International Journal of Engineering & Technology, 7 (2.33) (2018) 182-185

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

An effective reversible data hiding algorithm using Huffman coding and pixel value prediction algorithm based on the similarity between adjacent pixel values

Soo-Mok Jung *

Division of Computer Science & Engineering, Sahmyook University, Seoul, 01795, Korea
**Corresponding author E-mail:*

Abstract

In this paper, an effective reversible data hiding algorithm was proposed to increase the amount of data to be embedded data into cover image. Generally, there is similarity between adjacent pixel values in natural images. In the proposed algorithm, a technique that accurately predicts pixel value using adjacent pixel values was used and Huffman coding algorithm was used to increase the amount of data to be embedded into cover image. The proposed reversible data hiding algorithm increases greatly the amount of data embedded into cover image. Also, the visual quality of the secrete data embedded image is very good. So, it is difficult to know whether secrete data has been embedded into the image. Also, using the proposed algorithm, secrete data can be embedded at various levels. The experimental results show that the proposed reversible data hiding algorithm is superior to APD algorithm. Therefore, the proposed algorithm is very useful for reversible data hiding applications.

Keywords: *Cover Image; Data Hiding; Reversible Data Hiding; Stego-Image; Similarity.*

1. Introduction

The technique of hiding secrete data into cover media is called data hiding technique. In the data hiding technique, the quality of the image in which the secrete data is embedded should be excellent so that a person can't recognize whether or not the secrete data is embedded in the cover image. The technique of extracting the original cover image and the secrete data from images with secrete data without loss is called reversible data hiding technique. An image in which secrete data is hidden is called a stego-image.
Many reversible data concealment techniques have been developed in the meantime.(C. C. Chang, et al. 2006- Z. Ni, et al. 2006) Ni etc. proposed a reversible data hiding algorithm called NSAS to increase the amount of embedded bits. (Z. Ni, et al. 2006) The NSAS algorithm hides the data by shifting the histogram of the image. To improve the performance of the NSAS algorithm, Li et al. proposed the APD (Adjacent Pixel Difference) algorithm.(Y. C. Li, 2010) In the APD algorithm, the similarity of adjacent pixels of the image is used. Since neighboring pixel values have a similar value, the difference between adjacent pixels is very small, so that the maximum frequency value increases significantly in the histogram of the difference sequence. Therefore, the APD method using the histogram shifting of the difference sequence greatly increases the number of embedding data bits.
In order to improve the performance of the APD algorithm, we proposed an effective reversible data hiding algorithm which improves the existing method. (S. M. Jung, 2016-S. M. Jung 2016) the proposed algorithm uses an accurate pixel value prediction technique and Huffman coding technique. Using the technique of precisely predicting pixel values increases the maximum frequency in the histogram of the difference sequence. Therefore, the proposed scheme greatly increases the number of data bits to hide compared to the APD algorithm.

2. APD algorithm

In general, there is local similarity in natural images. Thus, the pixel value difference between adjacent pixels is close to zero. Therefore, the frequency around 0 is greatly increased in the histogram of the difference sequence constituted by the difference between adjacent pixels. The APD is an algorithm which greatly increases data to be embedded in a cover image by utilizing the property that values of adjacent pixels are similar. Figure 1 shows the data embedding process and the data extraction process of the APD algorithm.

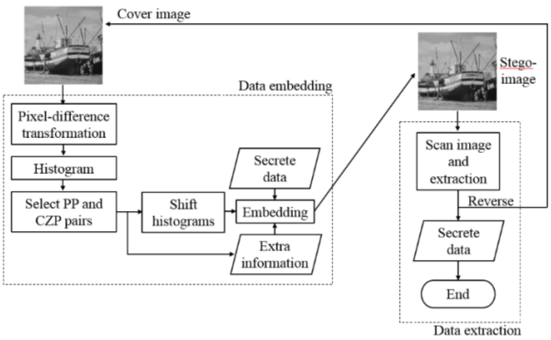


Fig. 1: Flowchart of the APD Method Including the Data Embedding Process and the Data Extraction Process.

