Pixel-Value Differencing Steganography: Attacks and Improvements

El-Sayed M. El-Alfy
College of Computer Sciences and Engineering
King Fahd University of Petroleum and Minerals
Dhahran 31261, Saudi Arabia
alfy@kfupm.edu.sa

Azzat A. Al-Sadi
College of Computer Sciences and Engineering
King Fahd University of Petroleum and Minerals
Dhahran 31261, Saudi Arabia
azzat.sadi@gmail.com

Abstract—**Hiding confidential data in digital images using the pixel-value differencing (PVD) method provide higher embedding capacity without very noticeable artifacts in the cover image to human eyes. However, the presence of hidden data can be revealed by a number of automatic approaches that can detect variations in statistical properties of the image due to embedding such as histogram analysis, chi-square test and universal detectors. This paper aims at reviewing different steganalytic techniques to attack the pixel-value differencing method. It also surveys several proposed methods in the literature to enhance the security of PVD against commonly known attacks.**

Keywords-steganography; steganalysis; pixel-value difference; security attacks; histogram analysis; universal detectors.

I. INTRODUCTION

With the rapid development and popularity of the Internet, protecting the confidentiality of data while being transmitted over the network has become a significant challenge. The essence of steganography is to hide the very presence of secret data [1, 2]. It conceals the secret data into another medium to protect it against unauthorized access. Data can be encrypted before applying steganography to increase the security. Many steganographic approaches have been developed over years for various types of cover media: text, image, audio and video. However, the pervasive application of steganography is using digital images as the cover media. This is due to their computational simplicity and extensive use over the Internet with a variety of image extensions. Although steganography has been known for a long time, it became imperative in the recent age of information technology.

Several steganographic algorithms have been proposed for embedding data in digital images as cover media, whether in spatial or frequency domain. Among various criteria that have been used to evaluate the effectiveness of a steganographic method is how secure it is against detection (a.k.a. resistance to steganalysis) [3]. Other criteria include embedding capacity and invisibility to naked human eyes. One of the relatively recent techniques is based on pixel value differencing (PVD) was proposed by Wu and Tsai [4]. Unlike the popular least-significant bit (LSB) steganographic method [3, 5, 6] which has been early proposed due to its simplicity, the PVD method adapts the number of embedded bits to the grayscale/color changes in consecutive pixels. This leads to increasing the embedding capacity without significant loss of image quality. As new attacks defeating PVD have been discovered, a number of other methods have been proposed in the literature to modify the original PVD method.

Although in the literature several studies evaluate various steganographic techniques against different attacks [7]-[12], few efforts have been attempted on PVD. Examples of known steganographic attacks reported in the literature are chi-square attack, RS analysis, sample pair analysis (SPA), weighted stego (WS) analysis, structural steganalysis, and blind or universal steganalysis [13]. Our aim in this paper is to review and discuss different steganographic attacks on PVD and survey the proposed methods to enhance the security of PVD against different attacks.

The rest of this paper is organized as follows. In the following section, we provide a brief description of the pixel-value differencing (PVD) method. Then, in Section III, we discuss PVD attacking techniques. In Section IV, we review several methods that have been proposed in the literature to improve the security of PVD. Finally in Section V, we conclude the paper.

II. PVD BACKGROUND

The pixel-value differencing (PVD) method was originally proposed to hide secret messages into 256 gray-valued images [4]. It can embed larger amount of data without much degradation in the image quality and thus are hardly noticeable by human eyes (i.e. more resistant to visual attacks than the traditional LSB). It is based on the fact that human eyes can easily observe small changes in the gray values of smooth areas in the image but they cannot observe relatively larger changes at the edges areas. PVD uses the difference of each pair of pixels to determine the number of message bits that can be embedded into that pixel pair. It starts at the upper-left corner of the cover image and scans the image in a zigzag manner as illustrated in Fig. 1. Then, it partitions the resulting sequence

into blocks where each block consists of two consecutive non-overlapping pixels. The differences of the two-pixel blocks are used to categorize the smoothness properties of the cover image. Pixels around an edge area will have larger differences whereas pixels at a smooth area will have smaller differences. The larger the difference, the more the bits that can be embedded into that pixel pair.

