Information Hiding Scheme with Reversibility Using Difference Segmentation and Histogram Adjustment

Yung-Chen Chou
Dept. of Computer Science and Info. Engineering, Asia University, Taichung 41354, Taiwan
yungchen@gmail.com

Chun-Yi Huang
Dept. of Computer Science and Info. Engineering, Asia University, Taichung 41354, Taiwan
huangjannie@gmail.com

Yu-Chen Hu
Dept. Computer Science and Info. Management, Providence University, Taichung 43301, Taiwan
ychu@pu.edu.tw

Abstract

Secret data delivery is a most important activity in the modern life. Steganography technique uses a cover medium to carry secret data and deliver it to right receivers over public computer networks. Because a stego medium is easier to cheat an unexpected user's observation, Steganography technique is more suitable for delivering secret data. Histogram adjustment is a good way for concealing secret data into a cover image with small distortion. Luo et al. use the pixel interpolation technique and histogram shifting to conceal secret data into a grayscale image. Visual quality and embedding capacity are two most important factors for designing a data hiding technique. The performance of Luo et al.'s method in terms of embedding capacity can be further improved. The proposed method is to increase the height of peak in difference histogram as many as possible. A segmentation strategy is adopted for narrowing down the range of difference between a pixel and a pseudo pixel.

keywords: Data hiding, Histogram adjustment, Secret data delivery, Steganography

1 Introduction

Data hiding technique by using image as the cover medium can be briefly classified into three classes namely spatial domain, frequency domain, and compression domain. Spatial domain is to modify the pixel value directly [3]. The advantages include low computation cost, high embedding rate, and easy implementation. However, spatial domain method has lower visual quality in comparison with frequency domain method. In frequency domain data hiding technique is to transform a cover image into frequency domain and to modify the coefficients for implying the secret data. The transformer includes Discrete Wavelet Transformation (DWT), Discrete Cosine Transformation (DCT), and Discrete Fourier Transformation (DFT). Frequency domain data hiding method has better visual quality of a stego image. Also, it is more robust than spatial domain methods. The robustness means that the secret data can be extracted even a stego image has been processed by general image processing (e.g., brightness, and compression). Compression domain method is to modify a compression procedure for implying secret data in compression code [2][4]. Image compression technique can significantly reduce the size of an image for saving the cost of storing image and sending image over computer networks.

To consider reversibility, data hiding scheme can be divided into irreversible and reversible. The irreversible type data hiding technique cannot reconstruct cover image as original one. Thus, the irreversible data hiding cares more about the embedding capacity than the visual quality of stego image. Contrary, reversible data hiding technique not only can conceal secret data into cover image but also can reconstruct the original image after extracting secret data [1][6] [8][11]. Reversible data hiding technique is useful in military image delivering or distance medical treatment.

Many strategies were adopted in data hiding method design for achieving the reversibility such as frequency transformation, differences extension, and histogram adjustment. Vleeschouwer et al. presented a reversible data hiding technique by using histogram adjustment [11]. Vleeschouwer et al.'s method classifies cover image's pixels into two sets and circularly interpolating two histograms for concealing secret data into cover image. Ni et al. also utilized histogram adjustment concept to design a data hiding technique with reversibility [10]. Ni et al.'s method analyzes the pixels distribution in a cover image to generate the histogram and then to figure out the peak point and zero point. After that the secret data is concealed into cover image by adjusting the histogram. Because Ni et al.'s method tiny adjusts the pixel value for concealing secret data, their method has good visual quality performance. Also, Ni et al.'s method improved the embedding capacity.

Hwang, et al. presented a reversible data hiding technique to achieve the reversibility by extending Ni et al.'s method. Hwang et al.'s method didn't figure out the peak and used zero point and use extra information [5]. Further, Lin and Hsueh utilized bin exchange to modify histogram for increasing pure embedding capacity and get lower distortion [7]. Then, Kim et al. presented a histogram shift based reversible data hiding scheme which selects two peak points and zero points for increasing the embedding capacity [6]. Luo et al. applies pixel prediction strategy to generate the prediction error [9]. Because the possible value of prediction error has significantly reduced (i.e., to compare with pixel value), the peak in the prediction error histogram is significantly increased. Thus, Luo et al.'s method successfully achieved a large embedding capacity.

