International Journal of Security and Its Applications
Vol.7, No.6 (2013), pp.285-294
http://dx.doi.org/10.14257/ijsia.2013.7.6.29

Steganography in Digital Images Using Maximum Difference of Neighboring Pixel Values

Gandharba Swain

Department of Information Technology, GMR Institute of Technology, Rajam-532127, Andhra Pradesh, India
gswain1234@gmail.com

Abstract

In this paper a pixel value differencing steganography based on the maximum difference of the neighboring pixel values have been proposed. There are four variants such as five, six, seven and eight neighbor differencing. In five neighbor differencing method the maximum difference amongst the five neighboring pixels such as upper, left, right, bottom and upper-right corner are used to take embedding decision. In six neighbor differencing method the maximum difference amongst the six neighboring pixels such as upper, left, right, bottom, upper-right corner, and upper-left corner are used to take embedding decision. In seven neighbor differencing method the maximum difference amongst the seven neighboring pixels such as upper, left, right, bottom, upper-right corner, upper-left corner and bottom-left corner are used to take embedding decision. In eight neighbor differencing method the maximum difference amongst all the eight neighboring pixels are used to take embedding decision. The message extraction process is very simple and does not require any knowledge of the original image. The experimental results show that the distortion is the minimum in five neighbor differencing. But the theoretical studies reveal that the hiding capacity is the highest in eight neighbor differencing.

Keywords: *steganography, pixel value differencing, five neighbor differencing, six neighbor differencing, seven neighbor differencing, eight neighbor differencing*

1. Introduction

Steganography is a technique for covert communication. It can be achieved by using carriers like image, audio and video [1]. Martin *et al.*, [2] had experimentally investigated that by adding the message inside an image, there is a change in statistics, but if this change is very small it can not be detected. In image carrier normally we use the least significant bit (LSB) substitution method. In this method the 8^{th} bit of every byte of the carrier file is substituted by one bit of the secret information. Instead of hiding a fixed number of bits in the LSBs of each pixel, one can also embed different number of bits in LSBs of different pixels based on pixel value range calculation [3]. If the secret message is shorter then only the LSBs of darkest and brightest pixels can be used to enhance the security [4, 5]. Zhang *et al.*, proposed a steganography method by adding 1 to the gray value or subtracting 1 from the gray value [6]. To achieve better security message bit dependent LSB steganography can also be used [7-9].

An RGB image is one in which each pixel is represented by 3 bytes to represent the intensities of red, green and blue channels. Parvez and Gutub [10] proposed a technique based on RGB intensity values of the pixels. They took one of the channels as indicator channel and used one or both of the remaining two channels to hide data bits. The last two bits of the

ISSN: 1738-9976 IJSIA

indicator channel indicate whether the data bits are hidden in the other two channels or not. Tiwari and Shandilya [11] proposed two techniques based on RGB concept. In the first technique they modified the technique of Parvez and Gutub, by changing the indicator for every subsequent pixel. The second technique is based on the generation of a random number. The generated random number determines the number of least significant bits that is used to hide the secret data. One can embed up to four LSBs in the data channels [12]. By associating some conditions in indicator based techniques the security level can be improved [13]. A block based RGB steganography has been proposed by Swain and Lenka [14]; wherein the image is divided into 8 blocks, the message is divided into 8 blocks and one message block is allocated to one image block through a user defined key. In each image block, one of the channels is made as indicator channel and the remaining two channels are used for embedding. The least significant bits can be treated as an array of bits and the message can be mapped into it and be embedded at maximum matching portion of the array so that distortion can be reduced [15].

If the pixels are located in edge areas they can tolerate larger changes than those in smooth areas. The range of changeable pixel value in smooth areas is small, whereas in edge areas it is large so that the stego-image maintains a good perceptual quality. Wu and Tsai [16] proposed a pixel value differencing (PVD) method, wherein a cover image is partitioned into non overlapping blocks of two consecutive pixels. A difference value is calculated from the value of the two pixels in each block. Secret data is embedded into the cover image by replacing the difference values of the two pixel blocks of the cover image with similar ones. But it has been found that PVD steganography is vulnerable to histogram based attacks, so it should be taken care [17]. To increase the hiding capacity tri-way PVD can also be used by taking four pixels as a block and computing three directional differences [18, 19]. Chang and Tseng [20] proposed two sided, three sided and four sided side match methods by using the side information of two, three and four neighboring pixels respectively. A similar method is proposed in [21]. Hossain et al. [22] also proposed four sided and eight sided side match methods. In four sided method four neighboring pixels were used to take embedding decision and in eight neighbor method eight neighboring pixels were used to take embedding decision. The fall in error problem for two sided, three sided and four sided side match methods have been addressed by Swain and Lenka [23]. Liao *et al.*, [24] proposed four pixel differencing and modified LSB substitution. Yang et al. [25] proposed a PVD steganography by taking four pixel blocks at a time and observed more capacity compared to that of Wu and Tsai's method.

