

# Reversible Data Hiding Based on CSD Data Representation

Wei-Sheng Lin , Cheng-Min Cho, Shuenn-Shyang Wang and Sz–Jiun Fan

Department of Electrical Engineering

Tatung University

40 Chungshan N. Road, 3<sup>rd</sup> Sec.

Taipei, Taiwan, R.O.C.

Email: [sswang@ttu.edu.tw](mailto:sswang@ttu.edu.tw)

**Abstract— In this paper, we propose a new reversible data hiding method which divides an input image into non-overlapping blocks and embeds data using canonical signed digit (CSD) data representation. To increase the data hiding capacity, the original image is divided into smaller blocks and then the CSD data are embedded by histogram modification. The CSD data representation has a minimum number of nonzero digits, thus it can decrease the alteration of the pixels when the hidden data is embedded. This method can recover the original image without any distortion from the marked image after the hidden data have been extracted. Experimental results show that the proposed method can achieve high data hiding capacity.**

## 1 Introduction

In recent years, as network being more and more developed, the digital multimedia such as images, audio clips or video streams are easily obtained from the Internet. It provides vast opportunities for hackers to steal useful information, thus security becomes an important issue [1-4]. Reversible data hiding is the technique that embeds hidden data inside an

image and later the hidden data can be retrieved so that the exact copy of the original image is recovered [2-12]. The reversible data hiding is crucial in some fields such as medical, astronomical, and military images due to legal reasons. It is also an emerging field for content authentication of images where the authentication information is embedded inside the image. To have high embedding capacity, it is desirable to embed more information inside the image under the requirement that the distortions due to data embedding are not visible. In the literature, most of the existing data hiding techniques are not reversible. For instance, the widely utilized spread-spectrum based data hiding methods [1] are not reversible owing to truncation error and round-off error. Fridrich *et al.* [2] developed a high capacity reversible data-embedding technique based on embedding message on bits in the status of group of pixels. De Vleeschouwer *et al.* [3] proposed a reversible data-embedding algorithm by circular interpretation of bijective transformations. Celik *et al.* [4] presented a high capacity, low distortion reversible data-embedding algorithm by compressing quantization residues. Ni *et al.* [9] have proposed

a histogram modification based lossless data hiding scheme. Xuan et al. [7] proposed the reversible data hiding algorithms that embedded data into the high-frequency wavelet coefficients, which uses the companding manner to prevent the pixel overflow and underflow. In this paper, we propose a new reversible data hiding method which divides an input image into non-overlapping blocks and embeds data using CSD data representation. We divide the original image into smaller blocks, in order to increase the data hiding capacity. The CSD data representation has a minimum number of nonzero digits, thus it can decrease the alteration of the pixels when the hidden data are embedded.

The organization of the rest of this paper is as follows. We describe the embedding algorithm and the extracting algorithm of the proposed reversible data hiding method in Section 2. Some experiment results are shown in Section 3, and conclude it in section 4.

## 2 Proposed Reversible Data Hiding Method

The canonical signed-digit representation (CSD) is a minimum weight signed-digit representation, i.e., it has a minimum number of nonzero digits, and it contains no adjacent nonzero digits. The conversion of a two's complement number into CSD representation can be done using Table 1 [12]. The  $b_i \in \{0,1\}$  are the bits of the two's complement number to be converted to CSD representation and the  $y_i \in \{0,1,\bar{1}\}$  are the CSD representation after the conversion, where  $\bar{1}$  denotes -1. The  $c_i$  is the carry generated in step  $i - 1$  and  $c_{i+1}$  is the carry out at step  $i$ . Due to CSD data representation has a minimum number of nonzero digits, we will convert hidden binary data into CSD

representation for data embedding. By this way, we can decrease the alteration of the pixels when the hidden data is embedded and achieve the improvement of PSNR.

Table 1 Right-to-Left CSD conversion table.

$b_{i+1}$	$b_i$	$c_i$	$y_i$	$c_{i+1}$
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	0	0
1	0	1	$\bar{1}$	1
1	1	0	$\bar{1}$	1
1	1	1	0	1

For data embedding, we will divide an input image into non-overlapping blocks of  $N \times N$  pixels. The objective of dividing the image into smaller blocks is to increase the data hiding capacity. Now, our reversible data hiding method is proposed and the hidden binary data can be embedded using the following steps.