Thus, instead of inserting a fixed number of bits into each pixel, as the least significant bit replacement method does, PVD adapts the number of embedded bits to the characteristics of each pixel pair. In order to accomplish that, the range of gray values (0, 255) is divided into smaller ranges and each range r_i is demarcated by lower and upper boundary, l_i and u_i, respectively. Then, the absolute value of the difference for each pixel pair is located into one range and the number of bits to be embedded into this pixel pair is determined by the width of this particular range. The width of range r_i is $w_i = u_i - l_i + 1$ and hence the number of bits to be embedded is given by $n_i = log_2 w_i$. Ranges close to the 0 bound represent smoother areas and thus have smaller widths. Similarly ranges close 255 represent clearer edges and thus have larger widths. Although widths of ranges can take any values, it is common to use values that are powers of 2 and grow exponentially as they move away from the 0 bound. In other words, the width of the first range is 8, the width of the second range is 16, and so on. The authors of PVD have tested two different sets of values for the range widths: {8, 8, 16, 32, 64, 128} and {2, 2, 4, 4, 4, 8, 8, 16, 16, 32, 32, 64, 64}. Note that the sum of all the values in each case should be 256.

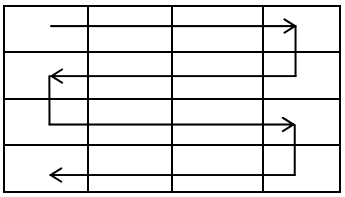


Figure 1. PVD zigzag scan of an image.

Once the number of bits to be embedded is determined for a particular pixel pair, the embedding process is executed as follows. Assume a pixel pair is denoted as p_j and p_{j+1} with gray values g_j and g_{j+1}, respectively, where j is the index of the first-pixel in the block as per the scanned sequence. The difference d_j of this pixel pair is calculated from $g_{j+1} - g_j$ which is a value from -255 to +255. The absolute value of the difference $|d_j|$ falls in the range from 0 to 255. Assume this difference also falls in the range r_i with lower bound l_i and width w_i. The embedding takes place in both pixels in the block to generate new gray scale values for these pixels such that the new difference is given by:

$$d'_j = \begin{cases} l_i + b_k & \text{for } d \geq 0 \\ -(l_i + b_k) & \text{otherwise} \end{cases} \quad (1)$$

where b_k is the equivalent decimal value of some secret bits to be embedded into this block and d'_j is the new pixel difference. Consequently, the new gray values for the block pixels are calculated from:

$$(g'_j, g'_{j+1}) = \begin{cases} (g_j - \lceil m_j \rceil, g_{j+1} + \lfloor m_j \rfloor) & \text{if } d_j \text{ is odd} \\ (g_j - \lfloor m_j \rfloor, g_{j+1} + \lceil m_j \rceil) & \text{if } d_j \text{ is even} \end{cases} \quad (2)$$

where $m_j = (d'_j - d_j)/2$. If the new pixel values fall outside the boundary (0, 255), which is not a valid gray level value, then the secret adata will not be embedded in these pixels. Wu and Tsai proposed a falling-off-boundary process to discover these pixels and skip them. However, this will result in lowering the embedding capacity.

Fig. 2 demonstrates the embedding process of PVD. Although PVD has the potential to hide a large amount of secret data, it has some defects. First of all, only two pixels are considered each time, therefore it cannot capture the different features of edges sufficiently [14]. Second, the falling-off-boundary procedure is a significant problem even with the solution proposed by Wu and Tsai. Third, most of the image is a smooth area, so the secret bits will be hidden in the ranges with small values [15]. Fourth, each pixel in the pixel pair can have different values, therefore it may hide different amount of data from its neighbor. Fifth, the two-pixel block is non-overlapping, and it will lower the embedding capacity [14].