In this paper a pixel value differencing technique is proposed, wherein the number of bits to be embedded in a target pixel depends upon the largest difference amongst the neighboring pixel values. There are four variants of this technique. In section 2 the proposed technique is explained. In section 3 the experimental results are discussed and in section 4 the paper is concluded.

2. The Proposed Methods

2.1. Five Neighbor Differencing

The secret data is embedded into an image by accessing the pixels in raster-scan order. The pixels in the image are categorized into three categories as shown in Figure 1. The pixels which are marked by a darker color are used as target pixels, where data is to be hidden. The white colored pixels are used as neighbors. The pixels marked with light gray color are neither treated as neighbor nor to hide data.

This method uses the upper, left, right, bottom and upper-right corner neighboring pixels for estimating the number of bits to be embedded in the target pixel. Let g_x be the gray value

of the target pixel P_x; g_u, g_l, g_r, g_b and g_{ur} be the gray values of its upper pixel P_u, left pixel P_l, right pixel P_r, bottom pixel P_b and upper-right corner pixel P_{ur} respectively. Then the difference value d is computed using equation 1.

$$d=g_{max}-g_{min} \quad (1)$$

where $g_{max} = \max(g_u, g_l, g_r, g_b, g_{ur})$ and $g_{min} = \min(g_u, g_l, g_r, g_b, g_{ur})$

The embedding capacity of pixel depends on the value of d. Suppose n be the number of bits which can be embedded in the target pixel P_x, then n is calculated using equation 2.

$$n = \begin{cases} 1, & \text{if } 0 \le d \le 1 \\ \log_2 d, & \text{if } d > 1 \end{cases} \quad (2)$$

As the quality of the stego-image may degrade drastically when n>4, so set n≡4. A sub-stream of n bits from the secret binary message is taken and is converted to integer b. Then the new value of the target pixel, g'_x is computed as in equation 3.

$$g'_x = g_x - g_x \bmod 2^n + b \quad (3)$$

Suppose δ_x be the difference between g_x and g'_x i.e. $\delta_x = g'_x - g_x$, then the following two adjustments shown in case 1 and case 2 below are done.

Case1: If $2^{n-1} < \delta_x < 2^n$ and $g'_x \ge 2^n$ then $g'_x = g'_x - 2^n$

Case 2: If $-2^n < \delta_x < -2^{n-1}$ and $g'_x < 256 - 2^n$ then $g'_x = g'_x + 2^n$

The secret data is extracted from the stego-image by scanning it in raster-scan order. Given a target pixel P^*_x with gray value g^*_x, let g^*_u, g^*_l, g^*_r, g^*_b and g^*_{ur} be the gray values of its upper pixel P^*_u, left pixel P^*_l, right pixel P^*_r, bottom pixel P^*_b and upper-right pixel P^*_{ur} respectively. The gray value difference d^* is computed using equation 4.

$$d^* = g^*_{max} - g^*_{min} \quad (4)$$

where $g^*_{max} = \max(g^*_u, g^*_l, g^*_r, g^*_b, g^*_{ur})$ and $g^*_{min} = \min(g^*_u, g^*_l, g^*_r, g^*_b, g^*_{ur})$

Let n^* be the number of bits which can be extracted from the target pixel P^*_x. The value n^* is calculated as in equation 5.

$$n^* = \begin{cases} 1, & \text{if } 0 \le d^* \le 1 \\ \log_2 d^*, & \text{if } d^* > 1 \end{cases} \quad (5)$$

If $n^* > 4$, then set $n \equiv 4$ and the value b is calculated as in equation 6.

$$b = g^*_x \bmod 2^{n^*} \quad (6)$$

Finally, n^* bits secret data can be obtained by converting the value b to a binary string.

2.2. Six Neighbor Differencing

The secret data is embedded into an image by accessing the pixels in raster-scan order. The pixels are categorized into three categories as shown in Figure 2. The pixels which are marked with a darker color are used as target pixels, where data is to be hidden. The white colored pixels are used as neighbors. The pixels marked with light gray color are neither treated as neighbors nor to hide data.

