DOUBLE-THRESHOLD REVERSIBLE DATA HIDING

Guorong Xuan1, Yun Q. Shi2, Jianzhong Teng1, Xuefeng Tong1, Peiqi Chai1
1Dept. of Computer Science, Tongji University, Shanghai, China
2Dept. of ECE, New Jersey Institute of Technology, Newark, New Jersey, USA
grxuan@public1.sta.net.cn, shi@njit.edu

ABSTRACT

This proposed scheme reversibly embeds data into image prediction-errors by using histogram-pair method with double thresholds (embedding threshold and fluctuation threshold). The embedding threshold is used to select only those prediction-errors, whose magnitude does not exceed this threshold, for possible reversible data hiding. The fluctuation threshold is used to select only those prediction-errors, whose associated neighbor fluctuation does not exceed this threshold, for possible reversible data hiding. Only when both thresholds are satisfied the reversible data hiding is carried out. Image gray level histogram modification is conducted to shrink the image histogram towards the center to avoid underflow and/or overflow only when this is necessary. The required bookkeeping data are embedded together with pure payload for original image recovery late. The experimental results have demonstrated that the proposed scheme outperforms recently published reversible image data hiding schemes in terms of the highest PSNR of marked image vs. original image at given pure payloads.

Keyword: reversible image data hiding, prediction error, neighborhood fluctuation, histogram pair scheme, gray level histogram modification

1. INTRODUCTION

Reversibility requires that not only the hidden data can be extracted correctly but also the marked image be inverted back to the original cover image exactly after the hidden data extraction. Research on reversible, also called lossless, image data hiding has attracted great interests recently, which can be manifested by increasing number of publications on this subject, e.g., [1-8] (just a rather incomplete list). This is because reversible data hiding has found wide applications in image content authentication, covert communications, e-banking and e-government, to name a few.

In this paper, based on the optimum histogram pair reversible data hiding method by Xuan et al. [6,7], we propose a new reversible image data hiding scheme, which uses two thresholds: embedding threshold and fluctuation threshold. The proposed scheme has two major differences from [6,7]. First, data is embedded into the prediction-error of pixels values. There is no use of discrete wavelet transforms. Second, an additional threshold, i.e., fluctuation threshold is used in data embedding. It has outperformed in our experimental works over that [1-8] by a distinct margin.

The rest of this paper is organized as follows. The prediction error and neighbor fluctuation, principle of two thresholds, the proposed algorithm are presented in Sections 2, 3, and 4, respectively. Section 5 contains a simple example to illustrate how the proposed double-threshold method runs. Section 6 presents the block diagram of the proposed scheme. Experimental results are shown in Section 7. Summary is made in Section 8.

2. PREDICTION ERROR AND NEIGHBOR FLUCTUATION

Eight-neighbor is considered here. The proposed method can also work with other types of neighbor.

(1) Prediction Error: \( E \)

Formula 1 demonstrates a central pixel, \( x \), which is under consideration for possible reversible data embedding; and its 8-neighbor pixels are \( x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8 \). The prediction of the central pixel using its 8-neighbor pixels as shown in Formula (2), i.e., a weighted average of the 8-neighbor pixels. The prediction-error is defined as the difference between the central pixel \( x \) and the prediction from its 8-neighbor pixels as shown in Formula (3).

\[
E = x - \bar{x} = x - \left\lfloor \frac{1}{6} \sum_{i=1}^{8} x_i \right\rfloor
\]

\[
\bar{x} = \left\lfloor \frac{1}{6} \sum_{i=1}^{8} x_i \right\rfloor
\]

\[
E = x - \bar{x}
\]

(2) Neighbor fluctuation: \( F \)

The neighbor fluctuation is defined as the weighted average of difference between 8-neighbor pixels and the prediction value \( \bar{x} \). That is,

\[
F = \frac{2}{3} \left( (x_2 - \bar{x})^2 + (x_4 - \bar{x})^2 + (x_5 - \bar{x})^2 + (x_7 - \bar{x})^2 \right)
\]

\[
+ \frac{1}{3} \left( (x_1 - \bar{x})^2 + (x_3 - \bar{x})^2 + (x_6 - \bar{x})^2 + (x_8 - \bar{x})^2 \right)
\]