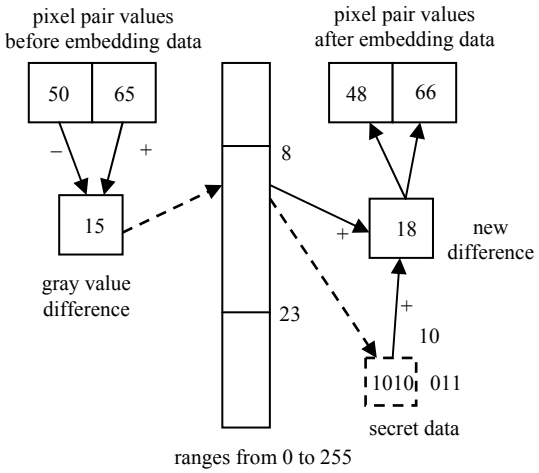


Figure 2. Secret data embedding by PVD [4].

III. PVD ATTACKS

Steganalysis is the art and science of analyzing an object to determine whether it has embedded data (stego-object) or not (cover-object). This discrimination between a stego-object and a cover-object can be with or without the knowledge of the steganographic algorithm that is used for embedding the secret message. Several steganalysis methods have been proposed in the literature. These methods can be classified into two general categories: method-specific methods and universal methods [3], [16], [17]. The first category targets a specific

steganographic approach and attempts to attack that approach. The second category, universal methods which is also sometimes known as blind methods, are more general and can be applied to one or more steganographic approaches. In this category, features that are common to different steganographic approaches are first extracted and a classification model is built. The classifier is then used to detect stego-images.

A. Histogram Attacks

One of the proposed methods for detecting steganography is histogram analysis. Histogram is used to visualize the changes made to the image histogram due to embedding. Image histogram is a graphical representation of the distribution of colors or grayscales in an image. It has been applied to detect embedding by methods based on least-significant bit (LSB) (e.g. LSB replacement and LSB matching) [18], [19], [20]. Although in general visual artifacts are not noticeable by human eyes in the stego-image, changes in the histogram can be easily observed. The pixel value differencing method (PVD) is not very sensitive to straightforward histogram analysis as compared to LSB. However, by drawing the histogram for the differences of pixel pairs, variations before and after embedding can be clearly observed. The histogram of the differences of pixel pairs has a smooth shape of a normal distribution whereas it has remarkable steps for the stego-image. This is due to the quantization ranges of the PVD method. When different differences fall in the same range, the calculation of the new differences will start from the same low boundary of that range. In general, the number of occurrences of a pixel difference decreases with the increase of the absolute value of the difference. In [14], the authors presented an analysis of the changes in the histogram of the pixel difference due to embedding secret data in a cover image using PVD. This analysis can be summarized as follows. The secret bits are assumed to be uniformly distributed (e.g., as a result of encryption before embedding) in [0, w_i-1], where w_i is width of range i. When range $i > 0$ this will make the number of differences falling into r_i, r_0 and r_{i-1} and their boundaries are [l_i u_i], [$-u_0$ u_0] and [$-l_i$ $-u_i$] consequently as shown in Fig. 3. The pixel difference histogram of the stego-image $\tilde{h}(d)$ will be approximated by [14]:

$$\tilde{h}(0) \approx (1-\alpha)\times h(0) + \frac{\alpha \times r_0}{w_0} \tag{3}$$

$$\tilde{h}(d) \approx (1-\alpha)\times h(d) + \frac{\alpha}{w_0} \times \sum_{j=0}^{u_0} h(j), 0 < d \le u_0 \tag{4}$$

$$\tilde{h}(d) \approx (1-\alpha)\times h(d) + \frac{\alpha}{w_0} \times \sum_{j=-u_0}^{-1} h(j), -u_0 \le d < 0 \tag{5}$$

$$\tilde{h}(d) \approx \begin{cases} (1-\alpha)\times h(d) + \dfrac{\alpha \times r_i}{w_i}, l_i \le d \le u_i \\ (1-\alpha)\times h(d) + \dfrac{\alpha \times r_{-i}}{w_i}, -u_i \le d \le -l_i \end{cases}, i > 0 \tag{6}$$

$$\alpha = \frac{no\ of\ blocks\ contains\ secret\ data}{Total\ number\ of\ blocks} \tag{7}$$

A gap will appear between $\tilde{h}(d)$ and $\tilde{h}(d+1)$ when their differences belong to two different ranges, because the difference between r_i / w_i and r_{i+1} / w_{i+1} is greater than the difference between $h(d)$ and $h(d+1)$ [14].

