Optimal Histogram-pair and Prediction-error Based Reversible Data Hiding for Medical Images[1]

Xuefeng Tong[1], Xin Wang[1], Guorong Xuan[1], Shumeng Li[1], Yun Q. Shi[2]

[1]Dept of Computer Science, Tongji University, Shanghai, China
[2]ECE, New Jersey Institute of Technology, Newark, New Jersey, USA

Xin Wang <2014wangx@tongji.edu.cn>

Abstract. In recent years, with the development of application research on medical images and medical documents, it is urgent to embed data, such as patient's personal information, diagnostic information and verification information into medical images. Reversible data hiding for medical images is the technique of embedding medical data into medical images. However, most existed schemes of reversible data hiding for medical images could not achieve high performance and high payloads. This paper presents a reversible data hiding scheme for medical images based on histogram-pair and prediction-error. As the prediction-error histogram of medical images, compared with the gray level histogram of medical images, is more in line with quasi-Laplace distribution, histogram-pair and prediction-error based method could achieve high performance. We adjust the following four thresholds for optimal performance: embedding threshold, fluctuation threshold, left- and right-histogram shrinking thresholds. The left- and right-histogram shrinking thresholds are used not only to avoid underflow and/or overflow but also to achieve optimum performance. Compared to previous works, the proposed scheme has significant improvement in embedding capacity and marked image quality for medical images.

Keywords: Medical image, reversible data hiding, histogram-pair, prediction-error.

1 Introduction

With the rapid development of information technology, medical information, such as medical images and electronic patient records, is digitized and transmitted conveniently and quickly among patients, medical professionals and medical institutions through the Internet. With these benefits, there are some issues that need to be addressed. First of all, patients' privacies should be preserved in open networks [1]. Hence, embedding patients' information data into medical images would be one of the effective methods for protecting privacies. Second, the cover image is supposed

1 This research is largely supported by Shanghai City Board of education scientific research innovation projects (12ZZ033) and National Natural Science Foundation of China (NSFC) on projects No.90304017 and No.61175014.

to be able to be recovered without any loss after the hidden information has been extracted in order to introduce any interference to diagnosis. To sum up, high quality, high capacity, authentication and reversibility are the main requirements for data hiding in medical images [1]. Therefore, it is important to develop an efficient reversible data hiding schemes to embedding data into medical images.

In the past two decades, a variety of reversible data hiding schemes have been proposed. Tian [2] proposed a technique of pixel-value difference expansion (DE) by performing arithmetic operations on pairs of pixel to explore hide-able spaces. Kim et al. [3] proposed an enhanced pixel-value difference expansion method achieving higher data hiding capacities by location map. With the DE-based schemes, data embedding capacity is limited to 0.5 bits per pixel (bpp) if the data hiding is done by once. Besides, Ni et al. [4] proposed a reversible data hiding method based on histogram shifting which shifts the part of the histogram between the maximum and minimum points to the right side by one unit to create a gap for hiding the data. This method enhanced the quality of marked image and has also been widely investigated and developed. And then Fallahpour et al. [5] proposed an improved scheme which dividing the image into tiles and inserting data into the gap created by histogram shifting. This method successfully improved the data hiding capacity and enhanced the marked image quality and was applied into reversible data hiding for medical images [6]. Later, reversibly embedding data into prediction-error [7] is another effective scheme to largely boost embedding effectiveness in terms of the peak signal to noise ratio (PSNR) of the image with data hidden with respect to the original image versus data embedding rate. Xuan et al. [8] proposed a scheme to reversibly embed data into image prediction-errors by using histogram-pair method with the following four thresholds: embedding threshold, fluctuation threshold, left- and right- histogram shrinking thresholds. This scheme significantly improves the data hiding payload and marked image quality.

In this paper, we propose a reversible data hiding method based on histogram-pair and prediction-error for medical images. Compared with previous work [6], the proposed method achieves high data hiding capacity and high marked image quality.

The rest of this paper is organized as follows: Section 2 describes the proposed method. Experimental results and analysis are presented in Section 3. The conclusion is made in Section 4.

2 Proposed Method

The main idea of the reversible data hiding method based on optimal histogram-pair and prediction-error is to select histogram-pairs in the image prediction-error histogram and then shift the prediction-error of those pixels within the histogram-pair frequency range by one level, towards the minimum frequency level in the histogram-pair.

In this section, a simple example to explain the principle of histogram-pair in the first place and then the prediction-error is mentioned below. Next, we illustrate the four thresholds. After that, the optimal algorithm is presented. Finally, the data embedding and data extracting algorithm are presented in detail.

2.1 Histogram-Pair

We present a very simple example for the purpose of understanding the fundamental principle of histogram-pair reversible data hiding.

