

# An Image Steganographic Method Based on Block-wise Pixel-Value Differencing

Hsiao-Shan Huang\*, Chin-Song Wu, Ming-Hsin Chang, and Gan-How Chang  
Telecommunication Laboratories, Chunghwa Telecom Co., Ltd.

**Abstract**—In order to increase the hiding capacity, we proposed a partitioning scheme based on pixel-value differencing (PVD) to embed data in gray scale images. The decision of the partitioning result is decided by the one with more edge-like modes. The main idea of the proposed method is to find more edge areas so as to hide more secret data by reason of the characteristics of human vision resolution which have a larger tolerance in edge areas than in smooth areas. In addition, a further altering process is proposed to ensure the hiding data can be correctly extracted. From the experimental results, all of the stego-images keep high visual quality and have large hiding capacity.

## I. INTRODUCTION

In recent years, security topics have become essential issues in the development of digital transmission of information through the Internet. There are many ways to prevent secret messages from being captured during transmission. Image steganographic techniques hide secret messages in images by making many visually imperceptible changes to encode the secret data. The secret data are hidden in images so they cannot be scanned, attacked, rejected or even deleted entirely in transmission.

Several steganographic methods have been proposed. Least-significant-bit (LSB) [1] is the most general one. It replaces length-fixed least bits of pixels to hide secret messages. In 2003, Wu and Tsai proposed a steganographic method by pixel-value differencing (PVD) [2]. The two continuous pixels are grouped together and the difference value is computed in order to embed the secret data. Based on LSB and PVD method, some enhanced steganographic schemes are presented [3], [5], [6], [7].

In 2007, Wang, C.-M, et al. proposed a high quality steganographic method [4]. By analyzing the characteristic of Wu and Tsai's method, they achieve the same hiding capacity as PVD method while the quality of image is higher. In the proposed method, we utilize the difference values of adjacent four pixels to provide better hiding capacity.

This paper is organized as follows. In Section II, we will review Wang, C.-M, et al.'s method. The proposed procedure for embedding and extracting algorithm will be described in Section III. The experimental results and analysis are illustrated in Section IV. Finally, the conclusion is presented in Section V.

## II. REVIEW OF WANG, C.-M, ET AL.'S METHOD [4]

In Wang, C.-M, et al.'s method, a cover image  $F$  sized

$M \times N$  is partitioned into  $k$  blocks where  $k = M \times N / 2$ . Each block  $F_i$  contains two continuous pixels  $P_{(x,y)}$  and  $P_{(x,y+1)}$  where  $i=1,2,\dots,k$ . The difference value  $d_i$  is obtained by subtracting  $P_{(x,y)}$  from  $P_{(x,y+1)}$  which can be derived by

$$d_i = |P_{(x,y)} - P_{(x,y+1)}|. \quad (1)$$

Then a range table  $T$  is designed to classify  $d_i$ .  $T$  is composed of  $n$  contiguous sub-ranges  $T_j$  where  $j=1,2,\dots,n$ . The lower and upper bound values of each sub-range  $T_j$  are defined as  $l_j$  and  $u_j$ , respectively. The width  $w_j$  of  $T_j$  is designated to be a power of 2, so it can be indicated that  $w_j = u_j - l_j + 1$ . The hiding capacity  $t_i$  of each block  $F_i$  depends on the difference value  $d_i$  which can be calculated by

$$t_i = \lfloor \log_2(w_j) \rfloor. \quad (2)$$

The remainder values of  $P_{(x,y)}$ ,  $P_{(x,y+1)}$  and  $F_i$  denoted as  $R_{(x,y)}$ ,  $R_{(x,y+1)}$  and  $R_i$  are clarified by

$$\begin{aligned} R_{(x,y)} &= P_{(x,y)} \bmod t^i \\ R_{(x,y+1)} &= P_{(x,y+1)} \bmod t^i \\ R_i &= (P_{(x,y)} + P_{(x,y+1)}) \bmod t^i \end{aligned} \quad (3)$$

where  $t^i$  is the decimal value of  $t_i$ .