B. Chi-Square Attacks

The chi-square (χ^2) test is another approach that can be used to determine whether the statistical properties of an image are changed due to altering the least significant bits (LSBs) of the image pixels. Unlike the message to be hidden, the LSBs of the image pixels are not random; the backgrounds of the majority of images contain comparable LSBs. The embedding data will affect the histogram of grayscale frequencies in a particular way. Westfeld and Pfitzmann used the chi-square test to determine whether the distribution that shows distortion from embedding secret data matches the frequency distribution in an image [9].

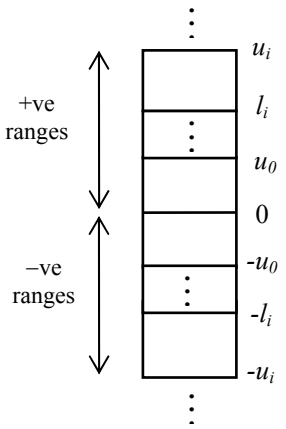
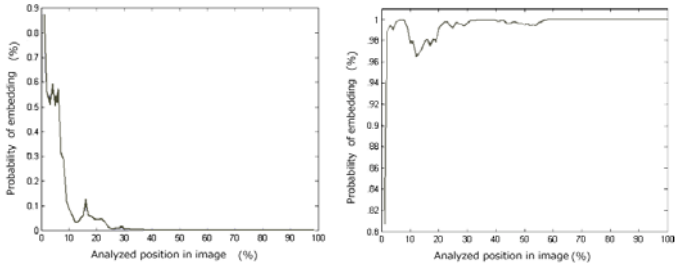


Figure 3. Ranges and their boundaries.



(a) (b)

Figure 4. Applying chi-square steganalysis on the substitute image of Lena image, (a) Before embedding and (b) After full embedding [21].

The idea of the chi-square attack is to compare the theoretically expected frequency distribution in stego-image with some possibility of error caused by the carrier medium. Sabeti *et al*. [21] utilized chi-square steganalysis to identify the existence of data embedded by PVD or by its enhanced version PVD+LSB [6], [20]. They generate a substitute image which is

created from the pixel-pair difference vector of the stego-image. Then, they applied chi-square steganalysis on the substitute image to detect the presence of embedded data. Fig. 4 shows a typical example of applying the chi-square steganalysis on the substitute image of Lena [21].

C. Universal Detectors

A more general class of steganalysis methods is known as universal detectors or blind detectors. This work is pioneered by Farid [22]. There are two main steps. In the first step, a set of discriminating features are extracted from the image by capturing statistical changes introduced by the embedding process. Then, the classification step where the extracted features are used as inputs to a suitable classifier which may have a single output to indicate whether the image contains embedded data or not. The classifier output can be a single binary value or a real value approximating how likely it is a stego-image. In the later case, a threshold value is then used to binarize the output such as if the probability is greater than the threshold value, then the image contains embedded data, otherwise it is not. Typically, the classifier is constructed using a training dataset of clean-images and stego-images. Several effective classifiers, such as Fuzzy logic (FL), support vector machine (SVM), neural network (NN) have been investigated [16], [13], [22], [11]. The performance of this category is normally shown using a confusion matrix similar to the one in Table I. Other performance metrics include the detection accuracy, specificity, sensitivity, false negative (miss) and false positive (false alarm) rates, ROC (receiver operating curve) and AUC (area under the curve).

Universal detectors have some advantages such as the ability to detect different kinds of steganographic methods. So new methods can be detected using universal detectors. On the other hand, universal detectors have lower reliability comparing with targeted detectors [7]. The ability of the features to detect the presence of a steganography with minimum error on average is known as detection accuracy.