- Step1: Convert the hidden binary data into CSD representation.
- Step2: Divide an input image into non-overlapping blocks of  $N \times N$  pixels. Generate its histogram for each block.
- Step3: In each histogram of image block, find the one maximum point  $h(p)$ , where  $p \in [0,255]$  and two minimum or zero points  $h(z_1)$ ,  $h(z_2)$ , where  $z_1, z_2 \in [0,255]$ . If the minimum point is greater than zero, recode the coordinate  $(i, j)$

of those pixels and the pixel grayscale as overhead bookkeeping information.

Step4: For each image block, perform histogram shifting. There are three possible cases as illustrated in Figure 1.

Case1: Assume  $z_1 < p < z_2$ . Move the whole part of the histogram  $h(x)$  with  $x \in (p, z_1)$  to the left by 1 unit. This means that all the pixel grayscale values satisfying  $x \in (p, z_1)$  are subtracted by 1. Move the whole part of the histogram  $h(x)$  with  $x \in (p, z_2)$  to the right by 1 unit. This means that all the pixel grayscale value satisfying  $x \in (p, z_2)$  are added by 1.

Case2: Assume  $p < z_1 < z_2$ . Move the whole part of the histogram  $h(x)$  with  $x \in (p, z_1)$  to the right by 1 unit. This means that all the pixel grayscale values satisfying  $x \in (p, z_1)$  are added by 1. Move the whole part of the histogram with  $h(x)$  to the right by 1 unit once again. This means that all the pixel grayscale value satisfying  $x \in (p, z_2)$  are added by 1.

Case3: Assume  $z_2 < z_1 < p$ . Move the whole part of the histogram  $h(x)$  with  $x \in (p, z_1)$  to the left by 1 unit. This

means that all the pixel grayscale values satisfying  $x \in (p, z_1)$  are subtracted by 1. Move the whole part of the histogram with  $h(x)$  to the left by 1 unit once again. This means that all the pixel grayscale value satisfying  $x \in (p, z_2)$  are subtracted by 1.

Step5: For each image block, we embed the CSD data by histogram modification for one of three cases.

Case1: Scan the image and modify the pixel whose grayscale value is  $p$  according to the to-be-embedded bit. If the to-be-embedded bit is "1", the pixel grayscale value  $p$  is changed to  $p+1$ . If the to-be-embedded bit is "-1", the pixel grayscale value  $p$  is changed to  $p-1$ . If the bit is "0", the pixel value remains  $p$ .

Case2: Scan the image and modify the pixel whose grayscale value is  $p$  according to the to-be-embedded bit. If the to-be-embedded bit is "1", the pixel grayscale value  $p$  is changed to  $p+1$ . If the to-be-embedded bit is "-1", the pixel grayscale value  $p$  is changed to  $p+2$ . If the bit is "0", the pixel value remains  $p$ .

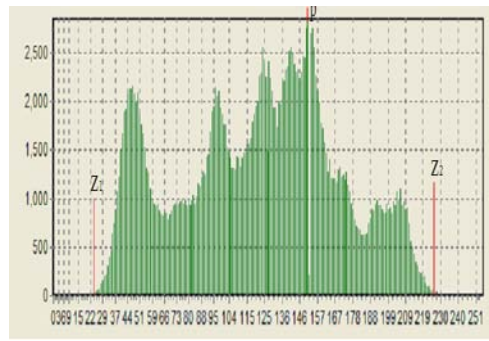
Case3: Scan the image and modify the pixel whose grayscale value is  $p$  according to the to-be-embedded bit. If the to-be-embedded bit is "1", the pixel grayscale value  $p$  is changed to  $p-1$ . If the to-be-embedded bit is "-1", the pixel grayscale value  $p$  is changed to  $p-2$ . If the bit is "0", the pixel value remains  $p$ .

Remark: The value  $p$ ,  $z_1$  and  $z_2$  of the maximum point and the minimum point or zero

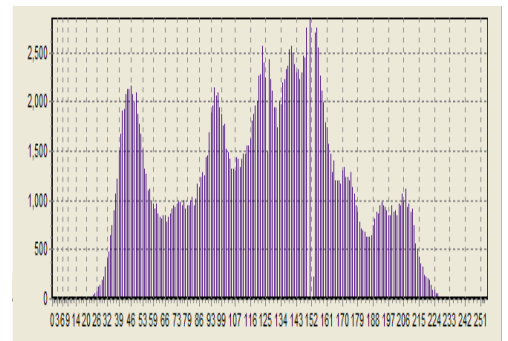
point will be treated side information that needs to be transmitted to the receiving side for data retrieval.