Generate new  $P'_{(x,y)}$  and  $P'_{(x,y+1)}$  to embed  $t_i$  bits of secret data as follows.

If  $R_i > t^i$  and  $m = |R_i - t^i| \leq 2^{t_i-1}$ , the modified factor is defined as  $m_c = m$ . Otherwise, it is changed to  $m_c = 2^{t_i} - |R_i - t^i|$  when  $R_i \leq t^i$  and  $m > 2^{t_i-1}$  by

$$\begin{aligned} &(P'_{(x,y)}, P'_{(x,y+1)}) \\ &= \begin{cases} (P_{(x,y)} - \lfloor m_c/2 \rfloor, P_{(x,y+1)} - \lfloor m_c/2 \rfloor), & \text{if } P_{(x,y)} \geq P_{(x,y+1)} \\ (P_{(x,y)} + \lfloor m_c/2 \rfloor, P_{(x,y+1)} + \lfloor m_c/2 \rfloor), & \text{if } P_{(x,y)} < P_{(x,y+1)} \end{cases}. \end{aligned} \quad (4)$$

In the same way, if  $R_i \leq t^i$  and  $m \leq 2^{t_i-1}$ , it remains  $m_c = m$ . Otherwise, it is altered to  $m_c = 2^{t_i} - |R_i - t^i|$  when  $R_i > t^i$  and  $m > 2^{t_i-1}$  by

$$\begin{aligned} &(P'_{(x,y)}, P'_{(x,y+1)}) \\ &= \begin{cases} (P_{(x,y)} + \lfloor m_c/2 \rfloor, P_{(x,y+1)} + \lfloor m_c/2 \rfloor), & \text{if } P_{(x,y)} \geq P_{(x,y+1)} \\ (P_{(x,y)} - \lfloor m_c/2 \rfloor, P_{(x,y+1)} - \lfloor m_c/2 \rfloor), & \text{if } P_{(x,y)} < P_{(x,y+1)} \end{cases}. \end{aligned} \quad (5)$$

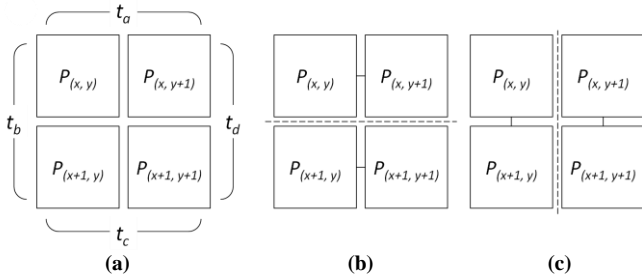
## III. THE PROPOSED METHOD

Instead of containing two continuous pixels, the proposed method groups the adjacent four pixels  $P_{(x,y)}$ ,  $P_{(x,y+1)}$ ,  $P_{(x+1,y)}$  and  $P_{(x+1,y+1)}$  in each block  $F_i$ . By using the difference values between these four pixels to provide better hiding capacity.

The proposed embedding and extracting methods are described in the following subsections.

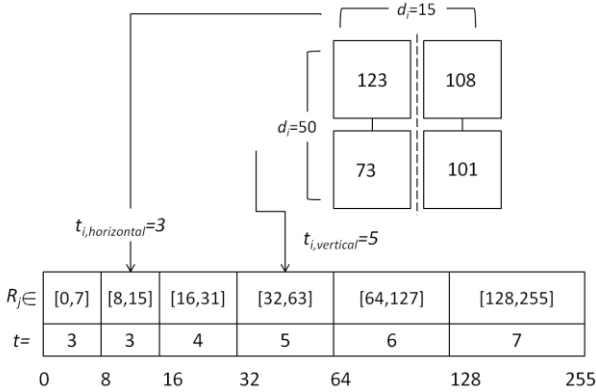
*A. The embedding algorithm*

As mentioned above, each block  $F_i$  contains four pixels. This four-pixel block will be partitioned into two two-pixel sub-blocks. There are two modes of partitioning result. The result depends on comparing the difference values between the two modes, as shown in Fig. 1. The proposed method dedicates to find more edge areas in order to hide more secret data by reason of the characteristics of human vision resolution which have a larger tolerance in edge areas than in smooth areas. Therefore, the values of hiding capacity  $t_i$  between  $(P_{(x,y)}, P_{(x,y+1)})$  and  $(P_{(x,y)}, P_{(x+1,y)})$  are compared. The one with larger  $t_i$  will be chosen as the partitioning result.