3. PRINCIPLE OF DOUBLE THRESHOLDING

The optimum histogram pair reversible data hiding method proposed by Xuan et al. [6,7] is applied to the histogram of the prediction error defined in Section 2. In data embedding, only a part of the prediction-error histogram, whose prediction errors are not larger than the embedding threshold \( T \), is possibly used to embed data. If data embedding begins with \( T>0 \) and the to-be embedded data sequence has not been embedded completely, the remaining data sequence will be embedded
into –T. In general, the embedding sequence, called t-sequence, will be T, -T, T-1, -(T-1),…,S, where S is called the stop point. If the data embedding begins with T, and T>0, then the t-sequence is: -T, T-1, -(T-1),…,S. Note that either T or –T is start point for data embedding. The stop point S is not necessarily equal to 0, and it can be positive or negative. In [6,7], it has been pointed out, S=0 does not necessarily lead to the highest PSNR for a given pure payload because embedding data into S=0 means that more part of histogram needs to be moved, hence, causing more changes to the marked image. The fluctuation value F, defined above, reflects the fluctuation existing in neighbor pixels, and is compared with fluctuation threshold T_F. If F<T_F, the prediction error, hence the corresponding central pixel, x, is selected for data embedding. Otherwise, the central pixel is not used for data embedding. It is expected that embedding data to less fluctuated image area will lead to high PSNR of marked image. This expectation has been verified by our experiments.

To simplify the discussion, we discuss the case in which the prediction-error is restricted to be positive. (For the case of negative prediction-error, the program runs accordingly.) The proposed method using double-threshold works as follows. We scan all of the image pixels in a certain sequence. When meet a pixel, whose prediction error E is equal to or less than the embedding threshold T (1st threshold), and at the same time the fluctuation value F associated with this pixel is equal to or less than the fluctuation threshold T_F (2nd threshold), data embedding is conducted. When the prediction error value E is greater than the embedding threshold T, or when and the fluctuation value F is larger than the fluctuation threshold T_F, then no data embedding for this pixel, and the program moves to the next pixel. The concrete algorithm is presented in the next section.

4. DATA EMBEDDING AND EXTRACTION

From Section 3, it is known that this proposed method is essentially an optimal neighbor-fluctuation guided prediction-error histogram pair reversible data hiding scheme. We describe the proposed algorithm as follows. Notation b_i represents a to-be embedded binary bit, and b_i ∈ {0,1}. Without loss of generality, assume that embedding threshold T is the starting point and S for the ending point of data embedding.

Choose T and T_P, let P=T. Step 1: Scan the whole image in a sequence, say, from left to right and from top to bottom. Use Formuale (1-4) to compute E and F for the current pixel, x, under consideration for data embedding.

If F<T_F, skip the current pixel, scan the next pixel, and use the current pixel is chosen for possible data embedding.

If P⊕0

\[ E = \begin{cases} 
E + b_i & \text{if } E = P \\
E + 1 & \text{if } E > P \\
E & \text{others}
\end{cases} \quad (5) \]

If P<0

The data embedding process is depicted in Fig. 1, where 3x3 windows are marked and moved from left to right as shown in Fig.1 (a)-(g). Fig. 1 (a) describes this portion of image’s 2-D array.

5. EXAMPLE OF PROPOSED METHOD

In order to illustrate the principle, one example is provided here to illustrate double thresholding principle. In this example, the 8 neighbor scenario is simplified as 4 neighbor. The corresponding 4 neighbor prediction, \( \overline{x} \), and 4 neighbor fluctuation, F, are defined below. For some notations used below, readers are referred to Formula (1).

\[ \overline{x} = \text{floor}\left(\frac{1}{4}\times(x2 + x4 + x5 + x7)\right) \quad (8) \]

\[ F = (x2-x)^2 + (x4-x)^2 + (x5-x)^2 + (x7-x)^2 \quad (9) \]

In this simple example, assume embedding threshold, T=0, (i.e., the data is to be embedded into pixels where the prediction error E is 0); fluctuation threshold T_F = 4, and the to-be embedded data is two bits: {1,0}.

The data embedding process is depicted in Fig. 1, where 3x3 windows are marked and moved from left to right as shown in Fig.1 (a)-(g). Fig. 1 (a) describes this portion of image’s 2-D array.

Fig. 1 Data embedding.
In Fig. 1 (b), the first 3x3 window is marked, surrounding the central pixel 158. Since the predicted central pixel is $\bar{x} = 158$ according Formula (8), we have $E = x - \bar{x} = 0$, which leads to $E = 0 = T$. Hence, this pixel may be used for data embedding. Further, it is found that $F = 1$, and $\bar{F} < T_F = 4$. Hence, according to embedding algorithm discussed in Section 4, we embed the first bit “1”, thus, according to Formula (5), we have $E = 1$, hence $x = 159$, as shown in Fig. 1 (c).

