International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 1 Issue 8, October - 2012

A Combined Image Steganography Technique Based on Edge Concept & Dynamic LSB

Marghny H. Mohamed[1], Naziha M. AL-Aidroos[2], and Mohamed A. Bamatraf[3]
[1]*Department of Computer Science, Faculty of Computers and Information, Asyut University, Egypt.*
[2]*Department of Computer Engineering, Faculty of Engineering and Petroleum, Hadhramout University, Yemen.*
[3]*Department of Computer Science, Faculty of Science, Hadhramout University, Yemen.*

Abstract

Steganography is the art and science of hiding data. The secret message is hidden in such a way that the observer cannot detect any changes in the original image. In this paper, we proposed an efficient steganographic scheme to hide data over gray scale images. This scheme is based on the property of human eye, which is more sensitive to the change in the smooth area than the edge area using pixel value difference, beside employing the LSB substitution technique as a fundamental stage. The experimental results over greyscale images showed the ability of embedding high data capacity with preserving stego image quality. Efficiency of the model is evaluated using two metrics, the PSNR value as one of the evaluation metrics, and the data payload (capacity) as the second. Moreover, based on that the secret message is replaced with dynamic LSBs, our scheme can effectively resist several image steganalysis techniques.

IJERT

Keywords: *Data Hiding; LSB Substitution; Pixel Value Difference; Steganography.*

1. Introduction

In recent years, enormous research efforts have been invested in the development of digital image steganographic techniques. Image steganography is a secret communication technique used to transmit secret messages that have been embedded into an image. In image steganography, the original image and the embedded image are called the cover image and the stego image, respectively. The sender hides the secret message in a cover image that has no meaning, and then transmits the stego image to the receiver through a public channel. In the transmission process, the public channel may be intentionally monitored by some opponent who tries to prevent the message from being successfully sent and received. The opponent may randomly attack the stego image if he/she doubts the stego image carries any secret message because the appearance of the stego image shows obvious artifacts of hiding effect (Liao et al., 2007). For this reason, an ideal steganography scheme, to keep the stego image from drawing attention from the opponent, should maintain an imperceptible stego image quality. That is to say, if there are more similarities between the cover image and the stego image, it will be harder for an attacker to find out that the stego image has important secret data hidden inside it (Wu and Hwang, 2007). This way, the secret data is more likely to travel from the sender to the receiver safe and sound.

There are many ways to hide information in images. Any text, image, or anything that can be embedded in a bit stream can be hidden in an image. Image steganography has come quite far in recent years with the development of fast, powerful graphical computers. An image in a computer is an array of numbers that represent light intensities at various points (pixels). These pixels make up the image's raster data. Digital images are stored in either 24-bit (true color images) or 8-bit per pixel files. Hence 8-bit color images, like GIF files, can be used to hide information. Here, each pixel is represented as a single byte, and the pixel's value is between 0 and 255. Grey scale images are preferred because the shades are changed very gradually between palette entries. This increases the image's ability to hide information (Franz et al., 1996).

Image steganographic techniques can be divided into two groups (Wu and Tsai, 2003): the Spatial Domain technique group, and the Transform Domain technique group. The Spatial domain technique embeds information in the intensity of the pixels directly, while the Transform domain technique embeds information in frequency domain of previously transformed image. Our proposed scheme is a kind of the spatial domain techniques.

The most well known techniques to data hiding in images are least significant bit (LSB) substitution, and masking & filtering techniques. LSB is a simple approach to embedding information in an image. But image manipulation can destroy the hidden information in this image.

The steganography domain is growing up very quickly. A lot of mathematical papers and practical trials are published every day. Some of these papers are used LSB method in an attempt to get a new data hiding techniques or to develop an existing techniques, such as: (Battisti et al., 2006; Chan and Cheng, 2004; Chang et al., 2003; Huang et al., 2011; Kekre et al., 2008; Ker, 2005; Khalaf and Sulaiman, 2011; Mohamed et al., 2011; Wang et al., 2001).