TABLE I. CONFUSION MATRIX

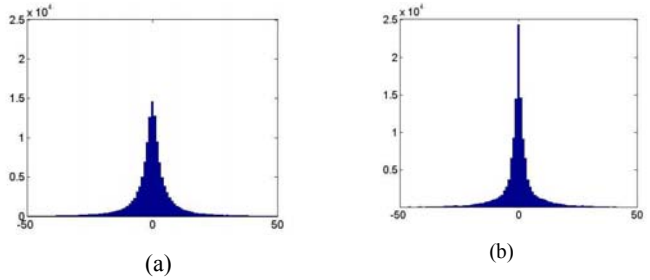
Decision/ truth	***Cover***	***Stego***
Cover	True negative	False negative (miss)
Stego	False positive (false alarm)	True positive

IV. IMPROVING THE SECURITY OF PVD

Several methods have been proposed to enhance the security of PVD. In this section we review a number of these methods were developed with this goal.

A. Improved PVD

Zaker *et al.* [23] succeeded to overcome the PVD histogram detection by preserving the Gaussian shape of the original histogram difference. Their modification on the original PVD makes the absolute difference always less or equal to the difference. This will force probability of d' to follow the distribution of d in every range r. Furthermore, their modification eliminates the problem of falling-off boundary in the original PVD, but the capacity of the embedding data is reduced. The following rules are used in their approach:



(a) (b)

Figure 5. Example of the difference histogram of Lena image (a) without embedded data, (b) with embedding data using Zaker *et al.'s* rules [23].

- For the difference d belongs to the range i, the maximum number of secret bits are selected to let the new difference $\tilde{d}$ satisfy the condition:

$$|d'| \le |d| .$$

- A slight overlap in the boundaries of applied set of ranges is made.

The first rule may decrease the capacity because it will not be possible to embed enough data in some situations. For example: When $d = l_i$ and $|d'| \le |d|$, only one bit can be embedded with restricted value of zero. On the other hand, the second rule which allowing d' to be shifted to new neighbor range will increase the embedding capacity. The effects of these rules are shown in Fig. 5.

B. Vraiable-Range PVD

In [14], another method based on PVD is proposed to increase the immunity of PVD to the histogram steganalysis. Instead of the fixed ranges of the original PVD, variable ranges for different blocks are introduced. The authors generated new ranges using a pseudo-random parameter β where $\beta \in [0, 1]$. The upper and the lower limit of the new range are varied for each block as follows:

$$l_i' = l_i + \lfloor \beta \times w_i \rfloor \quad (8)$$

$$u_i' = u_i + \lfloor \beta \times w_{i+1} \rfloor \quad (9)$$

where w_i is the width of range i. Varying the value of β will make the steps on the histogram of the pixel-pair difference disappear. After calculating the boundaries of each range, the absolute value of difference between the pixel pair is located to one range and the total number of message bits to be embedded is calculated based on the range width in a similar manner to PVD. The new difference d' is set to be the closed value to d of all the values in the same range and having residue b mod w_i. Thus d' is calculated from:

$$d' = \begin{cases} \underset{l_i' \le e \le u_i',\ \mathrm{mod}(e,w_i)=b}{\arg\min} (|e-d|), & if \quad d>0 \\ -\underset{l_i' \le e \le u_i',\ \mathrm{mod}(e,w_i)=-b}{\arg\min} (|e-d|), & if \quad d<0 \end{cases} \tag{10}$$

But when $0 \le |d'| \le u_0'$, the value of d' is given by:

$$d' = \underset{-u_0' \le e \le u_0',\ \mathrm{mod}(e,w_0)=b}{\arg\min} (|e-d|) \tag{11}$$

After that the pixel-pair is modified as in the original PVD method.

C. Modified PVD

Another approach is proposed in [24] where the embedding regions and the size of embedded message, M, are selected according to the difference between pixel-pair block in the cover image. This approach utilizes the edges efficiently by embedding secret bits in the edge regions and keeping the other smoother regions as they are. Data will be embedded using the LSB matching revisited approach. When the embedding rate is lower, this approach uses only the sharper edge regions. As the embedding rate increases, few parameters will be adjusted to release more edge regions. These parameters will be saved in the predetermined part of the image. Fig. 6 illustrates the embedding procedure.