In this paper, a reversible data hiding technique is presented to achieve higher embedding capacity. The proposed method segments the range of minimum pixel to maximum pixel in a block. After segmentation, each segment will generate a pseudo pixel to play a predicted pixel. Thus, the differences between pseudo pixels and image pixels were generated for generating difference histogram. After that, secret data is embedded into the cover image by adopting the histogram adjustment.

2 The Proposed Method

2.1 The Embedding Phase

The proposed method utilizes the segmentation strategy to divide the range of difference between a maximum and minimum pixels in a block into several segments. For every segment, a pseudo pixel will generated by calculating the central point of the segment. After that, the histogram adjustment is adopted to conceal secret data into cover image. First, divide I into non-overlapping blocks sized $n \times n$ and denoted as $I = \{B_i | i = 1, 2, \ldots, N_B\}$, where N_B represent the number of blocks after division and $B_i = \{x_j | j = 1, 2, \ldots, n \times n\}$. In order to achieve the reversibility, the border blocks will be reserved for concealing the extra data. Here, the extra data include the Left Zero (LZ), Left Peak (LP), Right Peak (RP), Right Zero (RZ), and non-embeddable blocks' information. Further, the secret data S is a data stream encrypted by any crypto system (e.g., RSA, DES, and AES). Let the secret stream be denoted as $S = \{b_k | k = 1, 2, \ldots, N_S\}$, where $b_k \in \{0, 1\}$ and N_S represents the length of secret stream.

For a block B_i, except reserved blocks, calculate the length $D_i = B_{\max} - B_{\min}$ between the maximum pixel $B_{\max} = \max\{x_j \in B_i\}$ and minimum pixel $B_{\min} = \min\{x_j \in B_i\}$ in B_i. Here, N_{seg} represents the number of segments we want to partition. According to the proposed embedding procedure, some blocks that cannot be used to conceal data are called non-embeddable blocks. If a block satisfies one of following case then the block belongs to non-embeddable block.

Case 1: If $D_i < N_{seg}$, then do nothing.
Case 2: $B_{\min} \leq N_{seg}$, then add $B_{\min}$ data to extra information and set $B_{\min}$ as 0.
Case 3: $B_{\max} \geq 255 - N_{seg}$ then add $B_{\max}$ data to extra information and set $B_{\max}$ as 255.
If a block satisfies **Case 1** then the block is a smooth block and all of pixels must be very similar so it will hinder the segment procedure. For preventing underflow and overflow problem, if a

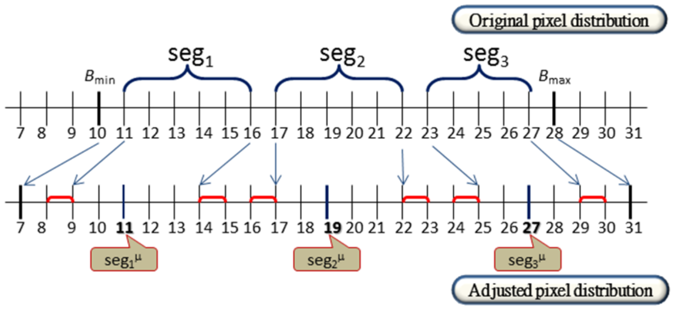


Figure 1. An example for segmentation strategy and pixel adjustment

block satisfies **Case 2** and **Case 3** then it will be not used to conceal secret data, because the smallest/largest pixel in B_i is very close to 0/255. For **Case 2** and **Case 3**, the original pixel values are remembered in the extra information. In order to reduce the size of extra information, the extra information is compressed by using data compression algorithm (e.g., JBig, and Run length coding).