The LSB-based methods, directly embed the secret data into the spatial domain in an unreasonable way without taking into consideration the difference in hiding capacity between edge and smooth areas. In general, the alteration tolerance of an edge area is higher than that of a smooth area. That is to say, an edge area can conceal more secret data than a smooth area. While human perception is less sensitive to subtle changes in edge areas of a pixel, it is more sensitive to change in the smooth areas. So, new embedding techniques using the advantage of edge detection technique are introduced, such as, (Chang and Tseng, 2004; Chen and Wu, 2009; Li et al., 2006) which evaluate the correlation between neighboring pixels to determine whether a pixel is located in an edge area or a smooth area, and how many bits should be embedded in that pixel accordingly, using method named "Side Match". Another technique proposed by Wu and Tsai (2003), using the pixel-value differencing (PVD) method to distinguish edge and smooth areas. The PVD technique can embed more data in the edge area which guarantees high imperceptibility. Based on this technique, several papers are suggested in order to provide data hiding methods achieve high embedding capacity as possible. From these papers, (Chang et al., 2008; Chen et al., 2010; Padmaa and Venkataramani, 2010; Wang et al., 2008; Yang and Weng, 2006; Zhang et al., 2009).

Because the PVD method does not utilize the smooth area to hide large amount of secret data, the capacity is still low. In order to achieve higher capacity, another researchers used a combination of PVD and LSB. These techniques are based on the idea of using PVD when the difference between a pair of pixels is large (edge area), and using LSB method when the difference is small (smooth area), such as, (Al-Sadi and El-Alfy, 2011; Kim et al., 2008; Li et al., 2006; Liao et al., 2011; Wu et al., 2005; Yang et al., 2007; Yang et al., 2008).

The proposed approach introduces a new hybrid method integrating both a dynamic classical LSB data hiding technique and the pixel value difference technique. Our primary target here is to increase the capacity of embedded data without much distortion, by using the difference value between pixels adjacent to the target pixel. The difference value is used to decide which pixels belong to an edge area and which are not. To embed the secret data bits within the spatial domain of these pixels by comparing the secret data bits with the similar corresponding pixel bits of the cover image pixels in the smooth area. We experiment on various standard images to evaluate the efficiency of the proposed method, and as a result, our method is able to produce stego image more similar to the original image.

The paper is organized as follows; briefly background about the techniques used in the proposed method is introduced in Section 2. Our scheme is presented in Section 3. The experimental results with some analyses and discussions are shown in Section 4. Finally, the conclusions are provided in Section 5.

2. Background

2.1. Least Significant Bit Hiding (LSB) Scheme:

Least significant bit (LSB) insertion is a common and simple approach to embed information in a cover file: it overwrites the LSB of a pixel with an M's bit. Unfortunately, modifying the cover image changes its statistical properties, so eavesdroppers can detect the distortions in the resulting stego image's statistical properties. The general operation of data hiding by using a simple LSB substitution method is described in this section.

Many steganogrphic methods embed a large amount of the secret information in the first k LSB's of the cover image pixels. Because of the imperfect sensibility of the human visual system, the existence of the embedded secret information can be imperceptible.

A digital image I can be represented by a two dimensional, with size (N= *width* × *height*). Each image is composed of finite elements each of which has a definite location and amplitude. These elements are referred to as image pixels, $I = \{P_1,..,P_N\}$, where every pixel P_i in the greyscale image consists of 8 bits: $|P_i| = 8$, $P_i = \{b_1, .., b_8\}$, where: $b_j \in \{1, 0\}$.

LSB with k embedding factor $1 \le k \le 8$, for every P_i targeted for embedding data bits replaces the set of bits: $T = t_1, ..., t_k$, keeping the rest of bits $\bar{T} = (i.e.\ t_{k+1} .. t_8)$ without effect.

The generated set of pixels $\{P'_1, ..., P'_N\}$ represents as the stego image I', where: $P'_i = \{b'_1, .., b'_8\}$, $b'_j \in \{1, 0\}$.

The following example describes how the LSB embedding happens, suppose we have the following pixels:

P_1= [11001011], P_2= [00011010], P_3= [01001100].

And the bits want to embed it in the LSBs positions are M = [010], the resulted pixels after embedding will be:

P_1= [1100101**0**], P_2= [0001101**1**], P_3= [0100110**0**].

The quality of the stego image produced by simple LSB substitution may not be acceptable. It means that the method degrades the image quality and probably attracts unauthorized attention. Once he/she notices the stego image, secret message can be easily extracted by simple LSB analysis.

The proposed method can solve these problems, even though it is based on classic LSB, yet it is totally different as the embedding process is not a uniform in terms of what pixels are selected for embedding in the cover image. This increases the complexity of the

IJERT

hidden data extraction in one hand and preserves the image quality in the other hand, by minimizing the number of the modified LSB bits.