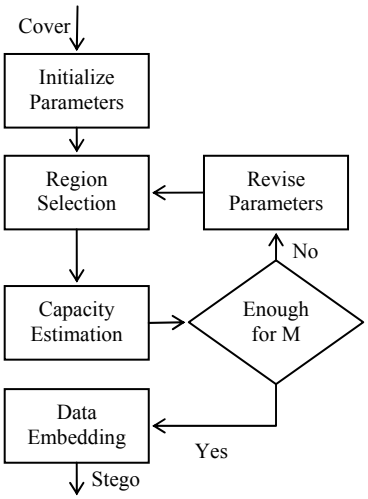


Figure 6. Embedding procedure of the modified PVD.

D. Adaptive PVD

To enhance the security of the PVD approach further, Luo *et al*. [25] use the difference of three pixels instead of two pixels. The cover-image will be partitioned into non-overlapping squares with size multiple of three. To utilize more edges in different directions and defeat the tracking of the embedding units, these squares will be rotated with 0, 90, 180 and 270 degrees using a secret key before scanning the image in zigzag manner. The image will be divided again into embedding units where each unit consists of three pixels. To embed more data in edges, a threshold value is used to estimate the strength of the edge regions. Secret bits will be embedded in the middle pixel where the number of the embedded bits will depend on the relation between the pixels in the embedding unit. Even this method has lower peak signal to noise ratio than the original PVD and variable-range PVD, it resists PVD histogram analysis. In addition, this approach was successfully passed some of the targeted attacks and universal steganalysis.

E. Modulus PVD

Wang *et al*. [26] proposed PVD with modulus function steganographic method to enhance the image quality by reducing the difference between the pixel pair before and after embedding of secret data. Instead of using the difference value, this approach modified the remainder of the pixel pair. As a result, this method increased the PSNR up to % 8.2 more than the original PVD method. In addition, the falling-off boundary problem when the pixel exceeds the value of 255 after data has been embedded is solved by using readjusting conditions. As Fig. 7 illustrates this method is more secure against the traditional pixel difference histogram analysis which reveals the existence of hidden messages embedded by PVD. The modulus PVD can be briefly described in the following steps:

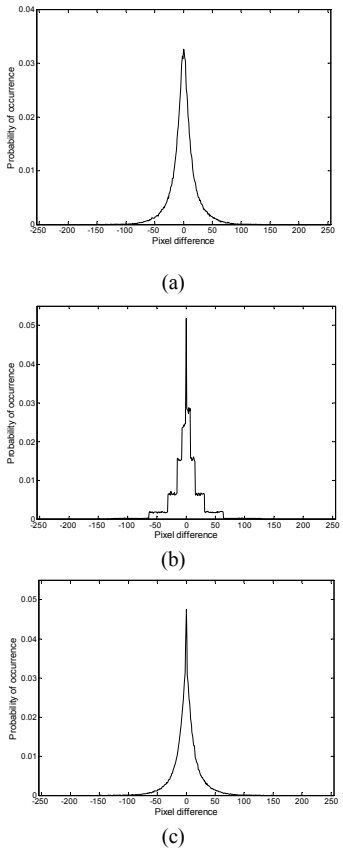


Figure 7. The pixel difference histogram of the baboon imag (a) the original image, (b) PVD and (c) modulus PVD

- Find the difference between consecutive pixels similar to the original PVD and determine the range where this difference falls.

- Compute the remainder using the following equation:

$$F_{rem(i)} = (P_i + P_{i+1}) \bmod t_i' \quad (12)$$

where $t_i' = 2^{t_i}$ and t_i is the hiding capacity of the pixel block.

- Embed n secret bits into the pixel block such that the equivalent decimal value b is equal to F_{rem}.

To keep the difference in the same range before and after the embedding, a method to alter the remainder of the pixel-pair is proposed [26].