The embedding procedure of the APD algorithm is as follows:

1) Generate Sequence C by scanning the pixel values of the cover image in inverse s-order.
2) Generate difference sequence D between adjacent pixels satisfying equation (1).

$$\mathrm{D_i} = \left\{ \begin{array}{c} \mathrm{C_i}\mathit{if}\ \mathrm{i} = 0 \\ \\ \mathrm{C_{i-1}} - \mathrm{C_i}\mathit{elseif}\ 1 \le \mathrm{i} \le \mathrm{n} - 1 \end{array} \right\} \tag{1}$$

3) Generate the histogram of sequence D. Determine PP1 and PP2 with the highest frequency and the second highest frequency. Determine CZP1 and CZP2 which are pixel values with a frequency value of 0 closest to PP1 and PP2.
4) Generate sequence DS as shown in equation (2) using sdj in equation (3).

$$\mathrm{DS_i} = \left\{ \begin{array}{c} \mathrm{D_i}\mathit{if}\ \mathrm{i} = 0\ \mathit{or}\mathrm{D_i} \notin [\mathrm{PP_j} + \mathrm{sd_j}, \mathrm{CZP_j}] \\ \\ \mathrm{D_i} + \mathrm{sd_j}\mathit{if}\mathrm{D_i} \in [\mathrm{PP_j} + \mathrm{sd_j}, \mathrm{CZP_j}] \end{array} \right\} \tag{2}$$

$$\mathrm{sd_j} = \left\{ \begin{array}{c} 1\ \mathit{if}\mathrm{PP_j} < \mathrm{CZP_j} \\ \\ -1\ \mathit{elseif}\mathrm{CZP_j} < \mathrm{PP_j} \end{array} \right\} \text{Where j} \in \{1,2\} \tag{3}$$

5) Generate embedded sequence DE by embedding secrete data as shown in equation (4).

$$\mathrm{DE_i} = \left\{ \begin{array}{c} \mathrm{DS_i}\mathit{if}\ \mathrm{i} = 0\ \mathit{or}\mathrm{DS_i} \ne \mathrm{PP_j}\mathit{or}\ \text{data} = 0 \\ \\ \mathrm{DS_i} + \mathrm{sd_j}\mathit{if}\mathrm{DS_i} = \mathrm{PP_j}\mathit{or}\ \text{data} = 1 \end{array} \right\} \tag{4}$$

6) Generate the stego-image sequence S according to equation (5).

$$\mathrm{S_i} = \left\{ \begin{array}{c} \mathrm{DE_i}\mathit{if}\ \mathrm{i} = 0 \\ \\ \mathrm{C_{i-1}} - \mathrm{DE_i}\mathit{if}\ 1 \le \mathrm{i} \le \mathrm{n} - 1 \end{array} \right\} \tag{5}$$

Where n=(image height)*(image width)

7) Generate stego-image by arranging S in reverse s-order.

The above embedding procedure is the same as the parenthesized data embedding in Figure 1. The visual quality of the generated stego-image is excellent, so that it can't be recognized whether secrete data is inserted into the stego-image. Stego-image contains both cover images and secrete data information. Thus, in the data extraction procedure, the original cover image can be completely restored from the stego-image and the secrete data can be extracted completely from the stego-image. APD is a very good technique to hide large amounts of secrete data into cover image using histogram shift. The extraction procedure of APD algorithm is as follows:

1) Generate sequence S by scanning the stego-image in inverse s-order.
2) Restore C and DE by repeating steps 2.1 and 2.2 with increasing i value.
a) Restore the sequence DE according to equation (6).

$$\mathrm{DE_i} = \left\{ \begin{array}{c} \mathrm{C_i}\mathit{if}\ \mathrm{i} = 0 \\ \\ \mathrm{C_{i-1}} - \mathrm{S_i}\mathit{else} \end{array} \right\} \tag{6}$$

b) Restore sequence C using equation (7).

$$\mathrm{C_i} = \left\{ \begin{array}{c} \mathrm{S_i} + \mathrm{sd_j}\mathit{if}\ 1 \le \mathrm{i} \le \mathrm{n} - 1\ \mathit{and}\mathrm{C_{i-1}} - \mathrm{S_i} \in [\mathrm{PP_j} + \mathrm{sd_j}, \mathrm{CZP_j}] \\ \\ \mathrm{S_i}\mathit{eslse} \end{array} \right\} \tag{7}$$

3) Extract secrete data embedded in stego-image by applying equation (8) to Dei

$$\text{Extraction bit} = \left\{ \begin{array}{c} 0\ \mathit{if}\mathrm{DE_i} = \mathrm{PP_j} \\ \\ 1\ \mathit{elseif}\mathrm{DE_i} = \mathrm{PP_j} + \mathrm{sd_j} \end{array} \right\} \tag{8}$$

4) Generate original cover image by arranging s in inverse s-order.