2.2. Pixel-Value Differencing (PVD) Scheme:

The alteration of edge areas in the human visual system cannot be distinguished well, but the alteration of smooth areas can be distinguished well. That is, an edge area can hide more secret data than a smooth area. With this concept, Wu and Tasi, (2003) proposed a novel steganography technique using the pixel-value differencing (PVD) method to distinguish edge and smooth areas. This method relies on the idea that not all pixels can store the same number of bits of the secret data. Instead of inserting the secret bits directly to the end of each byte of the cover image (which is the way in which LSB works), they determine the number of bits to be embedded based on the differences between pairs of adjacent pixels. This allows the method to embed more data in the edge area of the cover image without too much reduction in the stego image quality.

An edge is characterized by significant dissimilarity in gray levels being used to indicate the boundary between two regions in an image fragment. Edge detection is a significant area of the image processing and machine vision due to the fact that edges are considered to be the important features for analyzing the most essential information contained in images (Chen et al., 2010).

In the proposed method, we employed the PVD idea to generate the edges of the cover image.

IJERT

3. The Proposed Scheme

In this section, the proposed scheme will be introduced in detail, it consists of two procedures, the embedding procedure and the extracting procedure.

3.1. The Embedding Procedure:

Any image I consist of set of pixels:

$$I = \{P_1,.., P_N\},$$

$$|P_i| = 8 \text{ bits},\ P_i = \{b_1, .., b_8\},\ b_j \in \{1, 0\}. \quad (1)$$

The image size is computed as:

$$N = W \times H \quad (2)$$

Where W, H are the image width and height respectively. Suppose M is the secret data bits, with length n,

$$M = \{m_1, m_2, \ldots, m_n\},\ m_i \in \{1, 0\}. \quad (3)$$

The proposed method takes the dependency advantage of pixels on its surrounding neighbors. The correlation between a pixel and its neighbors decides whether it is located in smooth area or in an edge area. The data embedding algorithm's steps are as follows.

Step 1: Obtain the edge image of the cover image I_C, by dividing the image into overlapping blocks, each block consists of 4 neighboring pixels (Figure 1).

$P_{(i,j)}$	$P_{(i,j+1)}$
$P_{(i+1,j)}$	$P_{(i+1,j+1)}$

Figure1. A target pixel and three neighboring pixels

Where the target pixel $P_{(i,j)}$ with gray value $g_{(i,j)}$, let $g_{(i,j+1)}$, $g_{(i+1,j+1)}$ and $g_{(i+1,j)}$ be the gray value of the neighboring pixels, right pixel as $P_{(i,j+1)}$, down-right pixel as $P_{(i+1,j+1)}$ and down pixel as $P_{(i+1,j)}$, respectively.

Step 2: Calculate the difference value d of each block to indicate the smooth and edge regions, then select the maximum difference among them as follows,

$$d1 = |g_{(i,j+1)} - g_{(i,j)}|,$$
$$d2 = |g_{(i+1,j)} - g_{(i,j)}|,$$
$$d3 = |g_{(i+1,j+1)} - g_{(i,j)}|,$$
$$d = max\ (d1, d2, d3). \quad (4)$$

Step 3: Using Equation (4), we decide whether the target pixel is included in an edge area, if d value is more or equal to a certain threshold, otherwise it included in a smooth area, after this step we obtain two sets of pixels, the first called edge pixels denoted as (E), and the other called smooth pixels denoted as (S).

$$E = \{E_1, .., E_e\},$$
$$S = \{S_1, .., S_s\}. \quad (5)$$

Where e, s are the edge pixels size, and smooth pixels size respectively, $N = e + s$.

Step 4: Initially set the secret data bits index (k) to 1. For all smooth pixel $S_i \in S$, where $1 \le i \le N$, as a first step in the embedding process, the first 2 LSB bits are reserved immediately for embedding, as follows,

$$S_i[1] = M_k,\ k = k + 1,$$
$$S_i[2] = M_k,\ k = k + 1. \quad (6)$$

Where $S_i[1]$, $S_i[2]$ are the first and second bits of the pixel S_i respectively. Then, the embedding continues as follows,

$$for\ (\ j = 3\ to\ 8)$$
$$if\ (S_i[j] == M_k)$$
$$embed\ in\ S_i[j],$$
$$k = k + 1, j = j + 1,$$
$$Else$$
$$save\ j\ in\ 3LSB\ of\ E_i,$$
$$go\ to\ next\ pixel\ S_{i+1}. \quad (7)$$

Where j is the bit index of the smooth pixel S_i, the metadata about last embedding in the pixel S_i is saved in the 3LSB's of the corresponding edge pixel E_i.