This method uses the upper, left, right, bottom, upper-left corner and upper-right corner neighboring pixels for estimating the number of bits to be embedded in the target pixel. Let g_x

be the gray value of the target pixel P_x ; g_u, g_l , g_r , g_b , g_{ul} and g_{ur} be the gray values of its upper pixel P_u, left pixel P_l , right pixel P_r ,bottom pixel P_b, upper-left corner pixel P_{ul} and upper-right corner pixel P_{ur} respectively. Then the difference value d is computed using equation 7.

$$d = g_{max} - g_{min} \tag{7}$$

where $g_{max} = \max(g_u, g_l, g_r, g_b, g_{ul}, g_{ur})$ and $g_{min} = \min(g_u, g_l, g_r, g_b, g_{ul}, g_{ur})$

The embedding capacity of a pixel depends on the value of d. Suppose n be the number of bits which can be embedded in the target pixel P_x, then n is calculated as in equation 8.

$$n = \begin{cases} 1, & \text{if } 0 \le d \le 1 \\ \log_2 d, & \text{if } d > 1 \end{cases} \tag{8}$$

As the quality of the stego-image may degrade drastically when n>4, so set n≡4. A sub-stream of n bits from the secret binary message is taken and is converted to integer b. Then the new value of the target pixel, g'_x is computed as in equation 9.

$$g'_x = g_x - g_x \bmod 2^n + b \tag{9}$$

After getting g'_x like this, the two cases for adjustment are applied as in five neighbor differencing method. The extraction procedure is also identical to that of five neighbor differencing method.

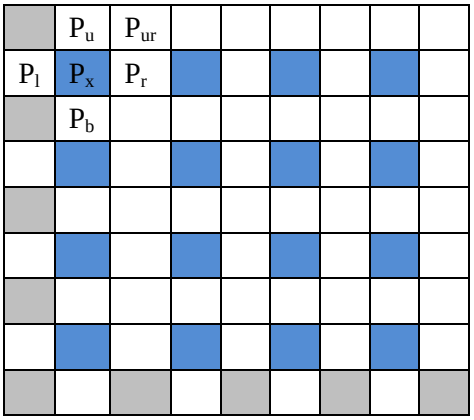


Figure 1. Sampling arrangement of pixels in five neighbor differencing method

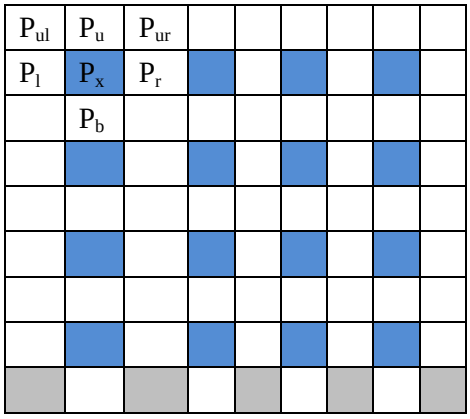


Figure 2. Sampling arrangement of pixels in six neighbor differencing method

2.3. Seven Neighbor Differencing

The secret data is embedded into an image by accessing the pixels in raster-scan order. The pixels are categorized into three categories as shown in Figure 3. The pixels which are marked by a darker color are used as target pixels, where data is to be hidden. The white colored pixels are used as neighbors. The pixel marked with light gray color is neither treated as neighbor nor to hide data.

This method uses the upper, left, right, bottom, upper-left, upper-right and bottom-left neighbor pixels for estimating the number of bits to be embedded in the target pixel. Let g_x be the gray value of the target pixel P_x ; g_u, g_l , g_r , g_b , g_{ul} , g_{ur} and g_{bl} be the gray values of its upper pixel P_u, left pixel P_l , right pixel P_r ,bottom pixel P_b, upper-left pixel P_{ul} , upper-right

pixel P_{ur} and bottom-left pixel P_{bl} respectively. Then the difference value d is computed as in equation 10.

$$d = g_{max} - g_{min} \tag{10}$$

where $g_{max} = \max(g_u, g_l, g_r, g_b, g_{ul}, g_{ur}, g_{bl})$ and $g_{min} = \min(g_u, g_l, g_r, g_b, g_{ul}, g_{ur}, g_{bl})$

The embedding capacity of the pixel depends on the value of d. Suppose n be the number of bits which can be embedded in the target pixel P_x, then n is calculated as in equation 11.

$$n = \begin{cases} 1, & \text{if } 0 \le d \le 1 \\ \log_2 d, & \text{if } d > 1 \end{cases} \tag{11}$$

As the quality of the stego-image may degrade drastically when n>4, so set n≡4. A sub-stream of n bits from the secret binary message is taken and is converted to integer b. Then the new value of the target pixel, g'_x is computed as in equation 12.

$$g'_x = g_x - g_x \bmod 2^n + b \tag{12}$$

After getting g'_x like this, the two cases for adjustment are applied as in five neighbor differencing method. The extraction procedure is also identical to that of five neighbor differencing method.