In Fig. 1(a), the original image containing only 9 pixels has 4 different gray values: $\{6,7,8,9\}$. Its histogram is [h(6),h(7),h(8),h(9)]=[4,3,1,1]. We suppose the data to be embedded are four bits: [0,1,1,0].

In Fig. 1(b), to avoid overflow or underflow after embedding data, histogram modification is executed. A pixel with gray value 9 is changed to 8. The information of this change called bookkeeping data will also be embedded into image in order to recover the original image reversibly as the hidden data are extracted late. So the histogram is changed to [h(6),h(7),h(8),h(9)]=[4,3,2,0].

In Fig. 1(c), we create a histogram-pair [h(6),h(7)]=[4,0] by shifting three pixels' gray values form 7 to 8 and two pixels' gray values from 8 to 9. Hence, the histogram is changed to [h(6),h(7),h(8),h(9)]=[4,0,3,2]. The histogram-pair [h(6),h(7)]=[4,0] can be used to embed data.

In Fig. 1(d), four bits [0,1,1,0] are embedded into this histogram-pair and the histogram-pair [h(6),h(7)]=[4,0] changes to [h(6),h(7)]=[2,2]. After four bits embedded, the histogram changes to [h(6),h(7),h(8),h(9)]=[2,2,3,2]. It's worth pointing out that the bookkeeping data need to be embedded into the image for the image recovery late and this detail is not shown here.

6	7	6
6	8	7
7	6	9

(a)

6	7	6
6	8	7
7	6	8

(b)

6	8	6
6	9	8
8	6	9

(c)

6	8	7
7	9	8
8	6	9

(d)

Fig. 1. Simple example of data hiding by histogram-pair (a) original image,

(b) histogram modification, (c) creation of histogram-pair, (d) after data embedded.

2.2 Prediction-Error

In the proposed method, we choose a pixel, x, for embedding a bit and consider its eight-neighbor, x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, shown in Eq. 1. Because the fluctuation and prediction-error could not be calculated by the eight-neighbor in image edge part, in a M×H image, we scan those pixels, (x,y), $2 \le x \le M-1$, $2 \le y \le H-1$, by raster order.

$$\begin{bmatrix} x_1 & x_4 & x_6 \\ x_2 & x & x_7 \\ x_3 & x_5 & x_8 \end{bmatrix} \tag{1}$$

The prediction-error, P_E, defined as Eq. 2. The average value of eight-neighbor, $\bar{x}$, could be seen as the prediction of the central pixel. The prediction-error is the difference between the pixel to be embedded and the average value of its eight-neighbor pixels. Due to the spatial correlation of the image pixels, this difference is normally rather small. The prediction-error histogram obeys quasi-Laplace distribution. For this reason, the prediction-error and histogram-pair based method is more suitable for reversible data hiding.

$$P_E = x - \bar{x} \quad where \quad \bar{x} = \left\lfloor (1/12)\left(\sum_{i=1,3,6,8} x_i + \sum_{i=2,4,5,7} 2x_i \right) \right\rfloor \tag{2}$$

2.3 Four Thresholds

1. Fluctuation threshold, T_F

Before embedding data, we calculate the fluctuation F of each candidate pixel using Eq. 3 from its surrounding eight-neighbor and compare the fluctuation with the fluctuation threshold T_F. Then, All pixels in the image are divided into two parts. One part of pixels whose fluctuation is no less than the fluctuation threshold would be untouched. The other part of pixels can possibly be embedded. In this way, we could embed data in smooth areas of the original image to improve the quality of marked image.

$$F = (1/3)\left(\sum_{i=1,3,6,8} \left(x_i - \bar{x}\right)^2 + \sum_{i=2,4,5,7} 2\left(x_i - \bar{x}\right)^2 \right) \tag{3}$$

2. Embedding threshold, T

We select the embedding threshold T and generate a pair of thresholds, positive embedding threshold T_P and negative embedding threshold T_N, by T. Next, the histogram-pair method is used to embed data. Different from previous works, the histogram-pairs are not chosen from the gray level histogram but the prediction-error histogram. Each pixel under consideration, which has satisfied the fluctuation threshold, is changed or not by comparing the positive or negative embedding threshold with the corresponding the positive or negative value of the pixel's prediction-error. All details are shown in Eq. 7.