Step 5: After embedding all the secret data bits, the stego image I' is generating.

The block diagram of the entire embedding procedure is represented in Figure 2.

3.2. The Extracting Procedure:

To recover the original secret data at the receiving side, the original image I must be known to determine the original edge (E) and smooth (S) areas before embedding, then the following steps are done.

Step 1: Find the edge and smooth areas of the cover image, as the same way described in the previous section.

Step 2: Using the position of the original smooth and edge areas we can determine which pixels belong to the edge area (E'), and which are belong to the smooth area (S') in the stego image (I'), easily.

Step 3: For each smooth pixel $S'_i \in S'$ in the stego image, extract the metadata which defines the position of the last occurrence of embedded bits on it from the 3LSB of the corresponding edge pixel E'_i.

$$\begin{aligned} &set\ metadata = 3lsb's\ of\ E'_i, \\ &for\ (\ j = 1\ to\ metadata\ ) \\ &\quad M'_k = S'_i[j], \\ &\quad k = k+1, \\ &\quad j = j+1. \end{aligned} \tag{8}$$

Where M' is the extracted data bits.

Step 4: At this stage, the retrieving algorithm finishes and the embedded data has been retrieved completely.

The extracting procedure steps are described in Figure 3.

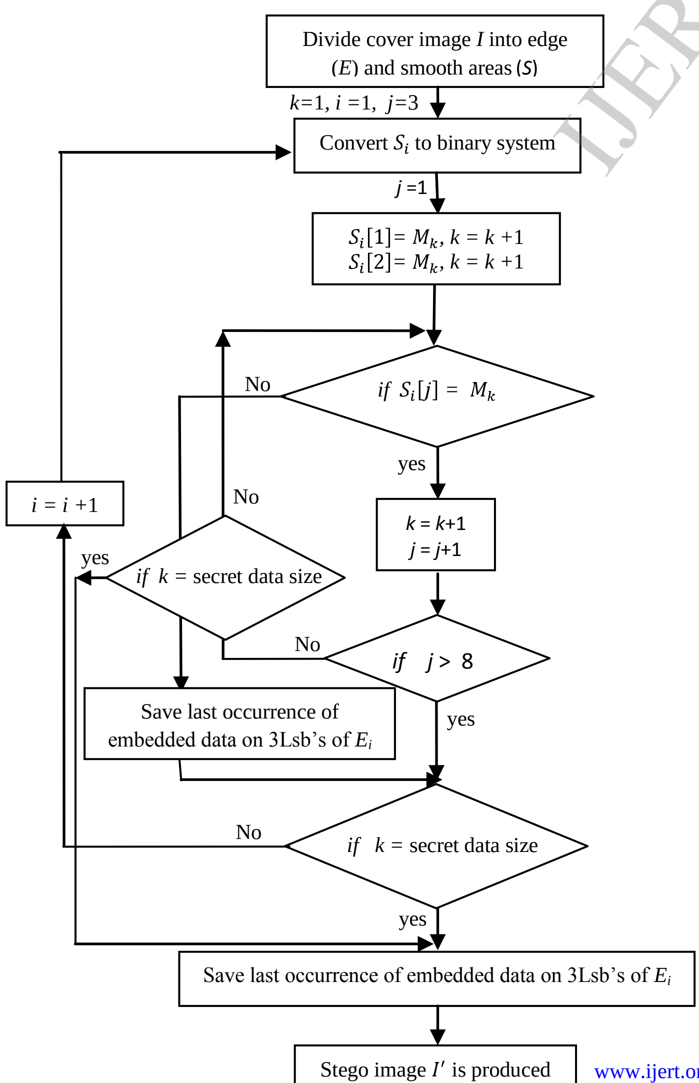


Figure 2. A Block Diagram of Embedding Steps

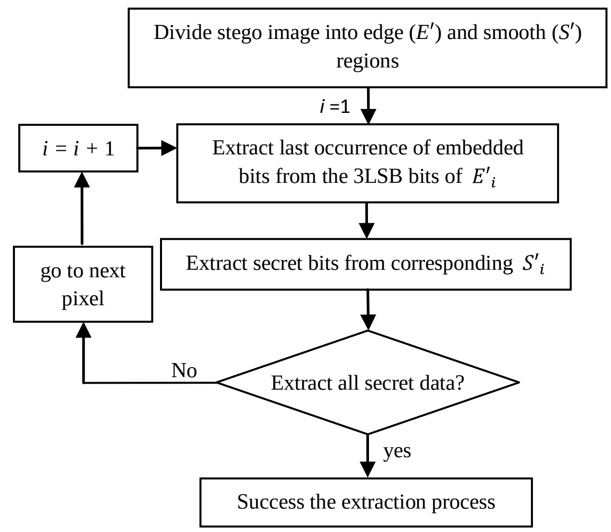


Figure 3. A Block Diagram of Extracting Steps

4. Experimental Results

The experimental results presented in this section demonstrate the performance of our proposed scheme. To conduct our experiments, we used four 128×128 standard grayscale images, ''Baboon'', ''Lena'', ''Pepper'' and ''Cameraman''. These images are shown in Figure 4.

(a) Baboon (b) Lena

(c) Pepper (d) Cameraman

Figure 4. Four 128×128 grayscale images

IJERT

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 1 Issue 8, October - 2012

A series of pseudo random binary numbers are used as the secret data to be embedded into the cover images.

The performance of the proposed scheme is considered from two viewpoints. The visual quality of the stego image and the data payload (capacity).