2.4. Eight Neighbor Differencing

The secret data is embedded into an image by accessing the pixels in raster-scan order. The pixels are categorized into two categories as shown in Figure 4. The pixels which are marked by a darker color are used as target pixels, where data is to be hidden. The white colored pixels are used as neighbors.

This method uses the upper, left, right, bottom, upper-left corner, upper-right corner, bottom-left corner and bottom-right corner neighboring pixels for estimating the number of bits to be embedded in the target pixel. Let g_x be the gray value of the target pixel P_x; g_u, g_l, g_r, g_b, g_{ul}, g_{ur}, g_{bl} and g_{br} be the gray values of its upper pixel P_u, left pixel P_l, right pixel P_r, bottom pixel P_b, upper-left pixel P_{ul}, upper-right pixel P_{ur}, bottom-left pixel P_{bl} and bottom-right pixel P_{br} respectively. Then the difference value d is computed as in equation 13.

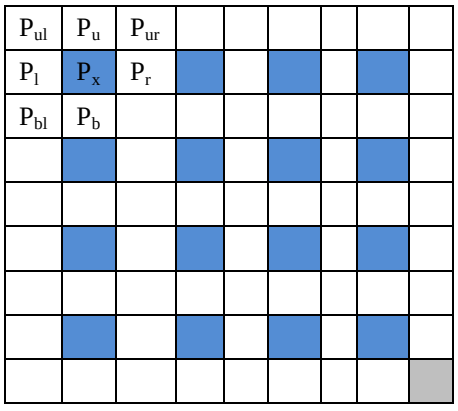


Figure 3. Sampling arrangement of pixels in seven neighbor differencing method

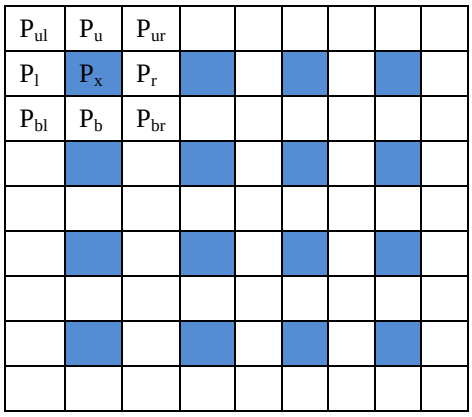


Figure 4. Sampling arrangement of pixels in eight neighbor differencing method

$$d = g_{max} - g_{min} \quad (13)$$

where $g_{max} = \max(g_u, g_l, g_r, g_b, g_{ul}, g_{ur}, g_{bl}, g_{br})$ and $g_{min} = \min(g_u, g_l, g_r, g_b, g_{ul}, g_{ur}, g_{bl}, g_{br})$

The embedding capacity of pixel depends on the value of d. Suppose n be the number of bits which can be embedded in the target pixel P_x, then n is calculated as in equation 14.

$$n = \begin{cases} 1, & \text{if } 0 \le d \le 1 \\ \log_2 d, & \text{if } d > 1 \end{cases} \quad (14)$$

As the quality of the stego-image may degrade drastically when n>4, so set n≡4. A sub-stream of n bits from the secret binary message is taken and is converted to integer b. Then the new value of target pixel, g'_x is computed as in equation 15.

$$g'_x = g_x - g_x \bmod 2^n + b \quad (15)$$

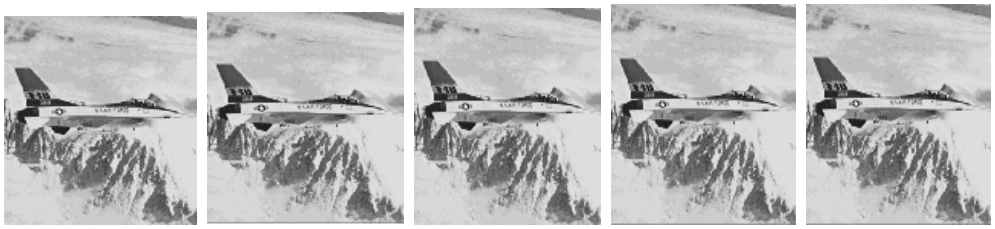
After getting g'_x like this, the two cases for adjustment are applied as in five neighbor differencing method. The extraction procedure is also identical to that of five neighbor differencing method.

