324. DOI. 10.1109/83.855427

[21] P. Howard, F. Kossentini, B. Martins, S. Forchhammer, W. Rucklidge, "The emerging JBIG2 standard," *IEEE Transactions on Circuits and Systems for Video Technology*, 8 (1998), pp. 838-848. DOI. 10.1109/76.735380

[22] USC-SIPI image database. Available: http://sipi.use.edu/database/.

Jeanne Chen received her B. Sc in Computer Science from the University of Saskatchewan, Canada in 1982 and M. Sc. from the University of Louisiana in 1985. Currently, she is an associate professor in the Department of Computer Science and Information Engineering at National Taichung University of Science and Technology, Taichung, Taiwan. Her research interests include image compression, steganography, image cryptosystem and bioinformatics.

Tung-Shou Chen received his B.S. and Ph.D. degrees from National Chiao Tung University in 1986 and 1992, respectively, both in Computer Science and Information Engineering. Currently, he is a professor and head of the Department of Computer Science and Information Engineering at National Taichung University of Science and Technology, Taichung, Taiwan. His current research interests include data mining, image cryptosystems, steganography, image compression and bioinformatics

.

Wien Hong received his M.S. and Ph.D. degree from the State University of New York at Buffalo, USA in 1994 and 1997, respectively. He has also published several books on Java and Matlab. He is currently a professor in the Department of Information Management at Yu-Da College of Business, Taiwan. His research interests include steganography, watermarking and image compression

Gwoboa Horng received the B.S. degree in Electrical Engineering from National Taiwan University in 1981 and the M.S. and Ph.D. degrees from University of Southern California in 1987 and 1992 respectively, all in Computer Science. Since 1992, he has been on the faculty of the Department of Computer Science and Engineering at National Chung-Hsing University, Taichung, Taiwan, R.O.C. His current research interests include artificial intelligence, cryptography and information security.

Han-Yen Wu received his M.S. degree from Yu-Da College of Business, Department of Information Management, Taiwan, in 2013. Currently, he is working towards his Ph.D. degree at National Chiao-Tung University, Graduate School of Computer Science, Taiwan. His current research interests include data hiding and reversible data hiding in encrypted.

Chih-Wei Shiu received his M.S. degree from Yu-Da College of Business, Department of Information Management, Taiwan, in 2009. Currently, he is working towards his Ph.D. degree at National Chung-Hsing University, Graduate School of Computer Science and Engineering, Taiwan. His current research interests include game development, data compression and steganography.