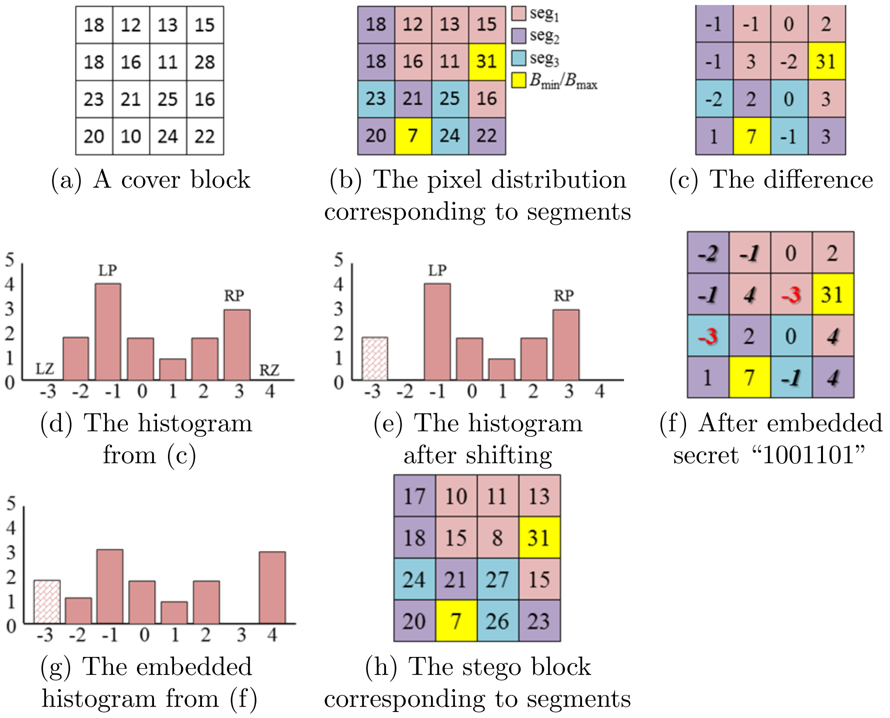
In order to prevent the ambiguous in data extraction phase. For embeddable block B_i, $B_{\min}$ and $B_{\max}$ where modified as $B'_{\min} = B_{\min} - N_{seg}$ and $B'_{\max} = B_{\max} + N_{seg}$. After that, segment the range D_i into N_{seg} segments. The original pixel will also be adjusted for prevent ambiguous problem. The adjusting rule is making a gap (2 pixels) between two neighboring segments. Then, generate the pseudo pixel for each segment. The length of segment is calculated by $\lfloor (B_{\max} - B_{\min})/N_{seg} \rfloor$. The pseudo pixel is calculated by $\lfloor (seg_k^{\max} - seg_k^{\min})/2 \rfloor$, where $seg_k^{\max}$ and $seg_k^{\min}$ represent the maximum and minimum value of k-th segment for a block, respectively. Then, generate the difference value between pixel and its corresponding pseudo pixel. After all of blocks have been checked, the difference histogram can be gained by statistically counting the difference values from whole image. Then, scan the histogram to figure first two highest peak points and let left side peak point as LP and right side peak as RP. Then, scan the histogram to figure out the zero point in LP's left side and set the zero point as LZ. The RZ is found by scanning RP's right side histogram and the found zero point as RZ. After all of LZ, LP, RP, and RZ have been found, then decrease the difference histogram between LZ to LP-1 by one to make LP-1 become zero. Again, increase the histogram between RP+1 to RZ by one to make RP+1 become zero. Thus, the secret can be embedded by checking the pixel difference equivalent to LP or RP. The embedding rule is as Eq. 1. For simple description, let the difference in the block is denoted as d_k and secret bit is b_r.

$$d'_k = \begin{cases} d_k, & \text{if } (d_k = \text{ LP })||(d_k = \text{ RP }) \text{ and } b_r = 0,\\ d_k - 1, & \text{if } (d_k = \text{ LP }) \text{ and } b_r = 1,\\ d_k + 1, & \text{if } (d_k = \text{ RP }) \text{ and } b_r = 1, \end{cases} \tag{1}$$

Embedding Example

Let Fig. 2(a) be a cover image block. The $B_{\min}$ and $B_{\max}$ are 10 and 28, respectively. Due the difference D_i is 18, thus the length of segment is calculated by $6 = \lfloor (28 - 10)/3 \rfloor$. After that, for preventing ambiguous problem, the pixels were adjusted as shown in Fig. 1. After pixel adjustment, $seg_1 = \{9, 10, 11, 12, 13, 14\}$ and 11 is the pseudo pixel for seg_1; $seg_2 = \{17, 18, 19, 20, 21, 22\}$ and 19 is the pseudo pixel for seg_2; $seg_3 = \{25, 26, 27, 28, 29\}$ and 27 is the pseudo pixel for seg_3. Fig. 2(b) is the pixel distribution for the segments. Fig. 2(c) is the difference for every pixel to subtract corresponding pseudo pixel, except $B_{\min}$ and $B_{\max}$. Fig. 2(d) shows the histogram for difference value from Fig. 2(c). Fig. 2(f) shows the difference value after embedding secret "1001101". Finally, the stego image block is shown in Fig. 2(h).