**Fig. 1 (a) The adjacent four pixels  $P_{(x,y)}$ ,  $P_{(x,y+1)}$ ,  $P_{(x+1,y)}$  and  $P_{(x+1,y+1)}$  are grouped together in each block  $F_i$ . There are two modes of partitioning result (b) horizontal mode (c) vertical mode.**

An illustration of the choosing procedure is shown in Fig. 2. Assume the range table  $T_0 \in [0,7]$ ,  $T_1 \in [8,15]$ ,  $T_2 \in [16,31]$ ,  $T_3 \in [32,63]$ ,  $T_4 \in [64,127]$ , and  $T_5 \in [128,255]$ . In the figure, the gray values of a block  $F_i$  are supposed to be  $(P_{(x,y)}, P_{(x,y+1)}, P_{(x+1,y)}, P_{(x+1,y+1)}) = (123, 108, 73, 101)$ . The difference values of *horizontal* and *vertical* mode are 15 and 50, respectively. By (2), the hiding capacity  $t_{i,vertical}=5$  is larger than  $t_{i,horizontal}=3$ . Therefore, *vertical* mode is chosen as the partitioning result. In other words, the case shown that *vertical* mode has more edge-like areas than *horizontal* one.



**Fig. 2 An illustration of the choosing procedure**

The next step is to embed secret data according to the mode chosen above. If it is *vertical* mode, the two adjacent pixels  $(P_{(x,y)}, P_{(x+1,y)})$  and  $(P_{(x,y+1)}, P_{(x+1,y+1)})$  are grouped together respectively. Otherwise, the *horizontal* mode is performed to organize  $(P_{(x,y)}, P_{(x,y+1)})$  and  $(P_{(x+1,y)}, P_{(x+1,y+1)})$

into two groups. Then the difference value  $d_i$  is acquired by subtracting each other in the two-pixel group by (1). And then compute the hiding capacity  $t_i$  by (2). Following (3), the remainder values  $R_{(x,y)}$ ,  $R_{(x,y+1)}$  and  $R_i$  can be obtained. The final procedure is calculating the new pixel values according to (4) and (5).

After embedding the secret data, calculate the new difference value  $d_i$  of *horizontal* and *vertical* mode by.

$$d_i' = |P'_{(x1,y1)} - P'_{(x2,y2)}|, \tag{6}$$

where  $P'_{(x1,y1)}$  and  $P'_{(x2,y2)}$  denote the new adjacent two pixels in a sub-block. If the comparison of the mode decision is different from the original solution, do the following step to fit the original condition.

Define the hiding capacity between  $P_{(x,y)}$ ,  $P_{(x,y+1)}$ ,  $P_{(x+1,y)}$  and  $P_{(x+1,y+1)}$  are  $t_a$ ,  $t_b$ ,  $t_c$  and  $t_d$  respectively as shown in Fig. 1(a). The optimal approach to further alter the four pixels in block  $F_i$  to achieve the original condition is as follows:

**Horizontal mode:** if the original hiding capacity  $t_a \geq t_b$  and the new hiding capacity  $t'_a < t'_b$ , re-adjust the new pixels by following rules:

**Case 1:**  $t'_c > t'_a$  and  $P'_{(x,y)} > P'_{(x+1,y)}$

$$(P''_{(x,y)}, P''_{(x,y+1)}) = (P'_{(x,y)} - t_d/2, P'_{(x,y+1)} - t_d/2)$$

**Case 2:**  $t'_c > t'_a$  and  $P'_{(x,y)} \leq P'_{(x+1,y)}$

$$(P''_{(x,y)}, P''_{(x,y+1)}) = (P'_{(x,y)} + t_d/2, P'_{(x,y+1)} + t_d/2)$$

**Case 3:**  $t'_c \leq t'_a$  and  $P'_{(x,y)} > P'_{(x+1,y)}$

$$(P''_{(x+1,y)}, P''_{(x+1,y+1)}) = (P'_{(x+1,y)} + t_c/2, P'_{(x+1,y+1)} + t_c/2)$$