Next, the 3x3 window is shifted towards right by one pixel. As shown in the caption of Fig. 1 (c), one can easily find out that $E = 0$, $F > T_F$, hence, no data embedding in this position, the central pixel remains 159.

The 3x3 window is now moving towards right by one more position, as shown in Fig. 1 (d). Here, it is easily to see that $\bar{x} = 158$, $E = 2 > P$, $F = 1 < T_F$, hence no data embedding into this position. But, according to Formula (5), $E = E + 1 = 3$, hence $x = 160 \rightarrow 161$, as shown in Fig. 1 (e).

Next, the wind center moves now one pixel towards right, i.e., $x = 158$, as shown in Fig. 1 (e). It can be verified that $E = 3$, $F = 5 > T_F$, no data embedding.

For Fig. 1 (f), $\bar{x} = 161$, hence $E = 2 > T_F$, $F = 12 > T_F$, no data embedding.

For Fig. 1 (g), $\bar{x} = 162$, hence, $E = 1 < T_F$, the next bit “0” is embedded, and the central pixels remains 162.

Data extraction is carried out oppositely. The bits $\{0,1\}$ are extracted using the same 3x3 windows, as depicted from right to left as shown in Fig. 2 from (h) to (n).

For Fig. 2 (d), $\bar{x} = 160$, hence $E = 0 > T_F$, no data embedding.

For Fig. 2 (e), $\bar{x} = 161$, hence, $E = 1 < T_F$, the next bit “0” is embedded, and the central pixels remains 162.

6. BLOCK DIAGRAM

The block diagram of the proposed data embedding scheme is shown in Fig. 3. It is noted that the block diagram of data extraction is not provided here due to the paper space limit. It is expected that the data extraction is not difficult to perceive once the data embedding process is understood.

In order to overcome the underflow and/or overflow problem, i.e., pixel values beyond the range of $[0,255]$, the histogram modification method used in [6,7] is adopted in this method. Whenever underflow and/or overflow is detected, the histogram is narrowed down towards the center, and the corresponding information of histogram modification is recorded as bookkeeping data, which is to be embedded into the image together with the pure payload. While the location map [1] is two-dimensional the histogram shrinking is one-dimensional in nature. Hence, the amount of bookkeeping data with location map is often larger than that of histogram shrinking in general.

For a required pure data embedding rate, we need to find out a neighbor fluctuation threshold $T_F$ and an embedding threshold $T$ such that the resultant PSNR of the marked image is the highest. This optimization is characterized by Formula (10), and it can be...
The four test images and their histograms are shown in Fig. 5. The first three are 512x512 commonly used test images: Lena, Barbara, and Baboon, while the last is the 960x768 Woman image which is one of JPEG2000 test images. It is noticed that the histogram of the Woman image has two peaks at two ends of [0,255], which is hence considered as a necessary test image for performance evaluation of reversible data hiding. The data embedding parameters and results for these four images are shown in Table 1. Note that GL and GR stand for the amount of histogram shrinking towards the center from the left and right sides in order to avoid underflow and overflow, respectively. The performance in terms of PSNR vs. pure payload for Lena image of the proposed scheme compared with several reversible data embedding schemes [1-8] is shown in Fig. 6.

![Fig. 5 Four test images.](image)

Note: Parameter GR and GL mean how many gray levels in the histogram are shrunk from right side (GR) and left side (GL) towards the center.

## 8. SUMMARY

1. Optimal performance can be obtained by adjusting the fluctuation threshold, embedding threshold, and some other parameters of histogram pair scheme such as GL and GR.
2. When underflow and/or overflow is detected, the scheme starts to shrink the histogram towards its center accordingly, and the associated bookkeeping data will be embedded into the image together with pure payload.
3. In terms of PSNR of marked image with respect to original image vs. pure payload, the proposed method has demonstrated its superior performance over the existing schemes [1-8] as Lena, Barbra, Baboon and Woman images are used in the test.
4. The price paid with the proposed method is the program execution time is longer than that used in [6,7]. For instance, to embed 0.1 bpp into 512x512 Lena image, it needs 0.3 sec (with 52 dB in PSNR) for [6,7], while it needs 5 sec (with 55 dB in PSNR) for the proposed method, both on the same computational platform (CPU: Intel Pentium M processor 1.86GHz, RAM: 1.0GB, OS: Windows XP, MATLAB 6.5).

## REFERENCES