The peak signal-to-noise ratio (PSNR) was utilized to evaluate the stego image quality. $PSNR$ is often expressed on a logarithmic scale in decibels (dB), it is applied on the stego and the cover images. It is defined as:

$$PSNR\ = 10\ \times\ log_{10}\frac{255^2}{MSE}\ (dB). \qquad (9)$$

Where MSE is the mean square error between the cover and stego images. For a cover image whose width and height are W and H, MSE is defined as:

$$MSE\ =\ \frac{1}{W\times H}\sum_{i=1}^{W}\sum_{j=1}^{H}(I_{ij}-I'_{ij})^2. \qquad (10)$$

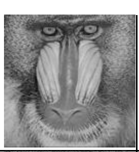
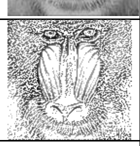
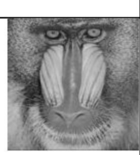
Where I_{ij} and I'_{ij} are the pixel values of the cover and stego images, respectively.

As a performance measurement for embedding capacity, the average number of bits embedded into each pixel is calculated as:

$$\text{Capcity} = \left(\frac{\text{Total number of bits embedded into image}}{\text{Total number of pixels on image}}\right)\ (\text{bits/pixel}). \qquad (11)$$

Table 1 shows the quality of the stego image and the number of edge pixels in each of them, which are generated by the proposed scheme.

Table 1. The stego image quality generated by our scheme

Greyscale images				
Edge images				
No. of edge pixels	8676 Pixel	2803 Pixel	7401 Pixel	7097 Pixel
Stego images				

The average of PSNR, and capacity is calculated for 6 runs with different secret data bits length, the results of the proposed method for the four grayscale images are shown in Table 2.

Table 2: Experimental results for the proposed method

Cover Images	Avg. Capacity (Bits/pixel)	Avg. PSNR (dB)
Baboon	1.56	41.74
Lena	1.47	41.10
Pepper	1.45	41.40
Cameraman	1.54	39.00

Note that, a large PSNR value means that the stego image is most similar to the original image and vice versa. Generally, if the PSNR value is larger than 30 dB, then the distortion on the stego image is hard to be detected by human eyes (Al-Sadi and El-Alfy, 2011; Chou et al., 2008). Our experimental results show that the proposed method can embed a large amount of secret data (reaching 1.56 *bpp*) while keeping a high visual quality (above 30 *dB*), as shown in Table 2.

The mentioned metrics could clearly prove the applicability of the proposed model. The selected metrics produced a set of results to be discussed and justified based on behaviour of the algorithm.