Because reserved blocks are used to remember the extra information (i.e., LP, LZ, RP, RZ, and non-embeddable blocks information), the reversed blocks' LSB will be collected as $LSB_R = \{r_i | i = 1, 2, \ldots, n \times n \times N_R\}$. For restoring the cover image, LSB_R is concatenated with secret



(a) A cover block

(b) The pixel distribution corresponding to segments

(c) The difference

(d) The histogram from (c)

(e) The histogram after shifting

(f) After embedded secret “1001101”

(g) The embedded histogram from (f)

(h) The stego block corresponding to segments

Figure 2. An example for the embedding procedure

data and embedded into cover image. Finally, the extra information is embedded into the reserved blocks by using LSB replacement.

2.2 The Extracting Phase

First, the extra information can be extracted by taking the LSB bits from the reserved blocks. Then, the LP, RP, LZ, and RZ information can be reconstructed. Then, calculate the difference D_i between $B_{\min}$ and $B_{\max}$. If $D_i < 2 \times N_{seg}$, then the block is a non-embeddable block. Also, if $B_{\min}$ = 0 or $B_{\max}$ = 255, the block is non-embeddable, otherwise, the block is embeddable. In **Case 2**, the pixels equal to $B_{\min}$ will be reconstructed by using the information from extra data. In **Case 3**, the pixels equal to $B_{\max}$ will be reconstructed by using the information from extra data. For an embeddable block, to calculate the segment length by $\lfloor D_i/N_{seg} \rfloor$.

After finishing the segmentation procedure, generate the pseudo pixel for every segment and calculate the difference between pixel and pseudo pixel. The secret data extraction can be done by using following rules.

Rule 1: If the difference equal to LP-1 or RP+1 then output secret bit ‘1’.

Rule 2: If the difference equal to LP or RP, then output secret bit ‘0’.

If the difference is located in LP-2 to LZ then increase the difference by one. If the difference is located in RP+2 to RZ, then decrease the difference by one. After that, the pixels will be reconstructed by calculating pseudo pixel plus difference. Finally, the pixel is readjusted to the original.

3 The Experimental Results

The visual quality of stego image and embedding capacity are two most important factors for evaluating a data hiding technique performance. Human vision is a popular measurement for eval-

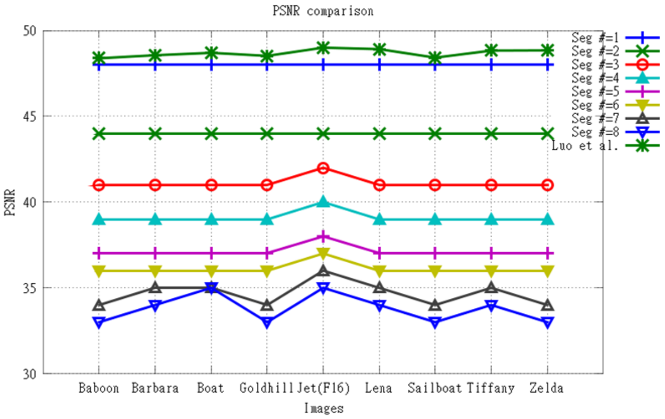


Figure 3. The PSNR comparison for different segment number (block size 4 × 4)

uating the visual quality of stego image. However, human vision is very subjective. Thus, peak signal-to-noise ratio (PSNR) is adopted for measuring the visual quality of stego image. The PSNR is defined in following equations.

$$PSNR = 10 \times \log_{10} \frac{255^2}{MSE} dB \tag{2}$$

$$MSE = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} (p_{i,j} - p'_{i,j})^2. \tag{3}$$

where, H and W represent the image's height and width. $P_{i,j}$ and $p'_{i,j}$ represent original pixel and stego pixel located at (i, j). On the other hand, the embedding capacity is to count the total bits embedded into a cover image. A larger embedding capacity is more useful for delivering secret, because it need not send stego image too many rounds. Fig. 3 shows the comparison of different segment numbers. As we see, the visual quality of stego images were became lower. The reason is when the segment increased then the pixel adjustment extent will become larger. However, all of the visual quality of stego images still greater than 30dB evens the segment number set to 8.