3. Results and Discussion

3.1. Results

These methods are implemented using MATLAB tested with many images. The observations for four standard images are as shown below.

Figure 5a is the Airplane image with a size of 192 kb. Figure 5b-e are the stego-images in five, six, seven and eight neighbor differencing methods respectively with 2048 bytes of data hidden in each and Figure 5f-j are their respective histograms. Figure 6a is the Boat image with a size of 768 kb. Figure 6b-e are the stego-images in five, six, seven and eight neighbor differencing methods respectively with 1024 bytes of data hidden in each and Figure 6f-j are their respective histograms. Figure 7a is the House image with a size of 768 kb. Figure 7b-e are the stego-images in five, six, seven and eight neighbor differencing methods respectively with 1024 bytes of data hidden in each and Figure 7f-j are their respective histograms. Figure 8a is the Baboon image with a size of 525 kb. Figure 8b-e are the stego-images in five, six, seven and eight neighbor differencing methods respectively with 2048 bytes of data hidden in each and Figure 8f-j are their respective histograms.



(a) **(b)** **(c)** **(d)** **(e)**

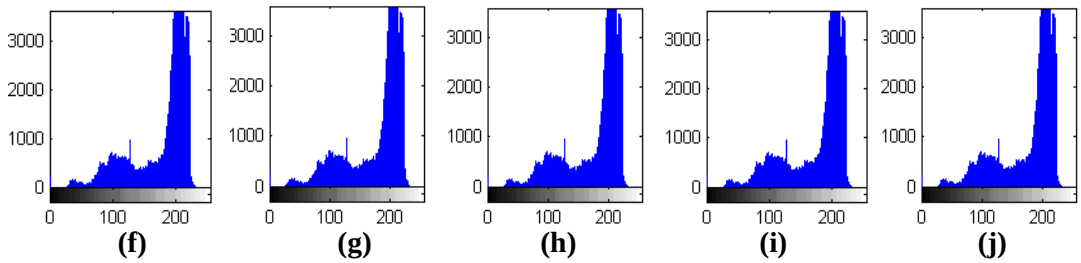


Figure 5. a Airplane image; b, c, d, e are stego-images; f, g, h, i, j are their histograms

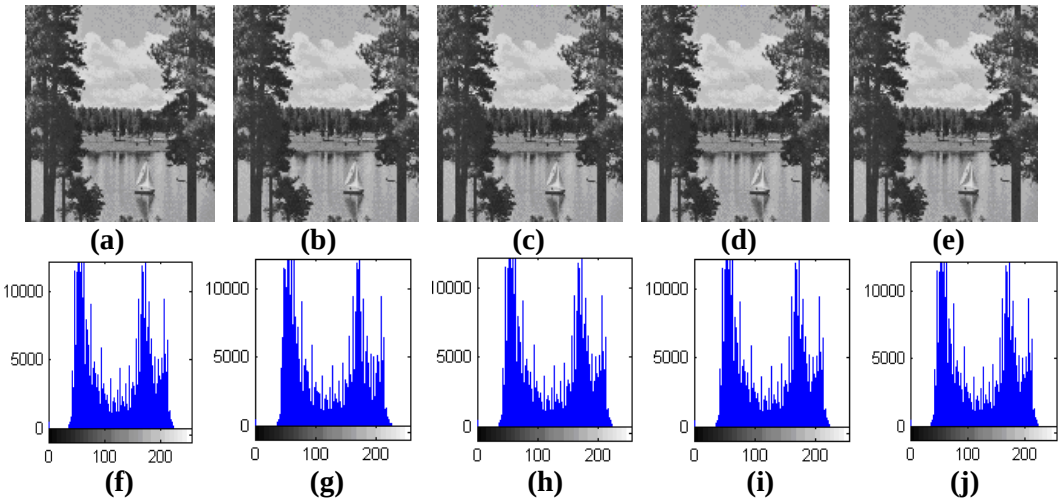


Figure 6. a Boat image; b, c, d , e are stego-images; f, g, h, i, j are their histograms