The above extraction procedure is the same as the parenthesized data extraction in Figure 1. The restored cover image is exactly the same as the original cover image, and the secrete data is completely restored without loss. Although the APD algorithm is a highly effective reversible data hiding technique, the maximum number of data bits to be embedded into cover image is limited to h(PP1) + h(PP2).

3. Proposed algorithm

The flowchart of the proposed algorithm is shown in figure 2 and pixel value prediction is performed using the adjacent pixels shown in Fig. 3. The predicted pixel values constitute the prediction image. Pixel value prediction is performed in reverse s-order with respect to the cover image as shown in Fig. 3 The reverse s-order means the left-right reversed s-order. Therefore, two rows on the top side, 2 columns on the right side and left side of the prediction image are the same as the cover image. The pixel value prediction using the adjacent pixel values is performed as shown in Equation (9)-(12). Adjacent pixels used in Equation (9) are pixel values already scanned in reverse s-order. The weight of influence of the pixel adjacent to the position at which the predicted pixel value is obtained on the predicted pixel value is set to 1. And the influence weight of the other nearby pixels on the predicted pixel value is .

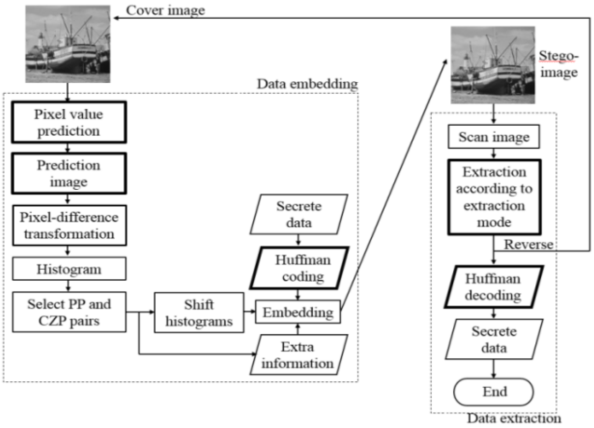


Fig. 2: Flowchart of the Proposed Algorithm Including the Data Embedding Process and the Data Extraction Process.

	A3	A2	A1	
A4	A5	A6	A7	A8
		Val	A10	A9

Fig. 3: Pixel Value Prediction.

As shown in equation (11), if the SAVD (Sum of the Absolute Values of the Differences) calculated using the V obtained in equation (9) is larger than, it is determined that the locality is low and the pixel value of the cover image is used as the predicted pixel value. Otherwise, it is judged the locality is high, and the V is used as the predicted pixel value as shown in equation (11), (12). If is zero, pixel value prediction can’t be executed. So, the proposed algorithm is equal to APD algorithm

V= (A5+A6+A7+A10+ (A1+A2+A3+A4+A8 +A9)* ・)/ (4.0+6* ・) (9)

SAVD=abs(V-A5)+abs(V-A6)+abs(V-A7)+abs(V-A10)+{abs(V-A1)+abs(V-A2)+abs(V- A3)+ abs(V-A4)+abs(V-A8)+abs(V-A9)}* (10)

If (SAVD>=・) Val= pixel value at the position of the cover image (11)

else Val=V (12)

Difference sequence D is generated with C and P as shown equation (13). C, P are sequences generated by scanning the cover image and the predicted image in the reverse S-order, respectively. The histogram of sequence D is generated and the following values are obtained. The largest Peak Point 1(PP1,), the Closest Zero Point 1(CZP1), the largest Peak Point 2(PP2), the Closest Zero Point 2(CZP2) can be determined in the histogram of D. Difference sequence D is shifted as APD algorithm using the PP1, PP2, CZP1, CZP2. The shifted D is denoted by DS. The sdj in equation (14) can be calculated as APD algorithm.

To increase the amount of embedding data bits, Huffman code is applied to the original secrete data. Huffman code table which is the information for restoring Huffman-coded secrete data is embedded into the front part of cover image and the Huffman coded secrete data are embedded in the next part of cover image. As shown in equation (15), Huffman table and Huffman coded secrete data are embedded into DS, the result sequence is called DE. As shown in equation (16), stego-image sequence is generated using DE and C. A Stego-image, in which secrete data is embedded cover image, is generated by constructing S in reverse s-order.