Fig. 4 demonstrates the embedding capacity comparison of testing different segment numbers. From experimental results, the embedding capacity didn't straight increased. We found that a complex content image (e.g., Baboon) will get higher embedding capacity when the segment number increased. In general content image (e.g., Lena), 5 is a suitable segment number. The proposed method has better performance than Luo et al.'s method in terms of embedding capacity when the segment number is greater than 1.

4 Conclusions

Steganography technique is a good way for delivering secret data over the public computer networks because it conceals secret data into a cover medium to cheat unexpected users. On the other hands, according to the sensitive limitation of humans' eyes, image is a good cover medium for secret data delivery. The proposed reversible data hiding scheme utilizes the segmentation strategy for trying to increase the height of peak point in histogram as many as possible. From experimental results, the proposed method significantly improves the performance of Luo et al.'s method in term of embedding capacity.

Acknowledgments.

This work was supported by the Asia University, Taiwan under Grant 100-ASIA-33.

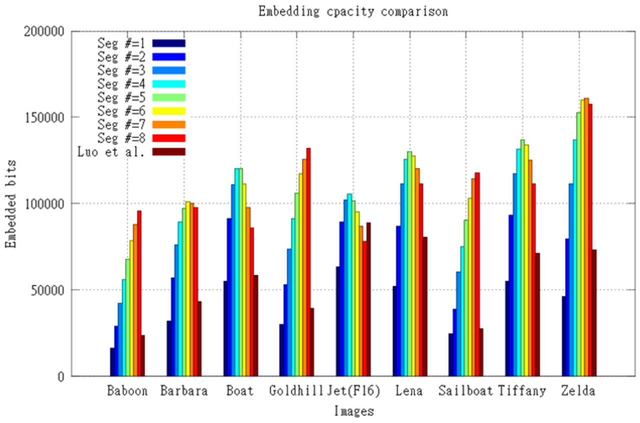


Figure 4. The embedding capacity comparison for different segment numbers (block size 4 × 4

References

[1] J. M. Barton. Method and apparatus for embedding authentication information within digital data, 1994.

[2] C. C. Chang, T. D. Kieu, and Y. C. Chou. Reversible information hiding for vq indices based on locally adaptive coding. *Journal of Visual Communication and Image Representation*, 20(1):57–64, Jan. 2009.

[3] Y. C. Chou, C. C. Chang, and K. M. Li. A large payload data embedding technique for color images. *Fundamenta Informaticae*, 88(1-2):47–61, Sept. 2008.

[4] Y. C. Chou and H. H. Chang. A high payload data hiding scheme for color image based on btc compression technique. In *Proceedings of the Fourth International Conference on Genetic and Evolutionary Computing (ICGEC - 2010)*, pages 626–629, Shenzhen, China, Dec. 2010.

[5] J. Hwang, J. W. Kim, and J. U. Choi. A reversible watermarking based on histogram shifting. In *Proceedings of International Workshop on Digital Watermarking*, volume 4283 of *Lecture Notes in Computer Science*, page 348361, Jeju Island, Korea, 2006. Springer-Verlag.

[6] K. S. Kim, M. J. Lee, H. Y. Lee, and H. K. Lee. Reversible data hiding exploiting spatial correlation between sub-sampled images. *Pattern Recognition*, 42(11):3083–3096, Nov. 2009.

[7] C. C. Lin and N. L. Hsueh. A lossless data hiding scheme based on three-pixel block differences. *Pattern Recognition*, 41(4):14151425, April 2008.

[8] C. Y. Lin and C. C. Chang. Hiding data in vq-compressed images using dissimilar pairs. *Journal of Computers*, 17(2):3–10, Jun. 2008.

[9] L. Luo, Z. Chen, M. Chen, X. Zeng, and Z. Xiong. Reversible image watermarking using interpolation technique. *IEEE Transactions on Information Forensics and Security*, 5(1):187–193, March 2010.

[10] Z. Ni, Y. Q. Shi, N. Ansari, and W. Su. Reversible data hiding. *IEEE Transactions on Circuits System Video Technology*, 16(3):354362, March 2006.

[11] C. D. Vleeschouwer, J. F. Delaigle, and B. Macq. Circular interpretation of bijective transformation in lossless watermarking for media as management. *IEEE Transactions on Multimedia*, 5(1):97105, Mar. 2003.