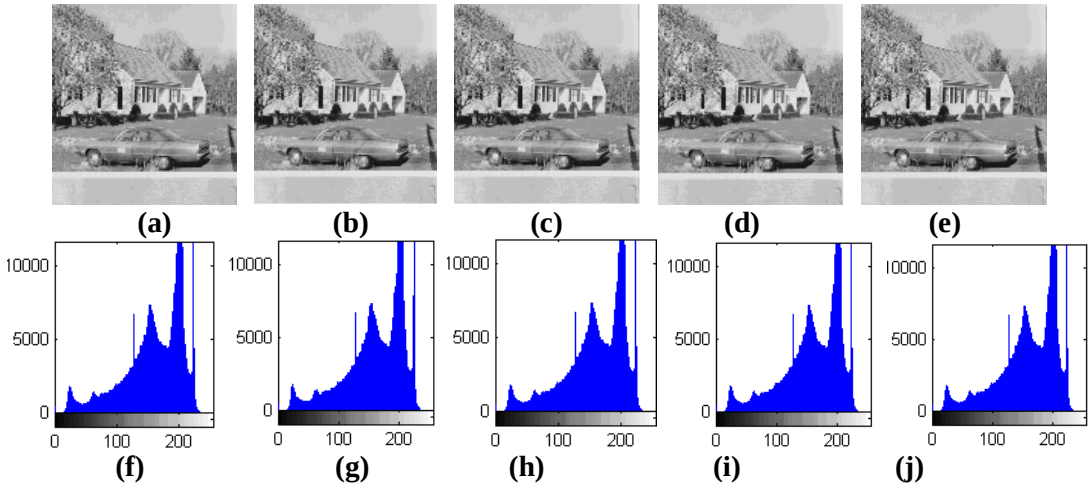


Figure 7. a House image; b, c, d, e are stego-images; f, g, h, i, j are their histograms

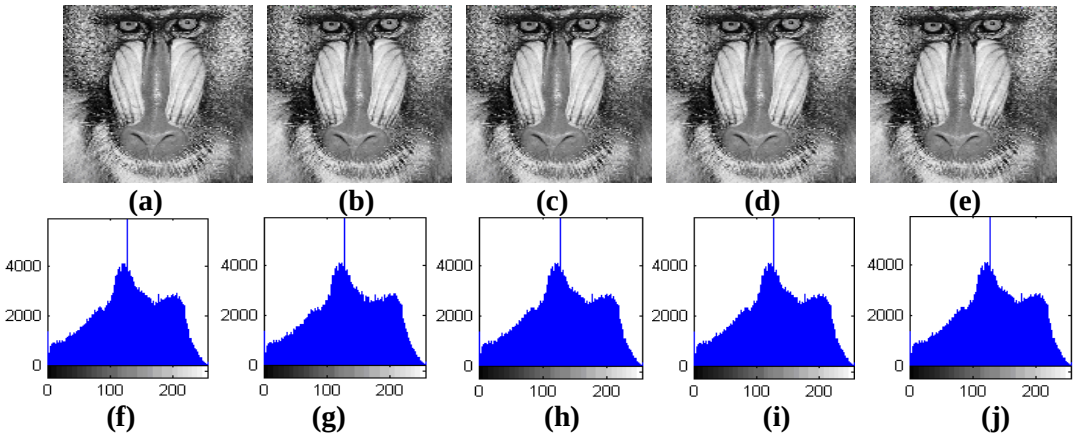


Figure 8. a Baboon image; b, c, d, e are stego-images; f, g, h, i, j are their histograms

3.2. Discussion

The steganographic methods proposed in this paper are secured because instead of replacing the LSBs of pixel values directly, the proposed methods adaptively change the pixel value into another value based on the maximum difference of neighboring pixel values. The traditional LSB steganography have a common weak point. The pixel value changes asymmetrically which can be captured by RS steganalysis. The proposed methods are tolerant to steganalysis methods like RS steganalysis because we are changing variable number of bits in different pixels. The histograms of original images and stego-images are completely identical. Thus histogram based steganalysis attacks are not possible.

For any steganography technique the hiding capacity should be as high as possible. We can see in Figures 1, 2, 3 and 4 that almost 20% of the pixels can be used to hide data and in each pixel up to a maximum of 4 bits. Thus the hiding capacity is also good.

