ICIT 2015 The 7th International Conference on Information Technology
doi:10.15849/icit.2015.0045

On The Improvement of the Tri-Way Pixel Value Differencing Steganography Algorithm

Nada Mahmoud Aboueata, Sara Yaqoob Al-Rasbi, Wafa Ahmed Al-Jaal, Jihad Al-Ja'am
Department of Computer Science and Engineering
Qatar University
Doha, Qatar
{nada.aboueata,sara.alrasbi,wa095710,jaam}@qu.edu.qa

***Abstract*—Steganography consists of hiding information into digital files so that they cannot be noticeable to human vision. These files include texts, images, audios, videos and protocols. In this work we consider only gray-scale images as they are the most used digital covers in steganography. The most challenging problem consists of finding the right pixels of the image to embed or hide the maximum amount of information without deteriorating its quality. Research in this area is still at the premature phase even though several steganography image-based algorithms have been proposed recently. In this paper, we study the well-known tri-way pixel value differencing algorithm (TPVD) aiming at improving its performance. We select randomly the starting pixel of the pixel-pair combinations to hide information rather than starting always with the first pixel as done in the TPVD regular behavior algorithm. Our first experiment shows that a slight improvement can be obtained with these random selections in preserving the quality of the stego-image (i.e., the image holding the information) and maximizing the amount of hidden information. We study also the encryption of the sensible information that should be embedded into the cover image using the AES algorithm. These randomness behavior along with the encryption technique render the retrieval of the hidden information very hard once the image is spotted as suspicious to be a stego-image and the hidden information are attempted to be extracted. Our second experiment shows that the encryption of the same amount of information may deteriorate the quality of the stego-image and makes it somehow perceptible for human vision and also vulnerable to stego-analysis techniques.**

Keywords—*information security; steganography; secret communication; tri-way algorithm*

I. INTRODUCTION

Exchanging information over computer networks the Internet is a challenging task due to the possible attacks that may occur during the transmission phase. Several encryption algorithms have been proposed to secure the information and made them illegible once detected by an illegal interceptor. However, encrypted information can be easily vulnerable to analysis and then decryption attempts. Another technique is proposed to secure transmitted secret or sensible information. It consists of embedding the information into a cover digital image and make it imperceptible to human vision. This technique is known as steganography and the cover image that hides the information is called a stego-image. This means of covert communication can be used in commercials and military applications. An image consists of a set of different numbers representing intensities in different areas of an image. This number-based representation constitutes a grid and the points are called pixels. Each pixel in a gray-scale image is represented by 8 bits and can have 256 different intensities. Several image-based steganography algorithms have been proposed recently aiming at maximising the amount of information to be hidden and preserve also the image quality [12,13]. However, they require major improvements [15]. The PVD algorithm is one of the popular approaches used in steganography. It consists of hiding information using the difference of two consecutive pixel-values. The stego-images obtained from the PVD algorithm and its derivatives can be easily detected by the difference histogram analysis techniques as they follow one regular direction in the embedding phase [15,16,17]. Luo et al. [15] have proposed a more secure steganography algorithm which consists of splitting the image into blocks and randomly rotating them. The resulting image is divided into units of three pixels where the middle one is selected to embed the information. Although the authors shown an improvement in the embedding phase, it can be considered as regular as it starts always with same pixel. Chang et al. [1] have proposed an efficient steganography algorithm called the tri-way pixel value differencing algorithm (TPVD). This algorithm consists of using always the first possible pixel-bit during the embedding phase of the information into the cover image. This selection is thought to preserve always the quality of the stego-image. In this paper we study the consideration of random pixels as starting pixels in the embedding of the information. We show that a slight improvement can be obtained with this technique. We study also the encryption of the information that need to be

doi:10.15849/icit.2015.0045

embedded into the cover image and we show how the quality of the stego-image is affected.

The rest of the paper is organized as follows: in section 2, we detail the TPVD algorithm. In section 3 we present our method which considers random-based combination of pixels in the embedding phase. In section 4 we show the results of our experiments and finally in section 5 we conclude the paper.