**Case 4:**  $t'_c \leq t'_a$  and  $P'_{(x,y)} \leq P'_{(x+1,y)}$

$$(P''_{(x+1,y)}, P''_{(x+1,y+1)}) = (P'_{(x+1,y)} - t_c/2, P'_{(x+1,y+1)} - t_c/2)$$

**Vertical mode:** if the original hiding capacity  $t_a < t_b$  and the new hiding capacity  $t'_a \geq t'_b$ , the new pixels are re-arranged as below:

**Case 1:**  $t'_d > t'_b$  and  $P'_{(x,y)} > P'_{(x+1,y)}$

$$(P''_{(x,y)}, P''_{(x+1,y)}) = (P'_{(x,y)} - t_b/2, P'_{(x+1,y)} - t_b/2)$$

**Case 2:**  $t'_d > t'_b$  and  $P'_{(x,y)} \leq P'_{(x+1,y)}$

$$(P''_{(x,y)}, P''_{(x+1,y)}) = (P'_{(x,y)} + t_b/2, P'_{(x+1,y)} + t_b/2)$$

**Case 3:**  $t'_d \leq t'_b$  and  $P'_{(x,y)} > P'_{(x+1,y)}$

$$(P''_{(x,y+1)}, P''_{(x+1,y+1)}) = (P'_{(x,y+1)} + t_d/2, P'_{(x+1,y+1)} + t_d/2)$$

**Case 4:**  $t'_d \leq t'_b$  and  $P'_{(x,y)} \leq P'_{(x+1,y)}$

$$(P''_{(x,y+1)}, P''_{(x+1,y+1)}) = (P'_{(x,y+1)} - t_d/2, P'_{(x+1,y+1)} - t_d/2)$$

Following the rules above, each block  $F_i$  can correctly extract the embedded data since the modified partitioning result is the same as the original one.

An example of further altering process is shown in Fig. 3. Suppose the original pixel of a block is  $(P_{(x,y)}, P_{(x,y+1)}, P_{(x+1,y)}, P_{(x+1,y+1)}) = (50, 37, 73, 79)$ , the partitioning result is *vertical* mode. By the embedding procedure, the first and second sub-blocks embedded  $0100_{(2)}$  and  $00111_{(2)}$ , respectively. However, it becomes *horizontal* mode when the new pixels  $(47, 31, 69, 72)$  are recalculated. Hence, the further altering process is performed to improve the case.

Compare the new hiding capacity  $t'_d=5$  and  $t'_b=4$ , and new pixel value  $P'_{(x,y)} = 47$  and  $P'_{(x,y+1)} = 31$ , it fit in with **Case 1** of *vertical* mode mentioned above. Therefore, the new pixels become  $(P''_{(x,y)}, P''_{(x+1,y)}) = (39, 61)$  after the further altering

process. Fig. 4 shows the flowchart of the proposed embedding method.