3. Left- and Right-histogram shrinking thresholds, T_L ,T_R

Before embedding data, left-histogram shrinking threshold T_L is used to shrink the gray level at left histogram by T_L to prevent underflow after data embedding. And right-histogram shrinking threshold T_R is used to shrink the gray level at right histogram by T_R to prevent overflow after data embedding. When shrinking the histogram, we record shrinking pixels as bookkeeping data for late recovering the original image reversibly. Because the bookkeeping data will be embedded into the host image, we must select the optimal shrinking gray values to enhance the pure payload and the quality of the marked image. The left-histogram shrinking cost of each gray value is calculated by the function, shown in Eq. 4. The right-histogram shrinking cost of each gray value is calculate by another function, shown in Eq. 5. In the two functions, w_1 and w_2 are two weight coefficients, x is the gray value and h(x) represents the number of pixels whose gray value equals to x. And then, some of the gray values with smaller shrinking costs are selected as left- and right-histogram shrinking gray values.

$$f_L(x) = w_1 \times \big(h(x) + h(x+1)\big) + w_2 \times \sum_{i=0}^{x} h(i),\ 0 \le x < 255 \tag{4}$$

$$f_R(x) = w_1 \times \big(h(x-1) + h(x)\big) + w_2 \times \sum_{i=x}^{255} h(i),\ 0 < x \le 255 \tag{5}$$

Note that some common test images, e.g., Lena, shown in Fig. 2, Barbara, and Airplane, have two ends of their histogram being zero, whereas medical images, e.g., Im3, shown in Fig. 3, with both sides of its histogram having peaks, shown in Fig. 4. Hence, reversible data hiding in medical images frequently may lead to underflow and/or overflow. Different from the previous works, in the proposed method the left- and right-histogram shrinking thresholds are adjusted not only to prevent underflow and/or overflow but also to achieve the optimal performance in reversible data hiding. The proposed scheme with left- and right-histogram shrinking thresholds is especially suitable for reversible data hiding in medical images.

Fig. 2. Lena (512×512)

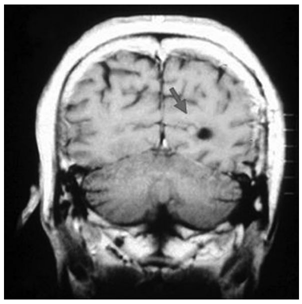


Fig. 3. Im3 (512×512)

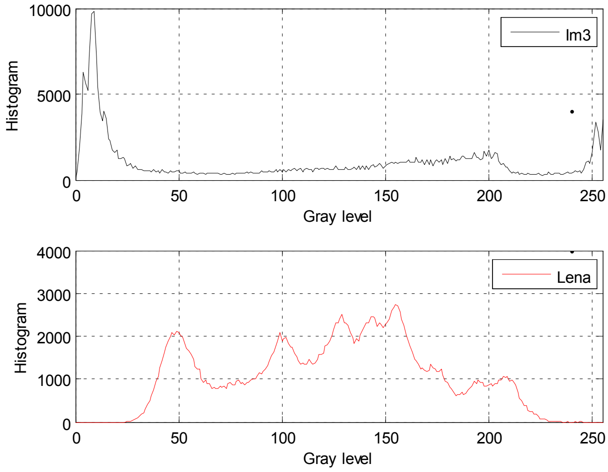


Fig. 4. Histogram of Im3 image and Lena image

2.4 Optimal Algorithm

Under the constraints of no underflow and/or overflow and the requirement of the given payloads, we adjust the four thresholds: embedding threshold T, fluctuation threshold T_F, left- and right-histogram shrinking thresholds T_L and T_R, for optimal marked image quality, shown in Eq. 6.

$$[T, T_F, T_L, T_R] = \underset{\substack{\text{neither underflow nor overflow} \\ \text{meeting embedding capacity}}}{\arg\max} \left[PSNR(Payload)\right] \quad (6)$$

2.5 Data Embedding Algorithm

First of all, we have the original image I of size M×H, each pixel grayscale value x∈[0,255], and to be embedded data D (length L) and then select a set of four thresholds, T, T_F, T_L and T_R. As mentioned before, we shrink the histogram of the original image towards the center by T_L and T_R and record the bookkeeping data before data embedding. Then, a pair of positive embedding threshold T_P and negative embedding threshold T_N, as said in Section 2.3, are generated from T. For example, if the algorithm embeds data in T=4, we scan the image and embed data into both T_P=4 and T_N=-4 corresponding with the scanned pixel's prediction-error. If the data have not been completely embedded yet, after data embedded into T_P=4 and T_N=-4, the next pair of T_P=3 and T_N=-3 is generated. In this way, the data would be embedded in an order of {4,-4},{3,-3},{2,-2},{1,-1},{0} or {-4,3},{-3,2},{-2,1},{-1,0}. In addition, It is not necessary that the data embedding algorithm has to end up at {0}. It can end up at any value in the sequence prior to 0 as long as the embedding capacity meets the requirement and the value is recorded as stop point S. With raster order scanning pixels, the data embedding algorithm is presented as follows:

Input: original image I, to be embedded data D (length L), four thresholds T, T_F, T_L and T_R

Output: marked image I', stop point S, stop pixel P

Procedures:

- Step 1. For the scanned pixel, We first calculate the pixel's fluctuation F and then compare it with the fluctuation threshold T_F. (1) If $F<T_F$, we carry out the next step. (2) If $F\geq T_F$, we implement step 4.
- Step 2. We calculate the pixel's prediction-error and then compare it with the corresponding positive or negative embedding threshold T_P or T_N. Next, because the prediction-error is directly related to the scanned pixel, shown in Eq. 2, we embed a bit into the pixel or expand the pixel by one level through altering the prediction-error, as shown in Eq. 7 where b stands for the bit to be embedded. After that, we execute the next step.
- Step 3. (1) If the length of embedded data reaches the L, we record the last T_P or T_N as the stop point S and the stop pixel P and then stop this algorithm. (2) If the length of embedded data don't reach the L, we execute the next step.
- Step 4. (1) If the scanned pixel is the last pixel, we choose the first pixel of a scan and select the next pair of positive and negative embedding thresholds T_P and T_N to embed data. Then, we implement Step 1. (2) If the scanned pixel is not the last pixel, we choose next pixel in raster order. Then, we implement Step 1.

$$P_E = \begin{cases} P_E - 1, & if\ P_E < T_N \\ P_E - b, & if\ P_E = T_N \\ P_E, & if\ T_N < P_E < T_P \\ P_E + b, & f\ P_E = T_P \\ P_E + 1, & if\ P_E > T_P \end{cases} \tag{7}$$

2.6 Data Extracting Algorithm

In data extracting, we recover the original image and extract data by stop point S, stop pixel P, marked image I', embedding threshold T and fluctuation threshold T_F. A set of T_P and T_N is derived from T and S. With reverse raster order scanning from stop pixel P, the data extracting algorithm is presented as follows.

Input: marked image I', embedding threshold T, fluctuation threshold T_F, stop point S, stop pixel P

Output: original image I, data D

Procedures:

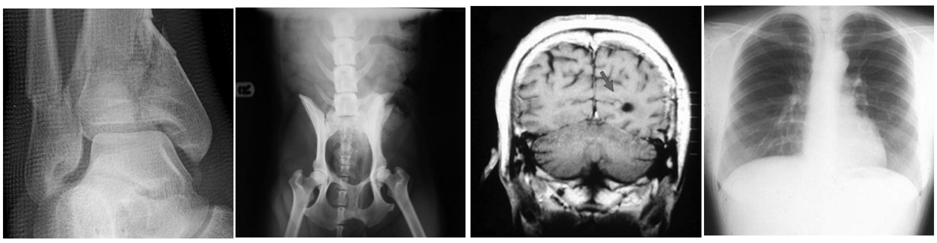
- Step 1. We calculate the scanning pixel's fluctuation F and then compare F with fluctuation threshold T_F. (1) If $F<T_F$, we execute the next step. (2) If $F \geq T_F$, we implement step 4.
- Step 2. The pixel's prediction-error is calculated and compared with the positive embedding threshold T_P and the negative embedding threshold T_N generated from T, as shown in Eq. 2 and Eq. 8 where b stands for the extracted bit. In this way, the bit is extracted and the pixel's value are recovered. After that, we execute the next step.
- Step 3. (1) If all of embedded data have been extracted, we stop data extracting algorithm. (2) If all of embedded data have not been extracted, we implement the next step.
- Step 4. (1) If the pixel is the first pixel which is also the last extracting pixel, we select the next pair of T_P and T_N, by $T_P=T_P+1$ and $T_N=T_N-1$. We choose the last pixel which is the first extracting pixel and then execute step 1. (2) If the pixel is not the first pixel, we choose next pixel in reverse raster order and then execute step 1.

After extracting the embedded data D of the length L, reversibly, we use the bookkeeping data to recover the original image reversibly.

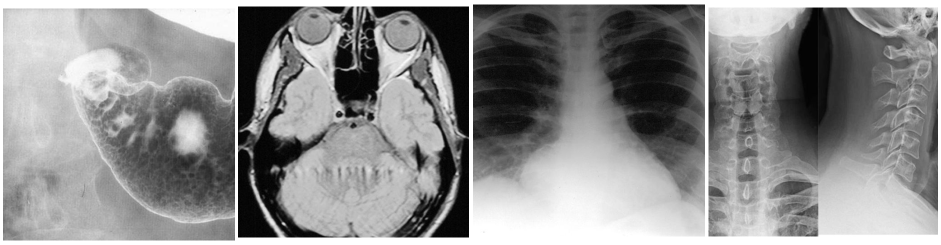
$$P_E = \begin{cases} P_E + 1, & \quad if\ P_E < T_N - 1 \\ P_E + 1, \quad b = 1 & \quad if\ P_E = T_N - 1 \\ P_E, \quad b = 0 & \quad if\ P_E = T_N \\ P_E, & \quad if\ T_N < P_E < T_P \\ P_E, \quad b = 0 & \quad if\ P_E = T_P \\ P_E - 1, \quad b = 1 & \quad if\ P_E = T_P + 1 \\ P_E - 1, & \quad if\ P_E > T_P + 1 \end{cases} \tag{8}$$

