2010 2nd International Conference on Education Technology and Computer (ICETC)

Data Hiding by Exploiting Modification Direction Technique Using Optimal Pixel Grouping

Kai Yung Lin[1], Wien Hong[1], Jeanne Chen[2], Tung Shou Chen[2], Wen Chin Chiang[1]

[1]Department of Information Management
Yu Da University
Miaoli, Taiwan
{linky,wienhong,98104508}@ydu.edu.tw

[2]Dept. of Computer Science and Information Eng.
National Taichung Institute of Technology
Taichung, Taiwan
{jeanne, tschen}@ntit.edu.tw

***Abstract*—Zhang and Wang proposed the Exploiting Modification Direction (EMD) in 2006. The pixels in the image were grouped into *n* pixels per group. A pixel in each group is modified one gray scale value at most to hide a secret digit in a (2*n*+1)-ary notational system. In this paper, we proposed an optimized EMD method by analyzing the relationship between *n* and payload. The study involves using known payload on EMD to ascertain the least distortion result to begin the actual hiding process. Comparison results showed that the PSNR of optimized EMD embedding is significantly higher than OPAP and LSB by at least 3 dBs.**

Keywords- Exploiting Modification Direction (EMD) ; Least Significant Bit (LSB) ; Optimal Pixel Adjustment Process (OPAP)

I. Introduction

Due to the increase traffic on the internet, protecting digital media on the net has become an important issue. Research on data hiding has become important for protecting the media. In data hiding, the secret is embedded in the media that can be retrieved later [1]. The media with the embedded secret must appear to be similar to its original. The media with hidden data is known as the stego-media. However, it is desirable to embed large amount of data while maintaining the least distortion [2]. In digital media, digital image on the internet is the most common and easily available. Therefore, much work has been done to study hiding data on the digital image [3]. Data hiding techniques are mainly reversible [4, 5, 6, 7] or irreversible [8, 9, 10, 11]. The most known reversible technique is the least significant bit (LSB) hiding.

LSB involves embedding the secret in the least few significant bits and payload size is determined by the number of least bits that are available for embedding. However, more bits used in embedding can result in higher image distortion. The LSB has two serious problems; that is embedding more data could result in higher distortion and the hidden data is easily detected.

Chan et al. [10] proposed the optimal pixel adjustment process (OPAP) which effectively reduced distortion. OPAP made use of the last n bits to embed data and at the same time toggles the n+1 bit while comparing the toggle with least distortion. For example, bit stream "0011" after embedding "00" will result in "0000" or "0100". The results in decimal digit of "0011", "0000" and "0100" will be 3, 0 and 4, respectively. It is obvious that changing 3 to 4 will cause less distortion in comparison to 0. Although distortion may be reduced, OPAP is only effective for embedding in two or more bits.

Zhang and Wang [11] proposed the Exploiting Modification Direction (EMD) to reduce distortion. EMD made use of *n* pixels as a group to embed secret digits in a (2*n*+1)-ary notational system. Embedding required adding or subtracting one from a particular pixel within the group. In this paper, the relationship between *n* and payload is analyzed to optimize EMD (Opt EMD) for hiding data. The aim is to hide more data but with reduced distortion.

II. Proposed Method

The EMD technique hide a (2*n*+1)-ary secret digit for every *n* pixels in the cover image by adding or subtracting one at most within one particular pixel in the group. Therefore, it is important to decide the value of *n* before the hiding process. After the *n* is decided, the secret data also need to be transformed into the (2*n*+1)-ary notational system. Equation (1) is used to transform each *n* pixel groups into a reference value f .

$$f(g_1, g_2, ..., g_n) = \left[\sum_{i=1}^{n} (g_i \times i) \right] \bmod (2n+1). \qquad (1)$$

In (1), $(g_1, g_2, ..., g_n)$ refers to the pixel values within each group. After gathering value f , compare f with secret digit *d.* If $d = f$, there is no need for any modification. If $d \neq s$, (2) is needed to find value *s.* The comparison result between *n* and *s* will decide which pixel is going to be modified.

$$s = d - f \bmod (2n+1). \qquad (2)$$

When *s* is less than *n,* add one to the pixel in the position g_s . If *s* is greater than *n,* subtract one to the pixel in

978-1-4244-6370-1/$26.00 © 2010 IEEE

the position g_{2n+1-s}. Follow this routine until the embedding process is completed. To extract the secret data from the stego-image, the pixels are separated according to *n*, and using (1) to find *f*. The *f* extracted from each pixel group is the secret digit in a (2*n*+1)-ary notational system embedded into this pixel group. By this way, the original secret data can be recovered.