IJERT

The proposed scheme has several advantages. Firstly, in the smooth area the embedding is done based on similarity between the data secret bits and the smooth pixels bits, except the first two least significant bits, this means, the proposed method does not cause any change in the pixel bits of smooth area almost, the embedding process in the 2 LSB bits can be considered as the worst case of the model capacity, in case of no match is met in the remaining part of the pixel. At the same time in case of further match(s) is/are found capacity will be improved without causing any effect in image quality as the similarity between bits is the embedding criteria for the remaining bits. Otherwise, in the edge area we just embed the metadata about last position we embed in the corresponding smooth pixels, only in the 3LSB's of it, this indicates that our method does not change the image statistics to a noticeable extent. Secondly, mapping capacity is variable and can be decided depending on the number of pixels in the smooth and edge area, and the amount of data to be mapped. Also, it depends on how much the secret data is similar to the image pixels bits by searching for the higher similarity rather that blindly embedding the data. All of that reduce the changes in the original cover image due to embedding secret data and makes it hard to be recognized by human eyes. Finally, the proposed scheme requires much less computations because its steps are so simple, easy and don't need complex computations such as some techniques which transform image in the frequency domain.

5. Conclusion

In this paper, a new method to embed secret data into the cover image by a dynamic LSB substitution method based on divide the cover image into edge and smooth areas, using the pixel-value differencing to improve the image quality of the stego image and increase the embedding capacity of the cover image. The main idea here is to utilize the similarity in the smooth area, unlike simple techniques that use the edge area. Because the feature that the human eye can't detect the changes on edge area easily, we embed only 3LSB's bits on its pixels to save the metadata about the last occurrence of the embedded bits in the corresponding smooth pixels. This advantages provided by the proposed scheme benefit in generating a higher quality stego images under the HVS more similar to the original image. Experimental results confirm that the proposed scheme is successful in obtaining a stego image of satisfactory quality. Moreover, it can resist steganalysis systems which are based on statistical analysis.

References

[1] Al-Sadi, A. and El-Alfy, E., (2011) "An Adaptive Steganographic Method for Color Images Based on LSB Substitution and Pixel Value Differencing", *ACC 2011, Part II, CCIS 191*, pp. 535–544.

[2] Battisti, F.; Carli, M.; Neri, A. and Egiaziarian, K., (2006) "A Generalized Fibonacci LSB Data Hiding Technique", *3rd International Conference on Computers and Devices for Communication (CODEC-06)*, Institute of Radio Physics and Electronics, University of Calcutta.

[3] Chan, C. and Cheng, L., (2004) "Hiding Data in Images by Simple LSB Substitution", *Pattern Recognition*, vol. 37, pp. 469–474.

[4] Chang, C.; Hsiao, J. and Chan, C., (2003) "Finding Optimal Least-Significant-Bit Substitution in Image Hiding by Dynamic Programming Strategy", *Pattern Recognition*, vol. 36, pp. 1583- 1595.

[5] Chang, C. and Tseng, H., (2004) "A Steganographic method for digital images using side match", *Pattern Recognition Letters*, vol. 25, pp. 1431-1437.

[6] Chang, K.; Chang, C.; Huang, P. and Tu, T., (2008) "A Novel Image Steganographic Method Using Tri-way Pixel-Value Differencing", *Journal of Multimedia*, vol. 3, no. 2, pp. 37-44.

[7] Chen, P. and Wu, W., (2009) "A Modified Side Match Scheme for Image Steganography", *International Journal of Applied Science and Engineering*, vol. 7, no.1, pp: 53-60.

[8] Chen, W.; Chang, C. and Le, T., (2010) "High payload steganography mechanism using hybrid edge detector", *Expert Systems with Applications*, vol. 37, pp 3292-3301.

[9] Chou, Y.; Chang, C. and Li, K., (2008) "A Large Payload Data Embedding Technique for Color Images", *Fundamenta Informaticae*, vol. 88, pp. 47-61.

[10] Franz, E.; Jerichow, A.; Moller, S.; Pftizmann, A. and Stierland, I., (1996) "Computer Based Steganography", *in Information Hiding, Springer Lucture Notes in Computer Science*, vol. 1174, pp 7-21.

[11] Huang, W.; Zhao, Y. and Ni, R., (2011) "Block-Based Adaptive Image Steganography Using LSB Matching Revisited", *Journal of Electronic Science and Technology*, vol. 9, no. 4, pp. 291-296.

[12] Kekre, H.; Athawale, A. and Halarnkar, P., (2008) "Increased Capacity of Information Hiding in LSB's Method for Text and Image", *International Journal of Electrical, Computer, and Systems Engineering*, vol. 2, no.4, pp. 246-249.