Regardless of what has been stated about the improvements made by using PVD with a modulus function, the embedding process can still cause a number of artifacts, such as abnormal increases and fluctuations in the PVD histogram, which can be used to reveal the existence of hidden data [27]. An attack on the modulus PVD is proposed in [27]-[28] using three steganalytic measures and support vector machine. In order to enhance the security further of the modulus PVD, a turnover policy with a novel adjusting process is proposed in [29] to prevent abnormal increases in the histogram values and remove fluctuations at the border of the various ranges in the PVD histogram.

Another enhancement is presented in [30] to make use of both the horizontal and vertical directions in the image. The cover image is divided into 2×2 non-overlapping blocks. Then a modulus function is applied in the horizontal direction and a simple PVD is applied in vertical direction. This method solves the falling-off-boundary problem, increases the embedding capacity and quality of the stego-image, and can resist common attacks [30].

V. Conclusion

In this paper, we discussed various steganalytic methods that their application on the pixel-value differencing (PVD) method. We also surveyed a number of improvements that are proposed in the literature to make PVD more robust to popular attacks such as visual attack, histogram analysis, chi-square attack and universal detectors. As future work, we plan to perform experiments to compare and benchmark different methods against various attacks.

Acknowledgment

The authors would like to thank King Fahd University of Petroleum and Minerals (KFUPM), Saudi Arabia, and Hadramout Est., Yemen, for support and providing computing facilities during this work.