Without compromising the levels of security and capacity, if the imperceptibility is kept higher, then the technique is preferable. If the stego-image looks innocuous one can believe that, this requirement is fulfilled. In Figures 5, 6, 7 and 8 it can be observed that the stego-images are looking very innocuous. Moreover, the distortion in a stego-image can be measured by peak signal-to-noise ratio (PSNR). The lesser is the PSNR means more is the distortion.

The amount of data hidden and peak signal-to-noise ratio (PSNR) values are shown in Table 1 for the four tested images. With the same amount of hidden data the distortion is minimum in five neighbor case than compared to six, seven and eight neighbor cases. But eight neighbor case can hide maximum amount of data compared to five, six and seven neighbor cases. This is because the difference value d is largest in eight neighbor case.

Table 1. The PSNR values for the tested images

Image name	Image size (in kb)	Hidden data size (in bytes)	Peak signal-to-noise ratio (in dB)			
			Five neighbor	Six neighbor	Seven neighbor	Eight neighbor
Airplane	192	2048	49.91	49.51	49.13	49.41
Boat	768	1024	50.90	50.70	50.89	50.58
House	768	1024	54.30	53.82	54.22	53.79
Baboon	525	2048	44.05	43.71	43.61	43.26

The Chang and Tseng's [20] two sided side match method is as follows. Given an input pixel P_x with gray value g_x, let g_u and g_l be the gray values of its upper neighboring pixel P_u and left neighboring pixel P_l respectively. Then a difference value d is computed as $d = (g_u + g_l)/2 - g_x$. Similarly for three sided match $d = (g_u + g_l + g_r) / 3 - g_x$, for four sided match $d = (g_{lu} + g_{ru} + g_{lb} + g_{rb}) / 4 - g_x$ or $(g_u + g_l + g_r + g_b) / 4 - g_x$. The number of bits to be embedded in a target pixel depends upon this difference value d. As the proposed method can give a larger difference value d, so embedding capacity is higher.

Hossain *et al.*, [22] proposed four neighbor and eight neighbor methods similar to that of side match methods. The d value is calculated in same way as that of side match methods. The proposed method calculates d value by taking the largest difference value among the neighboring pixel values. So embedding capacity is obviously higher.

4. Conclusion

In this paper a new image steganography technique by considering the maximum difference of neighboring pixel values have been proposed. There are four variants such as five, six, seven and eight neighbor differencing methods. The embedding capacity and peak signal-to-noise ratio values are good in all the four cases. With same amount of hidden data the distortion is minimum in five neighbor differencing method compared to other three methods. But eight neighbor differencing method can hide maximum amount of data compared to other three methods. This is because the difference value d is largest in eight neighbor case. After the information is embedded the change in quality of the images are not noticeable. The extraction process is very simple and does not require the original image.

References

[1] J. C. Cheddad, K. Curran and P. M. Kevitt, "Digital Image Steganography Survey and Analysis of Current Methods", Signal Processing, vol. 90, **(2010)**, pp. 727-752.
[2] A. Martin, G. Sapiro and G. Seroussi, "Is Image Steganography Natural", IEEE Transactions on Image Processing, vol.14, no.12, **(2005)**, pp. 2040-2050.
[3] Y. K. Jain and R. R. Ahirwal, "A Novel Image Steganography Method with Adaptive Number of Least Significant Bits Modification Based on Private Stego-Keys", International Journal of Computer Science and Security, vol. 4, no. 1, **(2010)**, pp. 40-49.
[4] G. Swain and S. K. Lenka, "A Hybrid Approach to Steganography- Embedding at Darkest and Brightest Pixels", International Conference on Communication and Computational Intelligence, **(2010)**, pp. 529-534.
[5] G. Swain and S. K. Lenka, "Application of a Large Key Cipher in Image Steganography by Exploring the Darkest and Brightest Pixels", International Journal of Computer Science and Communication, vol. 3, no. 1, **(2012)**, pp. 49-53.
[6] W. Zhang, X. Zhang and S. Wang, "A Double Layered Plus-Minus One Data Embedding Scheme", IEEE Signal Processing Letters, vol. 14, no. 11, **(2007)**, pp. 848-851.
[7] G. Swain and S. K. Lenka, "A Robust Image Steganography Technique Using Dynamic Embedding With Two Least Significant Bits", Advanced Materials Research, vols. 403-408, **(2012)**, pp. 835-841.
[8] G. Swain and S. K. Lenka, "A Dynamic Approach to Image Steganography Using the Three Least Significant Bits and Extended Hill Cipher", Advanced Materials Research, vols. 403-408, **(2012)**, pp. 842-849.
[9] G. Swain and S. K. Lenka, "A Technique for Secret Communication by Using a New Block Cipher with Dynamic Steganography", International Journal of Security and Its Applications. vol. 6, no. 2, **(2012)**, pp. 1-12.
[10] M. T. Parvez and A. A. Gutub, "RGB Based Variable-bits Image Steganography", In Proceedings of IEEE Asia Pacific Services Computing Conference, **(2008)**, pp. 1322-1327.
[11] N. Tiwari and M. Shandilya, "Secure RGB Image Steganography from Pixel Indicator to Triple Algorithm-an Incremental Growth", International Journal of Security and Its Applications, vol.4, no.4, **(2010)**, pp.53-62.
[12] M. Kaur, S. Gupta, P.S. Sandhu and J. Kaur, "A Dynamic RGB Intensity Based Steganography Scheme", World Academy of Science, Engineering and Technology, vol. 67, **(2010)**, pp. 833-838.
[13] G. Swain and S. K. Lenka, "A Better RGB Channel Based Image Steganography Technique", CCIS, vol. 270, no. 2, **(2012)**, pp. 470-478.