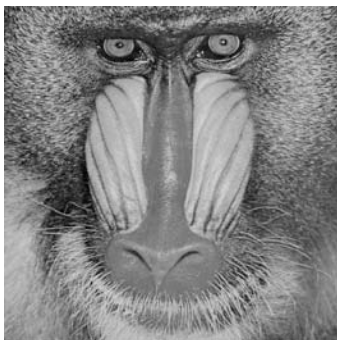
(a) Lena image



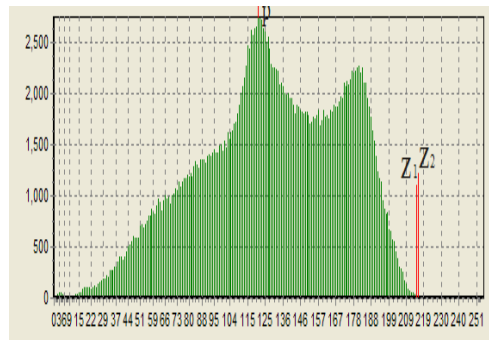
(a1) Histogram of the original image



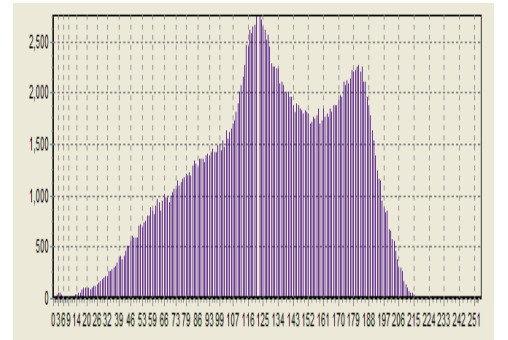
(a2) Histogram shifting of the Lena image



(b) Baboon image



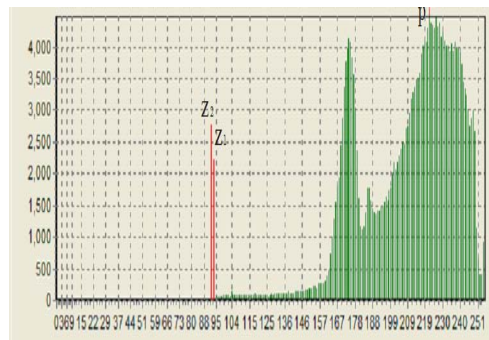
(b1) Histogram of the Baboon image



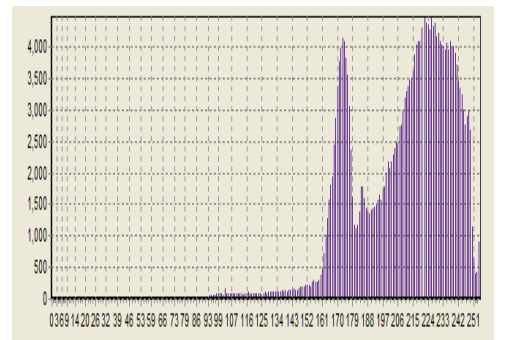
(b2) Histogram shifting of the original Baboon image



(c) Tiffany image



(c1) Histogram of the Tiffany image



(c2) Histogram shifting of the original Tiffany image

Figure 1 (a)-(c) Original image, (a1)-(c1) histogram of the original image, and (a2)-(c2) histogram shifting of the original image

To extract hidden binary data, the data extraction algorithm can be performed using the following steps.

Step1: Divide an input image into non-overlapping blocks of  $N \times N$  pixels. Generate its histogram for each block.

Step2: Scan each image block and perform the extraction of hidden CSD data according to one of three cases.

Case1: Once the maximum grayscale value  $p$  is met, the hidden data “0” is retrieved. If the grayscale value,  $p+1$  or  $p-1$  are met, the hidden data “1” or “-1” are respectively retrieved.