3 Experimental Results and Analysis

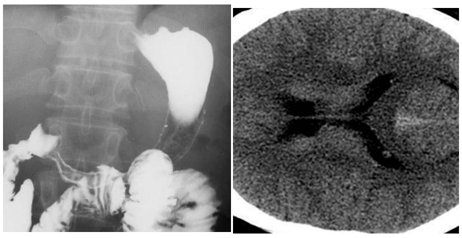
We have implemented the proposed method and compared the performance of the proposed method with algorithms in [6] on a variety of medical images, shown in Fig. 5. The size of the original medical images is 512×512 with 8-bit depth.



Im1 Im2 Im3 Im4



Im5 Im6 Im7 Im8



Im9 Im10

Fig. 5. Ten original medical images

The optimal four thresholds: T, T_F, T_L and T_R utilized, the resultant PSNR achieved on the ten test images by applying the optimal histogram-pair and prediction-error reversible data hiding method with various data embedding rates ranging from 0.01 bpp to 0.7 bpp are listed in Table 1. As said in Section 2.5, S is the stop value of the embedding threshold T. As the table shows, our presented method has high quality of ten marked images with embedding rate from 0.01 bpp to 0.7 bpp.

It is also observed that as low embedding rate is between 0.01 bpp and 0.1 bpp, T_L and/or T_R will not be all zero in most of ten medical images, therefore we have to shrink the histogram to avoid underflow and/or overflow. Whereas the T_L and T_R are all zero for Lena with data embedding rate not larger than 0.7 bpp, shown in [8]. Above all, overflow and/or underflow in reversible data hiding in medical images is more likely to occur compared with reversible data hiding into common images. Note that the proposed method with histogram shrinking not only avoids the problem but also works for optimal performance. It is suitable for reversible data hiding for medical images in particular.

Table 1. PSNR and optimal thresholds on ten test images with different payloads

Im1												
bpp	**0.01**	**0.02**	**0.03**	**0.04**	**0.05**	**0.1**	**0.2**	**0.3**	**0.4**	**0.5**	**0.6**	**0.7**
PSNR	**71.09**	**65.29**	**62.14**	**58.94**	**58.16**	**54.89**	**51.14**	**48.83**	**45.58**	**44.12**	**42.21**	**40.86**
T	**0**	**0**	**0**	**-1**	**-1**	**-1**	**-1**	**-1**	**-2**	**-2**	**-3**	**3**
T_F	**1**	**12**	**20**	**9**	**12**	**32**	**90**	**435**	**671**	**527**	**286**	**705**
T_L	**0**	**0**	**0**	**1**	**1**	**1**	**1**	**1**	**2**	**2**	**3**	**3**
T_R	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
S	**0**	**0**	**0**	**0**	**-1**	**-1**	**-1**	**-1**	**0**	**0**	**0**	**0**
Im2												
bpp	**0.01**	**0.02**	**0.03**	**0.04**	**0.05**	**0.1**	**0.2**	**0.3**	**0.4**	**0.5**	**0.6**	**0.7**
PSNR	**70.02**	**66.78**	**65.06**	**63.8**	**62.86**	**59.93**	**56.56**	**54.64**	**52.82**	**50.99**	**48.45**	**46.53**
T	**0**	**0**	**0**	**0**	**0**	**1**	**-1**	**-1**	**-1**	**-1**	**1**	**-2**