From the process above, it is clear that the value *n* could decide the amount of pixel groups and the payload for each group. When *n* is too small, it would take more pixels than is necessary to embed the secret digits. On the other hand, when *n* is too large, there would not be enough space to fully embed the secret digits. This paper proposed a method to calculate *n* which result only in minimum distortion for the cover image. Since EMD technique used *n* pixels to embed a secret digit in a (2*n*+1)-ary digit, (3) could be found from such relationships.

$$\left\lfloor \frac{I_s}{n} \right\rfloor \times \left\lceil \log_2(2n+1) \right\rceil \geq p. \tag{3}$$

In (3), I_s means the total number of pixels within the cover image, *n* means the amount of pixels for each group and *p* is the payload that is going to be embedded. Since p and I_s were known, it is not difficult to calculate maximum *n* that satisfies (3). For example, find the best *n* when embedding 262144 bits secret digit into an 8-bit gray scale image at size 512×512. The result of the equation is as follows after substituting these values into (3) to obtain

$$\left\lfloor 512^2 / n \right\rfloor \times \left\lceil \log_2(2n+1) \right\rceil \geq 262144 .$$

The maximum *n* that satisfying the above inequality can be found to be 4.

III. EXPERIMENTAL RESULTS

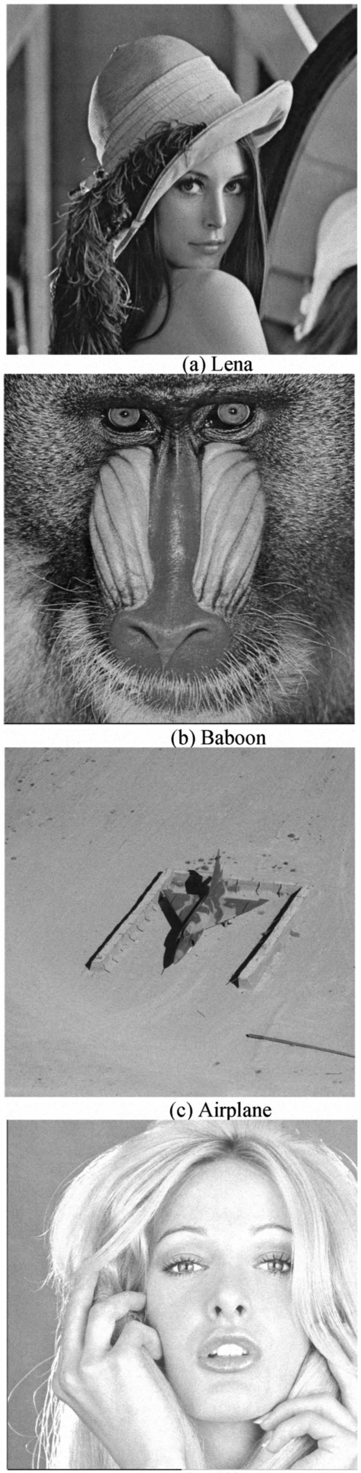
The test images shown in Fig. 1 are Lena, Baboon, Airplane and Tiffany. They are 8-bit gray scale images taken from USC-SIPI Data Base [12] with size 512×512. The secret data used were generated randomly. We will compare the PSNR for our method (Opt EMD) with EMD, OPAP and LSB technique under the same payload. The definition for PSNR as follows

$$\text{PSNR} = 10 \times \log_{10}\left(\frac{255^2}{\text{MSE}}\right). \tag{4}$$

In (4), MSE refers to the mean square error between two pixels. The definition for MSE as follows:

$$\text{MSE} = \frac{1}{m \times n}\sum_{i=1}^{m}\sum_{j=1}^{n}\left(A(i,j) - A'(i,j)\right)^2 . \tag{5}$$

Where *m* and *n* here refer to the width and height for the cover image, and *A*(*i*, *j*) and *A'*(*i*, *j*) refer to the pixel value of cover image and stego image.



(a) Lena

(b) Baboon

(c) Airplane

(d) Tiffany

Figure 1. Test Images (a) ~ (d)

The experimental results are shown in Table I, and Opt EMD refers to the proposed method in this paper. For the EMD technique, we set $n = 2$ as [11].