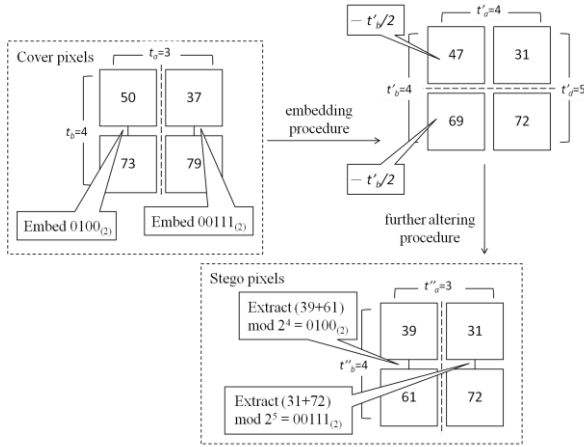


Fig. 3 An example of further altering process

### B. The extracting algorithm

The following steps are performed to extract the embedded data. First, given a block  $F_i$  contains adjacent four pixels  $P_{(x,y)}$ ,  $P_{(x,y+1)}$ ,  $P_{(x+1,y)}$  and  $P_{(x+1,y+1)}$ . According to the range table  $T$ , the difference values  $d_i$  and the hiding capacity  $t_i$  of the *horizontal* and *vertical mode* can be derived by (1) and (2). Following the rules that the larger  $t_i$  is chosen as the partitioning result, the *mode* can be correctly decided. Then, the secret data can be extracted by computing the remainder value  $R_i$  of the two sub-blocks by (3), and the length of secret data is decided by  $t_i$ . After that, the extracting procedure is accomplished.

A simple example is given below. Assume a block contains four pixel values  $(P_{(x,y)}, P_{(x,y+1)}, P_{(x+1,y)}, P_{(x+1,y+1)}) = (146, 160, 187, 156)$ , the hiding capacity  $t_{i,vertical}=5$  is larger than  $t_{i,horizontal}=3$ . Hence, the *vertical mode* is chosen as the partitioning result. In other words, the two adjacent pixels  $(146, 187)$  and  $(160, 156)$  are grouped together, respectively. By Wang, C.-M, et al.'s method, the remainder value of first sub-block  $(146, 187)$  can be obtained by  $(146+187) \bmod 2^5 = 13_{(10)} = 1101_{(2)}$ . Base on the embedding procedure, the length of the embedded bits is the same as the hiding capacity. So the hiding data in first sub-block is  $01101_{(2)}$ . Similarly, the hiding data of the second sub-block is  $100_{(2)}$ .

## IV. EXPERIMENTAL RESULTS AND ANALYSIS

In this section, the experimental results are demonstrated that the proposed method can perform better hiding capacity than Wang, C.-M, et al.'s method. The experiment conditions are described below. C programming language is used to implement the proposed and Wang, C.-M, et al.'s method. Five cover images sized  $512 \times 512$  were used as the test images in the experiments. And the range table is fixed to  $T_0 \in [0,7]$ ,  $T_1 \in [8,15]$ ,  $T_2 \in [16,31]$ ,  $T_3 \in [32,63]$ ,  $T_4 \in [64,127]$ , and  $T_5 \in [128,255]$ . The hiding capacity (bytes) and the peak signal-to-noise ratio (PSNR) were adopted to evaluate the performance of the proposed method. The

PSNR is defined as below:

$$PSNR = 10 \times \log_{10} (255^2 / MSE) \text{ dB} \quad (7)$$

The parameter  $MSE$  is the mean square error of the stego-image and the cover image. It is defined as follows:

$$MSE = \frac{1}{M \times N} \left( \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} (P_{xy} - P'_{xy})^2 \right) \quad (8)$$

where  $M \times N$  is the size of the image.