T_F	1	4	4	4	4	5	5	8	16	44	78	205
T_L	0	0	0	0	0	1	1	1	1	1	1	2
T_R	0	0	0	0	0	0	0	0	0	0	0	0
S	0	0	0	0	0	-1	-1	0	0	0	0	1
Im3												
bpp	0.01	0.02	0.03	0.04	0.05	0.1	0.2	0.3	0.4	0.5	0.6	0.7
PSNR	70.68	67.1	65.24	58.7	58.67	57.42	55.34	53.76	52.34	50.97	48.03	46.71
T	0	-1	-1	0	0	0	0	-1	-1	-1	1	-2
T_F	4	4	16	72	134	32	54	13	42	177	396	493
T_L	0	1	1	0	0	0	0	1	1	1	1	2
T_R	0	0	0	1	1	1	1	1	1	1	2	2
S	0	-1	-1	0	0	0	0	-1	-1	0	-1	-2
Im4												
bpp	0.01	0.02	0.03	0.04	0.05	0.1	0.2	0.3	0.4	0.5	0.6	0.7
PSNR	71.08	68.03	66.36	65.12	64.16	60.97	57.92	56.14	54.73	53.56	52.6	51.75
T	0	0	0	0	0	0	-1	-1	-1	-1	-1	-1
T_F	1	1	1	1	1	3	4	5	8	12	21	41
T_L	0	0	0	0	0	0	0	0	0	0	0	0
T_R	0	0	0	0	0	1	1	1	1	1	1	1
S	0	0	0	0	0	0	-1	0	-1	0	-1	-1
Im5												
bpp	0.01	0.02	0.03	0.04	0.05	0.1	0.2	0.3	0.4	0.5	0.6	0.7
PSNR	70.35	66.75	64.74	63.22	62.06	58.53	54.94	52.72	51.16	49.82	47.75	46.71
T	-1	-1	-1	1	1	-2	-2	1	-1	-1	1	-2
T_F	3	4	5	7	8	21	58	61	61	216	157	118
T_L	0	0	0	0	0	0	0	0	0	0	0	0
T_R	0	1	1	1	1	1	1	1	1	1	2	2
S	0	0	-1	-1	-1	-1	0	-1	0	-1	-1	1
Im6												
bpp	0.01	0.02	0.03	0.04	0.05	0.1	0.2	0.3	0.4	0.5	0.6	0.7
PSNR	68.52	65.82	64.19	62.89	61.97	58.84	55.05	52.81	50.38	48.88	47.55	46.12
T	-3	-3	-2	-2	-2	-2	1	1	1	1	-2	-2
T_F	11	17	8	10	11	21	33	158	130	181	177	1010
T_L	1	1	1	1	1	1	1	1	1	1	2	2
T_R	0	0	0	0	0	1	1	1	2	2	2	2
S	-3	-3	-2	-2	1	1	-1	-1	0	0	-1	-1
Im7												
bpp	0.01	0.02	0.03	0.04	0.05	0.1	0.2	0.3	0.4	0.5	0.6	0.7
PSNR	69.83	66.44	64.62	63.07	62.07	58.4	54.5	51.6	48.81	46.84	45.7	43.21
T	-2	1	-2	-2	-2	1	-1	-1	-2	1	-2	-2
T_F	4	4	7	9	10	12	20	56	30	136	115	85
T_L	0	0	0	0	0	0	0	0	2	1	2	3
T_R	0	0	0	0	0	0	0	0	2	2	2	2
S	1	-1	-2	1	1	-1	-1	-1	-1	-1	0	-2
Im8												
bpp	0.01	0.02	0.03	0.04	0.05	0.1	0.2	0.3	0.4	0.5	0.6	0.7
PSNR	70.99	67.84	66.04	64.72	63.72	60.53	56.91	54.68	52.81	50.93	48.69	47.12
T	-1	0	-1	-1	-1	-1	-1	-1	-1	-1	-2	-2
T_F	1	2	2	3	4	6	9	16	33	116	70	178
T_L	0	0	0	0	0	0	0	0	0	0	0	0
T_R	0	0	0	0	0	0	0	0	0	0	0	1
S	0	0	0	0	0	0	-1	0	0	-1	0	-2
Im9												
bpp	0.01	0.02	0.03	0.04	0.05	0.1	0.2	0.3	0.4	0.5	0.6	0.7
PSNR	71.09	68.04	66.36	65.01	64.06	60.93	57.76	55.88	54.42	53.18	52.13	51.18
T	0	0	0	0	0	-1	-1	-1	-1	-1	-1	-1
T_F	1	1	1	2	2	3	5	7	10	16	28	71
T_L	0	0	0	0	0	0	0	0	0	0	0	0
T_R	0	0	0	0	0	0	0	0	0	0	0	0
S	0	0	0	0	0	-1	-1	-1	-1	-1	0	-1
Im10												
bpp	0.01	0.02	0.03	0.04	0.05	0.1	0.2	0.3	0.4	0.5	0.6	0.7
PSNR	59.99	59.94	59.88	59.46	58.54	55.35	52.01	50.01	46.79	45.02	44.32	42.22
T	0	0	0	0	0	-1	-1	-1	-1	-2	-2	-2
T_F	12	2	6	22	41	25	141	354	1800	1000	760	3100
T_L	0	0	0	0	0	1	1	1	2	2	2	3
T_R	1	1	1	1	1	1	1	1	2	2	2	3
S	0	0	0	0	0	-1	0	0	-1	-1	0	1

The performance comparison on two medical images, Im5 and Im10, in terms of PSNR versus pure payload among the proposed method, and shifted histograms of whole, 4- and 16-tile versions [6], defined as WSH, TSH-4 and TSH-16, are shown in Fig. 6 and Fig. 7. Compared with Ni et al. [4] and Fallahpour et al. [6], the proposed method could significantly improve the data hiding capacity and the marked image quality.