Di=Ci when i=0, Ci-Pi when prediction value was used, Ci-1-Ci otherwise (13)

DSi=Di when i=0 or Di ∉[PPj+sdj, CZPj], Di+sdj when Di ∉[PPj+sdj, CZPj] (14)

DEi=DSi when i=0 or DSi≠PPj or data=0, DSi+sdj when DSi=PPj and data=1 (15)

Si=DEi when i=0, Ci-1-DEi when 1≤i≤n-1 (16)

Huffman code table, Huffman coded secrete data, and original cover image can be extracted from the stego-image. The extraction process is as follows:

Stego-image sequence S is constructed by scanning stego-image in inverse s-order. Stego-image is a cover image containing a Huffman code table, Huffman coded secrete data. Extraction mode is determined at each position in sequence S from i=0 to (image width)*(image height)-1 during the reverse process of the embedding process.

Mode 1: In the case of predicted pixel value was not used at the current position and predicted pixel value was not used at just before position.

Mode 2: In the case of predicted pixel value was used at the current position.

Mode 3: In the case of predicted pixel value was not used at the current position.

And predicted pixel value was used at just before position.

DE which is the difference sequence with Huffman table and Huffman coded secrete data is extracted as APD algorithm. At mode 1, Ci is extracted in the same way as the APD algorithm.

At Mode 2, Ci is extracted as shown in equation (17). At the Mode 3, Ci is extracted as shown in equation (19). Huffman code table and Huffman coded secrete data are extracted in the same manner as the APD algorithm as shown in Equation (20).The original secrete data can be restored easily using the extracted Huffman code table. In equation (17), RPi is the prediction pixel value at the restoration process at ithposition, RFi is the related factor at ith position to restore Ci, and they are calculated as shown in equation (18). Following these steps, the cover image and confidential data are extracted without loss. Therefore, the extracted cover image and secrete data are the same as the original cover image and the original secrete data.

Ci= RPi+RFi (17)

RFi=Ci-1-Si-sdj when Ci-1-Si∈[PPi+sdj, CZPj], Ci-1-Si in other cases (18)

Ci=RPi-RFi (19)

Extracted bit=0 when DEi=PPj , 1 when DEi=PPj+sdj (20)

4. Experimental results

The performance of the proposed algorithm was evaluated using a 512x512 pixels aerial, ship, girl, Reese images as cover images. In the experiment, □was 0.29 and □was 5, 10, 15. The abstract of this paper was converted into ASCII code and used as secrete data and embedded in the cover images. The cover images used in the experiment and the stego-images are shown in figure 4. As shown in Table 1, the PSNR (Peak Signal to Noise Ratio) value of the stego-image is reduced, but the visual quality of the stego-image is excellent as shown in Figure 4. Therefore, it is difficult to visually distinguish the stego-image and the cover image. As shown in Figure 4, the visual quality of the stego-images is very good so that a person can’t recognize whether or not the secrete data is embedded in the cover image.

The values which are measured in experiment such as the number of embedded characters, the image quality of the stego-image, prediction accuracy of the pixel value, average prediction error, and the rate of increase in the number of embedded characters are shown in Table 1. As shown in Table 1, it can be seen that the average of pixel value prediction errors is very small. Therefore, the maximum frequency is increased in the histogram of the difference sequence, and the number of bits to be embedded increases. The maximum increase rate of embedded bits of the proposed scheme is 122.4%.

(a1) aerial

(a2) APD

(a3) Pro (5)

(a4) Pro (10)

(a5)Pro (15)

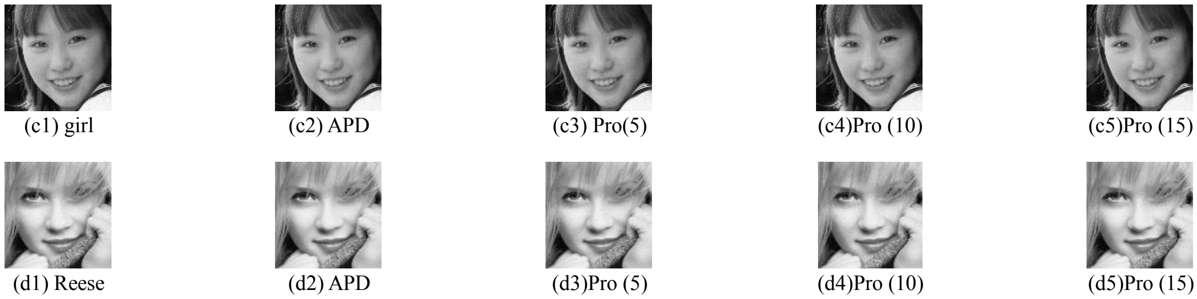
(b1) ship

(b2) APD

(b3)Pro(5)

(b4)Pro (10)

(b5)Pro (15)


(c1) girl (c2) APD (c3) Pro(5) (c4)Pro (10) (c5)Pro (15)

(d1) Reese (d2) APD (d3)Pro (5) (d4)Pro (10) (d5)Pro (15)

Fig 4: Cover Images & Stego-Images.