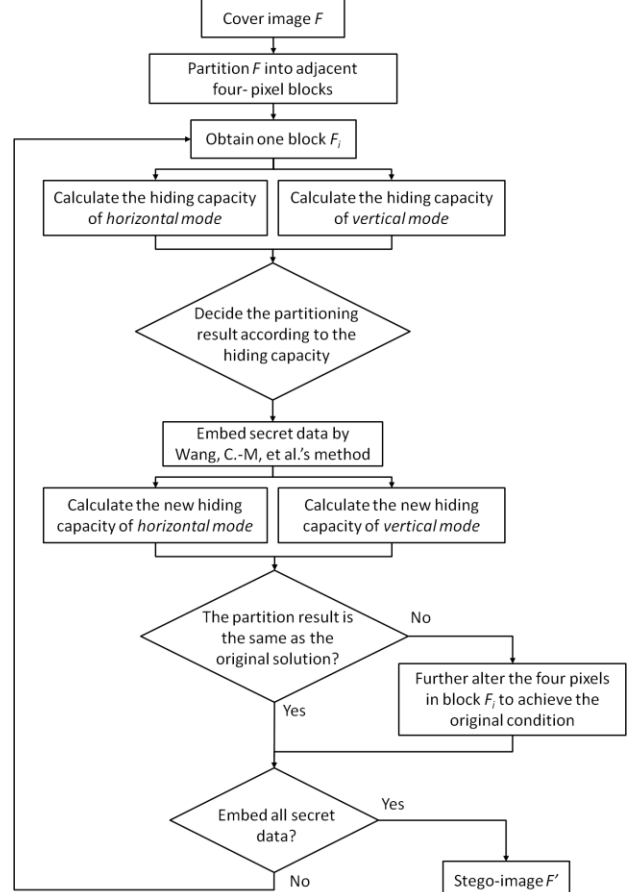


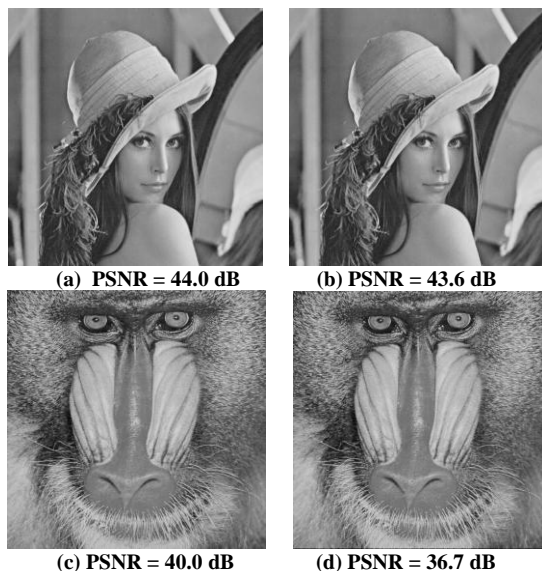
Fig. 4 The flowchart of the proposed embedding method

Table I shows the comparison of results of the proposed and Wang, C.-M, et al.'s method. On average, the proposed method has about 1500 bytes greater hiding capacity than Wang, C.-M, et al.'s method. This is because proposed method provides *vertical mode* which can embed more hiding data. The performance varies in accordance with the characteristic of the cover image. If the cover image has more edge-like areas in vertical direction than in horizontal one, like "Baboon," the more *vertical modes* will be chosen as the partitioning results, and the hiding capacity of the corresponding stego-image may be larger. However, the PSNR value will drop greater. In general, the difference on the stego-image is hard to be recognized by human vision when the PSNR value is greater than 30dB [8], [9]. The experimental results show that all of the stego-images keep high visual quality and have large hiding capacity. Two of the stego-images produced by proposed and Wang, C.-M, et al.'s method are shown in Fig. 5.

**TABLE I**  
COMPARISON OF RESULTS OF THE PROPOSED AND WANG, C.-M, ET AL.'S METHOD

Cover-images (512x512)	Wang C.-M, et al.'s method		Our method	
	Capacity (bytes)	PSNR (dB)	Capacity (bytes)	PSNR (dB)
Lena	51225.88	44.0	51574.75	43.6
Peppers	50955.38	42.4	51659.63	41.6
Baboon	57138.13	40.0	61728.38	36.7
Barbara	56655.88	39.0	57186.38	38.9
Goldhill	51487	43.9	52810.88	43.0
Average	53492.45	41.9	54992	40.8

Table II shows the partitioning results of the proposed method. Take a close look at the hiding capacity of these modes, the *vertical mode* can embed about 0.9 bits more than *horizontal* one on average. That is the key point of the proposed method which keep more secret messages. We also display the distribution of the partitioning results in Fig. 6 to further analyze the images. The black region denotes *vertical modes*, and the others contain *horizontal modes*. It reveals that the proposed method can exactly find the edge-like *vertical modes*.



**Fig. 5** The stego-image "Lena" produced by (a) Wang, C.-M, et al.'s method and (b) proposed method. The stego-image "Baboon" produced by (c) Wang, C.-M, et al.'s method and (d) proposed method.