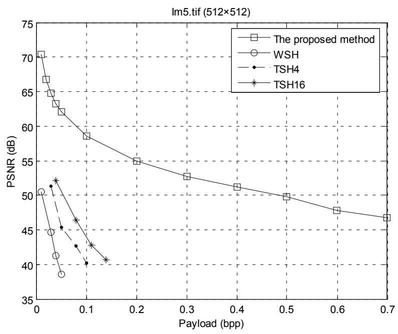


Fig. 6. Embedding for Im5

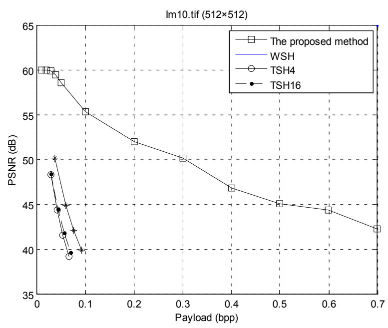


Fig. 7. Embedding for Im10

Different from shifting the histogram of the whole image and the histogram of the image sub-blocks method, our presented method selects histogram-pairs in prediction-error histogram and then shifts the histogram to embed data. The advantages of our presented method are illustrated as follows.

The histogram of prediction-error images obeys the quasi-Laplace distribution. Thus, it has the highest peak around zero prediction-error. A medical image, Im5, is shown in Fig. 8 (a). Its histogram is shown in Fig. 8 (b), which has peak value at gray level value 250. It is obvious that it will be difficult to apply the histogram-shifting method proposed in [4].

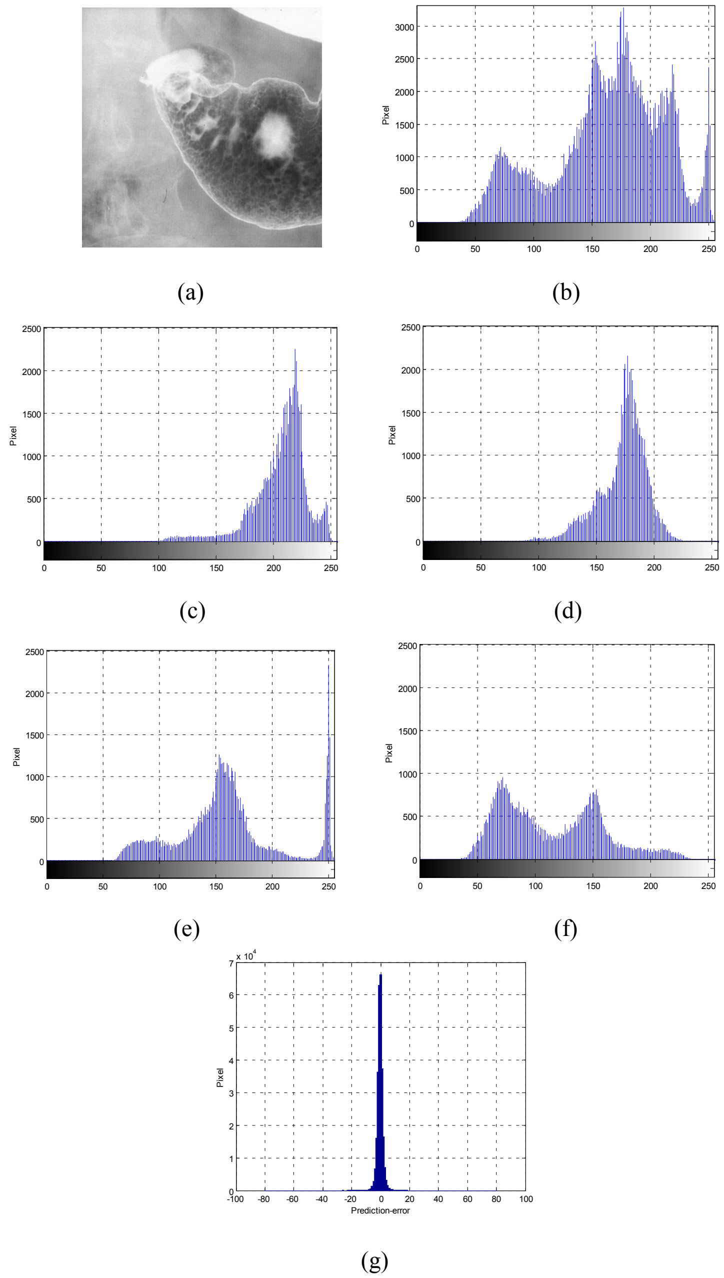


Fig. 8. Comparison of the histogram of the whole image, the histogram of the image sub-blocks and the prediction-error histogram (a) Im5, (b) The histogram of Im5, (c)(d)(e)(f) The histograms of the image sub-blocks in Im5, (g) The prediction-error histogram of Im5.