Case2: Once the maximum grayscale value  $p$  is met, the hidden data “0” is retrieved. If the grayscale value,  $p+1$  or  $p+2$  are met, the hidden data “1” or “-1” are respectively retrieved.

Case3: Once the maximum grayscale value  $p$  is met, the hidden data “0” is retrieved. If the grayscale value,  $p-1$  or  $p-2$  are met, the hidden data “1” or “-1” are respectively retrieved.

Step3: Scan each image block once again, perform histogram shifting and recover the original image block according to one of three cases.

Case1: If the pixels whose gray value lies between the maximum point  $p$  and the minimum point  $z_1$  is met, the gray value of those pixels will be added by 1. If the pixels whose gray value lies between the maximum point  $p$  and the minimum point  $z_2$  is met, the gray value of those pixels will be subtracted by 1.

Case2: If the pixels whose gray value lies between the maximum point  $p$  and the minimum point  $z_2$  is met, the gray value of those pixels will be subtracted by 1. If the pixels whose gray value lies between the maximum point  $p$  and the minimum point  $z_1$  is met, the gray value of those pixels will be subtracted by 1.

Case3: If the pixels whose gray value lies between the maximum point  $p$  and the minimum point  $z_2$  is met, the gray value of those pixels will be added by 1. If the pixels whose gray value lies between the maximum point  $p$  and the minimum point  $z_1$  is met, the gray value of those pixels will be added by 1.

Step4: Convert the extracted CSD data into the hidden binary data.

### 3 Experiment Results

The performance of our reversible data hiding method has been evaluated by a number of experiments on the 512x512 grayscale images shown in Figure 2(a)-(b). The hidden data to be embedded is the binary image of “Tatung university” shown in Figure 3, where the white pixel represents “1” and the black pixel represents “0”. The marked image is quantitatively measured by peak signal-to-noise ratio (PSNR) defined as

$$MSE = \frac{1}{mn} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} (s_{ij} - y_{ij})^2 \quad (1)$$

$$PSNR = 10 \times \log_{10} \left( \frac{255^2}{MSE} \right) \quad (2)$$

where  $s_{ij}$  is the grayscale value of  $(i, j)$  pixel of original image  $s$ ;  $y_{ij}$  is the grayscale value of  $(i, j)$  pixel of the marked image  $y$ ;  $MSE$  is the mean square error;  $m$  is the height of the image;  $n$  is the

width of the image; and 255 is the maximum value for 8-bit image.

For our proposed method, the experimental results under the same PSNR quality for different block size are shown in Table 2 - 3. It is seen that the proposed method can achieve high data hiding capacity when  $32 \times 32$  block size is adopted. Note that the data hiding capacity is the capacity minus overhead. As shown in Table 4, the proposed method achieves higher data hiding capacity than Ni et al. method [9] under the same PSNR. The experimental results of the marked images for our method are shown in Figure 4(a)-(b) after embedding the hidden data of Figure 3. Extensive simulations show that the embedding performance depends on the hidden data. For more zeros of the hidden CSD data, the higher PSNR can be achieved because there is little modification of the pixels.



(a) Lena image



(b) Barbara image

Figure 2 The original images



Figure 3 The Hidden data.

Table 2 Experimental results of proposed method under the same PSNR quality for different block sizes.

Image	Lena (512x512x8)			
Block size $N \times N$	PSNR (dB)	Capacity (bits)	Overhead (bits)	Capacity-Overhead
4x4	54.02	44537	309576	negative
8x8	54.02	17045	50856	negative
16x16	54.02	8517	9360	negative
32x32	54.02	5027	1848	3179
64x64	54.16	3410	408	3002
128x128	54.28	1922	96	1826
256x256	54.24	1378	24	1354

Table 3 Experimental results of proposed method under the same PSNR quality for different block sizes.

Image	Barbara (512x512x8)			
Block size $N \times N$	PSNR (dB)	Capacity (bits)	Overhead (bits)	Capacity-Overhead
4x4	54.28	46644	383952	negative
8x8	54.28	16226	58032	negative
16x16	54.28	7936	8712	negative
32x32	54.28	5009	2016	2993
64x64	54.13	2747	408	2339
128x128	54.26	1379	96	1283
256x256	54.22	1092	24	1068

Table 4 Experimental results of the Ni et al. method and the proposed method.