TABLE I. PSNRS FOR OPT EMD, EMD, OPAP AND LSB UNDER THE SAME PAYLOAD.

bpp	Method	Lena	Baboon	Airplane	Tiffany	AVG
1.5	Opt EMD n=2	52.11	52.10	52.11	52.11	52.11
	EMD n=2	52.11	52.10	52.11	52.11	52.11
	OPAP	47.40	47.41	47.41	47.39	47.40
	LSB	45.91	45.92	45.63	45.94	45.85
1.0	Opt EMD n=4	54.67	54.67	54.66	54.66	54.66
	EMD n=2	53.86	53.87	53.86	53.87	53.87
	OPAP	51.14	51.14	51.16	51.14	51.14
	LSB	51.14	51.14	51.16	51.14	51.14
0.5	Opt EMD n=9	58.37	58.36	58.36	58.38	58.37
	EMD n=2	56.88	56.89	56.88	56.89	56.89
	OPAP	54.16	54.15	54.16	54.15	54.16
	LSB	54.16	54.15	54.16	54.15	54.16

As shown in Table I, it is clear that PSNR for Opt EMD, EMD, OPAP and LSB were not affected by the test images but by payload. Also, OPAP and LSB have the same PSNR when bpp is less or equal to one. This is because when bpp is less or equal to one, both OPAP and LSB will use only one bit to embed the secret data. Therefore, OPAP's second bit modification could not help in reducing the pixel difference. When bpp is equal to 1.5, the PSNR for Opt EMD and EMD is also the same. This is because both methods have the same *n* value, and there is no space to be optimized. This means as payload gets larger there is less space to adjust *n* for reducing the image distortion.

PSNR results showed that Opt EMD has the best performance in keeping low distortion, EMD is the second, OPAP is the third, and the worst is the LSB. The reason for this is because there is less pixel modification needed for Opt EMD and EMD than OPAP and LSB. Besides, Opt EMD and EMD only required modifying the pixel value by one at most where OPAP and LSB are not. The proposed method can find the precise *n* according to the payload which also has the stego-image at the minimum distortion.

IV. CONCLUSIONS

In this paper, we have proposed an optimized EMD which can be manipulated for different payloads to have the most suitable grouping so that information may be embedded with the least distortion. The scheme prevents low PSNR from too many groups and payload smaller than that required of the original secret data. Experimental results showed that the Opt EMD can be used to achieve the most desired stego quality for EMD embedding. The PSNR for the scheme are significantly higher than hiding methods like OPAP and LSB.

REFERENCES

[1] N. Provos and P. Honeyman, “Hide and Seek: An Introduction to Steganography,” IEEE Security and Privacy Magazine, 1(3), 32-44, 2003.

[2] H. Wang and S. Wang, “Cyber Warfare–Steganography vs. Steganalysis,” Communications of the ACM, 47(10), 76-82, 2004.

[3] A. Cheddad, J. Condell, K. Curran and P.M. Kevitt, “Digital Image Steganography: Survey and Analysis of Current Methods,” Signal Processing, 90(3), 727-752, 2010.

[4] J. Tian, “Reversible Data Embedding Using a Difference Expansion,” IEEE Transactions on Circuits and Systems for Video Technology, 13(8), 890-896, 2003.

[5] A.M. Alattar, “Reversible Watermark Using the Difference Expansion of a Generalized Integer Transform,” IEEE Transactions on Image Processing, 13(8), 1147-1156, 2004.

[6] Z. Ni, Y.Q. Shi, N. Ansari and W. Su, “Reversible Data Hiding,” IEEE Transactions on Circuits and Systems for Video Technology, 16(3), 354-362, 2006.

[7] D.M. Thodi and J.J. Rodríguez, “Expansion Embedding Techniques for Reversible Watermarking,” IEEE Transactions on Image Processing, 16(3), 721-730, 2007.

[8] J. Mielikainen, “LSB Matching Revisited,” IEEE Signal Processing Letters, 13(5), 285-287, 2006.

[9] J.M. Guo, “Improved Data Hiding in Halftone Images with Cooperating Pair Toggling Human Visual System,” International Journal of Imaging Systems and Technology, 17(6), 328-332, 2008.

[10] C.K. Chan and L.M. Cheng, “Hiding Data in Images by Simple LSB Substitution,” Pattern Recognition, 37, 469-474, March 2004.

[11] X. Zhang and S. Wang, “Efficient Steganographic Embedding by Exploiting Modification Direction,” IEEE Communications Letters, 10(11), 781–783, 2006.

[12] The USC-SIPI Image Database, http://sipi.usc.edu/database/, Jan. 2009