In [6], it was proposed to split the whole image into four parts. The histogram of each part is shown in Fig. 8 (c)(d)(e)(f), respectively. It is clear that this method is relatively easier to hide more data with higher PSNR than by using method reported in [4].

If the proposed method is used to reversibly embed data into this medical image, shown in Fig. 8 (a), the resultant histogram of the prediction-error image, shown in Fig. 8 (g), has a much high peak. Therefore, it is possible to embed much more data in the same medical image, as shown in Fig. 6 and Table 1.

Note that the histogram of medical images, compared with that of common images, has one or both sides having peaks. And hence it is more likely to lead to overflow and/or underflow. Histogram shifting based method should be used to handle this issue. The proposed method uses left- and right-histogram shrinking thresholds to avoid underflow and/or overflow. Furthermore, it is also used to achieve optimal performance.

Finally, our experimental works have demonstrated that the performance in reversible data hiding is superior than the prior art [6]. The average PSNR of ten marked images is 58.58 dB with the payload of 0.1 bpp and 46.24 dB with the payload of 0.7 bpp.

4 Conclusion

In spatial domain, several reversible data hiding methods for medical images, such as difference expansion and histogram shifting, have been reported in the literature [6][9]. In this paper, we apply the histogram-pair and prediction-error based data hiding method to reversible data hiding for medical images in spatial domain. It is known that the medical images often have high peaks at the two sides of the histogram, which causes more challenge than normal images do. Compared with [6], the proposed method achieves 12.18 dB higher PSNR at given embedding rate 0.1 bpp and enhances the average of maximum embedding rate in ten medical images from 0.16 bpp to more than 0.7 bpp.

We have shown that data hiding based on the histogram-pair and prediction-error is much better than histogram shifting method. The proposed method obviously improves the data hiding capacity and the marked image quality. This is mainly due to the fact that the prediction-error could better exploit smooth areas of medical images to improve the marked image quality. The prediction-error histogram obeys quasi-Laplace distribution and is suitable for embedding data. When embedding data, fluctuation threshold and embedding threshold is not necessarily the smaller the better but the optimal solution exists. This is a unique advantage in the proposed method.

Besides, reversible data hiding in medical images would be more likely to cause underflow and/or overflow compared with data hiding in common images on account of medical images with both sides of its histogram having peaks. Left- and right-histogram shrinking thresholds are used not only for avoiding underflow and/or overflow but also for optimum performance.

Acknowledgment

The author would like to thank Mehdi Fallahpour for providing the ten medical images used in this study.

References

1. G. Coatrieux, H. Maitre, B. Sankur, Y. Rolland, R. Collorec.: Relevance of watermarking in medical imaging. In IEEE-embs Information Technology Applications in Biomedicine, Arlington, USA, pp. 250-255(2000)
2. J. Tian.: Reversible data embedding using a difference expansion. IEEE Tarnsaction on Circuits and Systems for Video Technology 13(8):890-896(2003)
3. H. J. Kim, V. Sachnev,Y. Q. Shi, J. Nam, and H.G.Choo.: A novel difference expansion transform for reversible data embedding. IEEE Transaction on Information Forensics and Security, 3(3):456-465(2008)
4. Z. Ni, Y. Q. Shi, N. Ansari, and W. Su.: Reversible data hiding. IEEE Transaction on Circuits and Systems for Video Technology, 16(3):354-362(2006)
5. M. Fallahpour, M. H. Sedaaghi.: High capacity lossless data hiding based on histogram modification. IEICE Electronic Express, 4(7):205-210(2007)
6. M. Fallahpour, D. Megias, M. Ghanbri.: Reversible and high-capacity data hiding in medical images. IET Image Processing, 5(2):190-197(2011)
7. D. M. Thodi, J. J. Rodriguez.: Expansion embedding techniques for reversible watermarking. IEEE Transactions on Image Processing, 16(3):721-730(2007)
8. G. R. Xuan, X. F. Tong, J. Z. Teng, X. J. Zhang, Y. Q. Shi.: Optimal histogram-pair and prediction-error based image reversible data hiding. In:Y.Q. Shi, H.J. Kim, and F. Perez-Gonzalez (eds.) IWDW 2012.LNCS, vol.7809, pp.368-383(2013)
9. C. V. Kumar, V. Natarajan, S. S.Muraledharan.: Difference expansion based reversible data hiding for medical images. IEEE International Conference on Communication and Signal Processing pp.720-723(2014)