References

[1] Y. R. Park, H. H. Kang, S. U. Shin, and K. R. Kwon, "An image steganography using pixel characteristics," Computational Intelligence and Security, 2005, pp. 581–588.
[2] C. Yang and C. Y. Weng, "A steganographic method for digital images by multi-pixel differencing," in Proc. of International Computer Symposium, Taipei, Taiwan, ROC, 2006.
[3] B. Li, J. He, J. Huang, and Y. Q. Shi, "A survey on image steganography and steganalysis," Journal of Information Hiding and Multimedia Signal Processing, vol. 2, no. 2, 2011, pp. 142–172.
[4] D. C. Wu and W. H. Tsai, "A steganographic method for images by pixel-value differencing," Pattern Recognition Letters, vol. 24, no. 9-10, pp. 1613–1626, 2003.
[5] C. K. Chan and L. Cheng, "Hiding data in images by simple LSB substitution," Pattern Recognition, vol. 37, no. 3, 2004, pp. 469–474.
[6] H. C. Wu, N. I. Wu, C. S. Tsai, and M. S. Hwang, "Image steganographic scheme based on pixel-value differencing and LSB replacement methods," IEE Proceedings on Vision, Image and Signal Processing, vol. 152, 2005, pp. 611–615.
[7] R. Chandramouli, M. Kharrazi, and N. Memon, "Image steganography and steganalysis: Concepts and practice," Digital Watermarking, 2004, pp. 204–211.
[8] J. Fridrich and M. Goljan, "Practical steganalysis of digital images – State of the art," in Proc. of SPIE, 2002.
[9] A. Westfeld and A. Pfitzmann, "Attacks on steganographic systems," in Information Hiding, 2000, pp. 61–76.
[10] J. Fridrich, M. Goljan, and R. Du, "Reliable detection of LSB steganography in grayscale and color images," in proc. ACM Workshop on Multimedia and Security, 2001, pp. 27–30.
[11] Y. Q. Shi, G. Xuan, D. Zou, J. Gao, C. Yang, Z. Zhang, P. Chai, W. Chen, C. Chen, "Image steganalysis based on moments of characteristic functions using wavelet decomposition, prediction-error image, and neural network," in Proc. IEEE International Conference on Multimedia and Expo, ICME, 2005.
[12] S. Cho, B.-H. Cha, J. Wang, and C.-C. J. Kuo, "Performance study on block-based image steganalysis," in Proc. IEEE International Symposium on Circuits and Systems (ISCAS), 2011, pp. 2649 -2652.
[13] D.-C. Lou, C.-L. Lin, and C.-L. Liu, "Universal steganalysis scheme using support vector machines," Opt. Eng. 46, 2007.
[14] X. Zhang and S. Wang, "Vulnerability of pixel-value differencing steganography to histogram analysis and modification for enhanced security," Pattern Recognition Letters, vol. 25, no. 3, 2004 pp. 331–339.
[15] K. C. Chang, C. P. Chang, P. S. Huang, and T. M. Tu, "A novel image steganographic method using tri-way pixel-value differencing," Journal of multimedia, vol. 3, no. 2, 2008, pp. 37–44.
[16] Q. Liu and A. H. Sung, "Feature mining and nuero-fuzzy inference system for steganalysis of LSB matching steganography in grayscale images," in Proceedings of the 20th international joint conference on Artifical intelligence, 2007, pp. 2808–2813.
[17] T. Pevný and J. Fridrich, "Merging markov and DCT features for multi-class JPEG steganalysis," in Proceedings SPIE, Electronic Imaging, Security, Steganography, and Watermarking of Multimedia Contents IX, San Jose, CA, vol. 6505, 2007.
[18] E. Zheng, X. Ping, T. Zhang, G Xiong, "Steganalysis of LSB matching based on local variance histogram," in Proc. 17th IEEE International Conference on Image Processing (ICIP'10), 2010.
[19] A. D. Ker, "Steganalysis of LSB matching in grayscale images," IEEE Signal Processing Letters, vol. 12, no. 6, 2005, pp. 441-444.
[20] C. H. Yang, S. J. Wang, and C. Y. Weng, "Analyses of pixel-value-differencing schemes with LSB replacement in stegonagraphy," in Proc. Third International Conference on Intelligent Information Hiding and Multimedia Signal Processing, IIHMSP, 2007.
[21] V. Sabeti, S. Samavi, M. Mahdavi, and S. Shirani, "Steganalysis of Pixel-Value Differencing Steganographic Method," in IEEE Pacific Rim Conference on Communications, Computers and Signal Processing, 2007. PacRim 2007, 2007, pp. 292-295.
[22] S. Lyu and H. Farid, "Detecting hidden messages using higher-order statistics and support vector machines," Information Hiding, 2003, pp. 340–354.
[23] N. Zaker, A. Hamzeh, S. D. Katebi, and S. Samavi, "Improving security of pixel value differencing steganographic method," in Proc. 3rd International Conference on New Technologies, Mobility and Security (NTMS), 2009.
[24] W. Luo, F. Huang, and J. Huang, "Edge adaptive image steganography based on LSB matching revisited," IEEE Transactions on Information Forensics and Security, vol. 5, no. 2, 2010, pp. 201–214.
[25] W. Luo, F. Huang, and J. Huang, "A more secure steganography based on adaptive pixel-value differencing scheme," Multimedia Tools and Applications, vol. 52, no. 2, 2011, pp. 407–430.
[26] C. M. Wang, N. I. Wu, C. S. Tsai, and M. S. Hwang, "A high quality steganographic method with pixel-value differencing and modulus function," Journal of Systems and Software, vol. 81, no. 1, 2008, pp. 150–158.
[27] J.-C. Joo, K.-S. Kim, H.-K. Lee and H.-Y. Lee, "Histogram estimation-scheme-based steganalysis defeating the steganography using pixel-value differencing and modulus function," Opt. Eng. 49, 077001, 2010.
[28] J.-C. Joo, H.-Y. Lee, C. N. Bui1, W.-Y. Yoo, and H.-K. Lee, "Steganalytic measures for the steganography using pixel-value differencing and modulus function," LNCS 5353, 2008, pp. 476–485.
[29] J.-C. Joo, H.-Y. Lee, and H.-K. Lee, "Improved steganographic method preserving pixel-value differencing histogram with modulus function," EURASIP Journal on Advances in Signal Processin, 2010.
[30] F. Pan, Jun Li, Xiaoyuan Yang, "Image steganography method based on PVD and modulus function," in Proc. International Conference on Electronics, Communications and Control (ICECC), 2011.