[14] G. Swain and S.K. Lenka, "A Novel Approach to RGB Channel Based Image Steganography Technique", International Arab Journal of e-Technology, vol. 2, no. 4, **(2012)**, pp. 181-186.
[15] G. Swain and S. K. Lenka, "LSB Array Based Image Steganography Technique by Exploring the Four Least Significant Bits", CCIS, vol. 270, no. 2, **(2012)**, pp. 479-488.
[16] D. C. Wu and W. H. Tsai, "A Steganograhic Method for Images by Pixel Value Differencing", Pattern Recognition Letters, vol. 24, no. 9-10, **(2003)**, pp. 1613-1626.
[17] X. Zhang and S. Wang, "Vulnerability of Pixel Value Differencing Steganography to Histogram Analysis and Modification for Enhanced Security", Pattern Recognition Letters, vol. 25, **(2004)**, pp. 331-339.
[18] K. C. Chang, C. P. Chang, P. S. Huang and T. M. Tu, "A Novel Image Steganography Method Using Tri-way Pixel Value Differencing", Journal of Multimedia, vol. 3, no. 2, **(2008)**, pp. 37-44.
[19] Y. P. Lee, J. C. Lee, W. K. Chen, K. C. Chang, I. J. Su and C. P. Chang, "High-Payload Image Hiding With Quality Recovery Using Tri-Way Pixel-Value Differencing", Information Sciences, vol. 191, **(2012)**, pp. 214-225.
[20] C. C. Chang and H. W. Tseng, "A Steganographic Method for Digital Images Using Side Match", Pattern Recognition Letters, vol. 25, no. 12, **(2004)**, pp. 1431-1437.
[21] K. J. Kim, K. H. Jung and K. Y. Yoo, "Image Steganographic Method with Variable Embedding Length", International Symposium on Ubiquitous Computing, **(2008)**, pp. 210-213.
[22] M. Hossain, S. A. Haque and F. Sharmin, "Variable Rate Steganography in Gray Scale Digital Images Using Neighborhood Pixel Information", The International Arab Journal of Information Technology, vol. 7, no. 1, **(2010)**, pp. 34-38.
[23] G. Swain and S. K. Lenka, "Steganography Using Two Sided, Three Sided, and Four Sided Side Match Methods", CSIT, vol. 1, no. 2, **(2013)**, pp. 127-133.
[24] X. Liao, Q. Y. Wen and J. Zhang, "A Steganographic Method for Digital Images with Four-Pixel Differencing and Modified LSB Substitution", Journal of Visual Communication and Image Representation, vol. 22, **(2011)**, pp. 1-8.
[25] C. H. Yang, C. Y. Weng, H. K. Tso and S. J. Wang, "A Data Hiding Scheme Using the Varieties of Pixel-Value Differencing in Multimedia Images", The Journal of Systems and Software, vol. 84, **(2011)**, pp. 669-678.

Author

Prof. Gandharba Swain is working as an Associate Professor in the Department of Information Technology, GMR Institute of Technology, Rajam, Andhra Pradesh, India. He received B.Sc(Hons) degree from Berhampur University in 1995, MCA degree from VSS University of Technology (formerly UCE), Burla, in 1999, M.Tech (CSE) degree from NIT, Rourkela, in 2004. He has more than 14 years of teaching experience. He has authored one text book, published several research articles in international journals and conferences. His research interests include network security, image steganography and water marking.