Image (512x512)	PSNR (dB)		Capacity (bits)		Overhead (bits)		Capacity-Overhead	
	Ni et al. method	Proposed method	Ni et al. method	Proposed method	Ni et al. method	Proposed method	Ni et al. method	Proposed method
Lena	54.02	54.02	2908	5027	16	1848	2892	3179
Barbara	54.28	54.28	2370	5009	16	2016	2354	2993



(a) Marked Lena image (54.02dB)



(b) Marked Barbara image (54.28 dB)

Figure 4 The marked image obtained after embedding the hidden data of Figure 3.

#### 4 Conclusions

In this paper, we propose a new reversible data hiding method which divides an input image into non-overlapping blocks and embeds data using CSD data representation. To increase the data hiding capacity, the original image is divided into smaller blocks and then the CSD data are embedded by histogram modification. The CSD data representation has a minimum number of nonzero digits, thus it can decrease the alteration of the pixels when the hidden data is embedded. This method can recover the original image without any distortion from the marked image after the hidden data have been extracted. Experimental results show that the proposed method can achieve high data hiding capacity.

#### 5 References

- [1] J. Cox, J. Kilian, T. Leighton, and T. Shamoan, "Secure spread spectrum watermarking for multimedia," *IEEE Trans. Image Process.*, vol. 6, no. 12, pp. 1673–1687, Dec. 1997.
- [2] J. Fridrich, M. Goljan, and R. Du, "Lossless data embedding—new paradigm in digital watermarking," *EURASIP J. Appl. Signal Processing*, vol. 2002, no. 2, pp. 185–196, Feb. 2002.
- [3] C. De Vleeschouwer, J. F. Delaigle, and B. Macq, "Circular interpretation of bijective transformations in lossless watermarking for media asset management," *IEEE Tran. Multimedia*, vol. 5, pp. 97–105, Mar. 2003.
- [4] M. U. Celik, G. Sharma, A. M. Tekalp, and E. Saber, "Reversible data hiding," in *Proc. Int. Conf. Image Processing*, vol. II, pp. 157–160, Sept. 2002,.
- [5] G. Xuan, J. Chen, J. Zhu, Y.Q. Shi, Z. Ni, and W. Su, "Lossless data hiding based on integer wavelet transform," *IEEE Workshop on Multimedia Signal Processing*. Marriott Beach Resort St. Thomas, US Virgin Islands, 9-11 Dec. 2002.
- [6] G. Xuan, Y.Q. Shi, Z.C. Ni, J. Chen, C. Yang, Y. Zhen, J. Zheng; "High capacity lossless data hiding based on integer wavelet transform," in *Pro. Int. Sym. Circuits and Systems*. Vd. 2, pp. II-29-32, May 2004.
- [7] G. Xuan, Y.Q. Shi, C. Yang, Y. Zheng, D. Zou, and P. Chai, "Lossless Data Hiding Using Integer Wavelet Transform and



Threshold Embedding Technique,” *IEEE Int. Conf. Multimedia and Expo*, PP. 1520-1523, July 2005.

- [8] J. Tian, “Reversible data embedding using a difference expansion,” *IEEE Tran. Circuits and Systems for Video Technology*, Vol. 13, Issue 8, pp. 890-896, Aug. 2003.
- [9] Z. Ni, Yun Q. Shi, N. Ansari, and W. Su, “Reversible data hiding,” *IEEE Tran. Circuits and Systems for Video Technology*, Vol. 16, No. 3, PP. 354-362, March 2006.
- [10] H.C. Huang, T.W. Chen, J.S. Pan, and J.H. Ho, “Copyright protection and annotation with reversible data hiding and adaptive visible watermarking,” *Second International Conference on Innovative Computing, Information and Control*, pp. 292 – 292, Sept. 2007
- [11] L. Yang, P. Hao, C. Zhang, “Progressive reversible data hiding by symmetrical histogram expansion with piecewise-linear haar transform,” *IEEE Int. Conf. Acoustics, Speech and Signal Processing*. Vol. 2, pp.II-265 - II-268, April 2007.
- [12] E. Backenius, E. Sail, O. Gustafsson, “Bidirectional conversion to minimum signed-digit representation,” in *Pro. IEEE Int. Sym. Circuits and Systems*. pp.4, May 2006.