[13] Ker, A., (2005) "Improved Detection of LSB Steganography in Grayscale Images", *in: J. Fridrich (Ed.), Information Hiding, Sixth International Workshop*, Lecture Notes in Computer Science, Springer, New York, vol. 3200, pp. 97–115.

[14] Khalaf, E. and Sulaiman, N., (2011) "A Robust Data Hiding Technique based on LSB Matching", *World Academy of Science, Engineering and Technology*, vol. 58, pp. 117-121.

[15] Kim, K.; Jung, K. and Yoo, Y., (2008) "A High Capacity Data Hiding Method using PVD and LSB", *International Conference on Computer Science and Software Engineering*, pp. 876-879.

[16] Li, S.; Leung, K.; Cheng, L. and Chan, C., (2006) "Performance Evaluation of a Steganographic Method for Digital Images Using Side Match", *First International Conference on Innovative Computing, Information and Control - Volume III (ICICIC'06)*, vol. 3, pp. 54-57.

[17] Li, S.; Leung, K.; Cheng, L. and Chan, C., (2006) "Data Hiding in Images by Adaptive LSB Substitution Based on the Pixel-Value Differencing", *Proceedings of the First International Conference on Innovative Computing, Information and Control (ICICIC'06)*.

[18] Liao, Z.; Huang, Y. and Li, C., (2007) "Research on data hiding capacity", *International Journal of Network Security*, vol. 5, pp140-144.

[19] Liao, X.; Wen, Q. and Zhang, J., (2011) "A steganographic method for digital images with four-pixel differencing and modified LSB substitution", *J. Vis. Commun. Image R.*, vol. 22, pp. 1-8.

[20] Mohamed, M.; Al-Afari, F. and Bamatraf, M., (2011) "Data Hiding by LSB Substitution Using Genetic Optimal Key-Permutation", *International Arab Journal of e-Technology*, vol. 2, pp. 11-17.

[21] Padmaa, M. and Venkataramani, Y., (2010) "Zig-Zag PVD - A Nontraditional Approach", *International Journal of Computer Applications*, vol.5, pp. 5-10.

[22] Wang, C.; Wu, N.; Tsai, C. and Hwang, M., (2008) "A high quality steganographic method with pixel-value differencing and modulus function", *The Journal of Systems and Software*, vol. 81, pp. 150-158.

[23] Wang, R.; Lin, C. and Lin, J., (2001) "Image Hiding by Optimal LSB Substitution and Genetic Algorithm", *Pattern Recognition*, vol. 34, no. 3, pp. 671-683.

[24] Wu, D. and Tsai, W., (2003) "A steganographic method for images by pixel-value differencing", *Pattern Recognition Letters*, vol. 24, pp. 1613-1626.

IJERT

[25] Wu, H.; Tsai, C.; Wu, N. and Hwang, M., (2005) "Image Steganographic Scheme Based on Pixel-Value Differencing and LSB Replacement Methods", *In: IEE Proceedings Vision, Image and Signal Processing*, vol. 152, pp. 611-615.

[26] Wu, N. and Hwang, M., (2007) "Data hiding: current status and key issues", *International Journal of Network Security*, vol. 4, pp 1-9.

[27] Yang, C. and Weng, C., (2006) "A Steganographic Method for Digital Images by Multi-Pixel Differencing", *in: Proceedings of International Computer Symposium*, Taipei, Taiwan, R.O.C., pp. 831-836.

[28] Yang, C.; Wang, S. and Weng, C., (2007) "Analyses of Pixel-Value-Differencing Schemes with LSB Replacement in Stegonagraphy", *In: Proc. of the 3rd Int. Conf. on Intelligent Information Hiding and Multimedia Signal Processing*, vol. 1, pp. 445-448.

[29] Yang, C.; Weng, C.; Wang, S. and Sun, H., (2008) "Adaptive Data Hiding in Edge Areas of Images With Spatial LSB Domain Systems", *IEEE Transactions on Information Forensics and Security*, vol. 3, no. 3, pp. 488-497.

[30] Zhang, H.; Geng, G. and Xiong, C., (2009) "Image Steganography using Pixel-Value Differencing", *Second International Symposium on Electronic Commerce and Security*, pp 109-112.

IJERT