## V. CONCLUSION

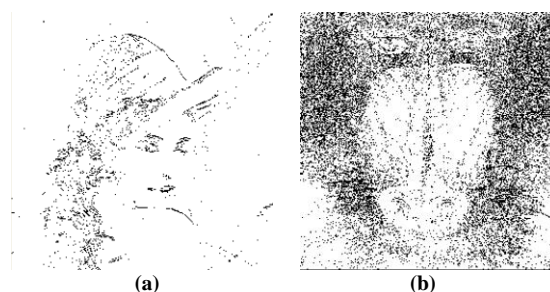
In our proposed method, by comparing the difference values between the adjacent four pixels, the one with more edge-like modes is decided as the chosen mode. Due to the characteristics of human vision resolution which have a larger tolerance in edge areas than in smooth areas, the main idea of the proposed method can find more edge areas in order to hide more secret data.

The experimental results show that the average hiding capacity is 54992 bytes in a 512x512 stego-image, and the PSNR value is 40.8 dB on average. We further analyze the

distribution of the partitioning results to verify the idea of the proposed method. The analysis demonstrated that the choosing procedure exactly finds more edge-like *vertical modes* which can embed more secret data. In conclusion, the proposed method can enhance communication security by providing stego-images which maintain imperceptible quality with large hiding capacity.

**TABLE II**  
THE PARTITIONING RESULTS OF THE PROPOSED METHOD

Cover-images (512x512)	Vertical mode	Horizontal mode	Average hiding capacity(bits)	
			Vertical mode	Horizontal mode
Lena	2062	63474	3.98	3.12
Peppers	2981	62555	4.20	3.10
Baboon	17710	47826	4.54	3.48
Barbara	4450	61086	4.10	3.45
Goldhill	6324	59212	3.99	3.14
Average	-	-	4.16	3.26



**Fig. 6** The partitioning results of image (a) "Lena" and (b) "Baboon"

## REFERENCES

- [1] Chan, Chi-Kwong, and Cheng, Lee-Ming. "Hiding data in images by simple LSB substitution." *Pattern recognition* 37.3 (2004): 469-474.
- [2] Wu, Da-Chun, and Tsai, Wen-Hsiang. "A steganographic method for images by pixel-value differencing." *Pattern Recognition Letters* 24.9 (2003): 1613-1626.
- [3] Wu, H.C., Wu, C.S., and Hwang M.S. "Image steganographic scheme based on pixel-value differencing and LSB replacement methods." *IEE Proceedings-Vision, Image and Signal Processing* 152.5 (2005): 611-615.
- [4] Wang, C. M., Wu, N. I., Tsai, C. S., and Hwang, M. S. "A high quality steganographic method with pixel-value differencing and modulus function." *Journal of Systems and Software* 81.1 (2008): 150-158.
- [5] Kim, Ki-Jong, Jung, Ki-Hyun, and Yoo, Kee-Young. "A high capacity data hiding method using PVD and LSB." *Computer Science and Software Engineering, 2008 International Conference on*. Vol. 3. IEEE, 2008.
- [6] Yang, C. H., Weng, C. Y., Tso, H. K., and Wang, S. J. "A data hiding scheme using the varieties of pixel-value differencing in multimedia images." *Journal of Systems and Software* 84.4 (2011): 669-678.
- [7] Mohamed, Marghny H., Naziha M. AL-Aidroos, and Mohamed A. Bamatraf. "A Combined Image Steganography Technique Based on Edge Concept & Dynamic LSB." *International Journal of Engineering* 1.8 (2012).
- [8] Al-Sadi, Azzat A., and M. El Sayed. "An Adaptive Steganographic Method for Color Images Based on LSB Substitution and Pixel Value Differencing." *Advances in Computing and Communications*. Springer Berlin Heidelberg, 2011. 535-544.
- [9] Chou, Yung-Chen, Chang, Chin-Chen and Li, Kuan-Ming. "A large payload data embedding technique for color images." *Fundamenta Informaticae* 88.1 (2008): 47-61.