Table 1: Experimental Results

	Algorithm	Embedded characters	PSNR (dB)	Prediction accuracy	Average prediction error	Increase rate of embedded characters
Aerial	APD	5,501	48.56	-	-	-
	Pro(5)	10,125	44.48	31.29%	1.20	84.1%
	Pro(10)	10,234	39.04	23.49%	1.72	86.0%
	Pro(15)	10,274	35.63	19.79%	2.20	86.8%
Ship	APD	5,803	48.58	-	-	-
	Pro(5)	11,352	43.00	21.32%	1.50	95.6%
	Pro(10)	12,674	37.28	19.89%	1.67	118.4%
	Pro(15)	12,904	35.00	18.40%	1.92	122.4%
Girl	APD	7,596	48.73	-	-	-
	Pro(5)	14,047	43.85	33.24%	1.16	84.9%
	Pro(10)	14,579	37.86	23.02%	1.60	91.9%
	Pro(15)	15,259	35.316	20.29%	1.83	100.9%
Reese	APD	11,665	49.01	-	-	-
	Pro(5)	21,295	43.49	39.13%	0.83	82.6%
	Pro(10)	22,035	39.58	32.62%	1.10	88.9%
	Pro(15)	22,285	37.32	29.27%	1.32	91.0%

5. Conclusions

In this paper, an effective reversible data hiding algorithm which use pixel value prediction based on locality and Huffman coding was proposed. In order to improve the performance of the APD algorithm, an accurate pixel value prediction technique and Huffman coding technique were used in the proposed algorithm. To evaluate the performance of the proposed algorithm, cover images with 512x512 pixels such as aerial, ship, girl, Reese were used in the experiment. . In the experiment, □was 0.29 and □was 5, 10, 15. The abstract of this paper was converted into ASCII code and used as secrete data and embedded in the cover images.

Applying the proposed algorithm, the restored cover image is exactly the same as the original cover image, and the secrete data is extracted without loss. The PSNR (Peak Signal to Noise Ratio) value of the stego-image is reduced, but the visual quality of the stego-image is excellent so that a person can’t recognize whether or not the secrete data is embedded in the cover image, and the maximum increase rate of embedded bits of the proposed scheme is 122.4% compared with the APD algorithm. The proposed algorithm is very good reversible data hiding algorithm and it is very useful for medical, military, and digital library applications.

References

[1] C. C. Chang, W. L. Tai, and C. C. Lin, 2006, A reversible data hiding scheme based on side match vector quantization, IEEE Trans. on Circuits and Systems for Video Technology, 16(10): 1301-1308.

[2] C. C. Lo, Y. C. Hu, W. L. Chen, and C. M. Wu, 2014, Reversible Data Hiding Scheme for BTC-compressed Images based on Histogram Shifting,International Journal of Security and Its Applications, 8(2): 301-314.

[3] H. C. Huang, C. M. Chu, and J. S. Pan, 2009, the optimized copyright protection system with genetic watermarking, Soft Computing, 13(4): 333-343.

[4] H. W. Tseng and C. P. Hsieh, 2008, Reversible data hiding based on image histogram modification, Imaging Science Journal, 56(5): 271-278.

[5] L. Kamstra, H.J.A.M. Heijmans, 2005, Reversible data embedding into images using wavelet techniques and sorting, IEEE Trans. on Image Process, 14(12): 2082-2090.

[6] S. M. Jung, Dec., 2016, An efficient data embedding algorithm with reversibility by using pixel value prediction and Huffman coding, International Conference on Computing Convergence and Applications, Dec., 61-64.

[7] S. M. Jung, Aug., 2016, Reversible data hiding algorithm using spatial locality and the surface characteristics of image, Journal of the Korea Society of Computer and Information, 21(8): 1-12.

[8] X. Li, B. Li, B. Yang, and T. Zeng, 2013, General Framework to Histogram shifting based Reversible Data Hiding, IEEE Trans. on Image Processing, 22(6): 2181-2191.

[9] Y. C. Li, C. M. Yeh, and C. C. Chang, 2010, Data hiding based on the similarity between neighboring pixels with reversibility, Digital Signal Processing, 20(4): 1116-1128.

[10] Z. Ni, Y. Q. Shi, N. Ansari, and W. Su, 2006, Reversible data hiding, IEEE Trans. on Circuits and Systems for Video Technology, 16(3): 354-362.