II. THE TPVD ALGORITHM

The TPVD algorithm is proposed as a significant improvement of the PVD algorithm. Both techniques split the cover image into a sequence of 2x2 blocks of pixels. They use then the difference of each pixels-pair to determine the number of bits that could be embedded. They assume human vision can easily observe changes in gray values of smooth area of a stego-image, but they are unable to notice relatively larger changes at the edges areas [1,15]. In order to determine the smoothness properties of the stego-image, the difference between every pixel-pair is calculated. To find the numbers of bits that should be embedded in each pixels pair, a vector ranged from 0 to 255 is built. The range of gray values is divided into smaller ranges as follows: [0-7, 8-15, 16-31, 32-63, 64-127, 128-255] where each region is defined by a lower bound (Li) and an upper bound (Ui). The absolute value of the difference for each pixel-pair is located into one range and the number of bits to be embedded into that pixel-pair is determined by the width of this range denoted by (Wi). This width is obtained by the following equation:

$$[Wi = Ui - Li+1]$$

while the number of bits to be embedded into that pixel-pair is calculated by [number of bits=log2(Wi)]. Ranges close to 0 represent smooth areas and thus have smaller widths. Similarly ranges close to 255 represent clear edges and thus have larger widths. The number of bits to hide is embedded as a difference between the pixels-pair and hence the pixels values are changed accordingly.

The PVD algorithm partitions the image into blocks where each one consists of two consecutive pixels in one direction (i.e., one edge). The TPVD algorithm upgrades the capacity of the information to be hidden by partitioning the image into 2x2 blocks. Each one consists of three pixel-pair in three different directional edges (i.e., horizontal, vertical, and left diagonal). Since setting larger embedding capacity can cause image distortion, an optimal approach of selecting the reference point and branch conditions are given to achieve a minimum square error (MSE) and to reduce the effect of the stego-image distortion.

A. Embedding Phase

The embedding phase of the TPVD algorithm involves the following steps:

1. Partition the gray level of the cover image into a sequence of 2x2 blocks of pixel-pair. Fig. 1 shows one block of pixels-pair.

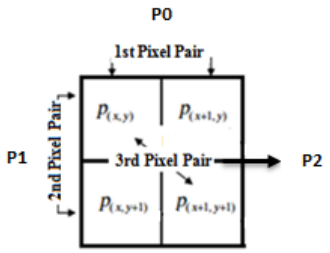


Fig.1. A 2 by 2 block of a cover image.

2. Calculate the difference for the following pixels pairs:

$$\begin{aligned} d_0 &= P_{(x+1,y)} - P_{(x,y)} \\ d_1 &= P_{(x,y+1)} - P_{(x,y)} \\ d_2 &= P_{(x+1,y+1)} - P_{(x,y)} \\ d_3 &= P_{(x,y+1)} - P_{(x+1,y+1)} \end{aligned} \quad (1)$$

3. Locate the range $[L_i,U_i]$ in the selected region for each $|d_i|$.

4. Compute then the width for each region as follows:

$$W_i = U_i - L_i + 1 \quad (2)$$

5. Compute the amount of bits (t_i) that can be embedded in each pixel-pair as follows:

$$t_i = \log_2(Wi) \quad (3)$$

6. Check whether every t_i for the pixel-pair P_0, P_1, P_2 satisfies at least one of the following branch conditions:

$$\begin{aligned} t(P_0) &\geq 5 \text{ and } t(P_1) \geq 4 \\ t(P_0) &< 5 \text{ and } t(P_2) \geq 6 \end{aligned} \quad (4)$$

These conditions are used to determine the maximum amount of bits that can be embedded into every pixel-pair without deteriorating the quality of the stego-image.

- If t_i of P_i is satisfying the branch conditions, the three pixel pairs (P_0, P_1, P_2) cannot be used for embedding information. The PVD algorithm should then be used and the pixel pairs (P_0, P_4) are selected for embedding as shown in figure 2.

Fig.2. Two Consecutive pixels blocks of PVD.

doi:10.15849/icit.2015.0045

- If t_i of P_i is not satisfying the branch conditions, the three pixel pairs (P_0, P_1, P_2) will be used for embedding.

7. Get the amount (t) bits from the information file and convert t to a decimal number (b).

8. Calculate the new difference for each pixel-pair as follows :

$$d_i' = L_i + b_i \text{ , if } d\ i \geq 0$$
$$d_i' = -1 * (L_i + b_i) \text{ , if } d\ i < 0 \quad (5)$$

9. Modify the pixels values as follows:

$$(P_n', P_{n+1}') = (P_n - \lceil m/2 \rceil , P_n + 1 + \lfloor m/2 \rfloor) \quad (6)$$
$$\text{where } m = d_i' - d_i$$

10. If the three pixel pairs (P_0, P_1, P_2) are used for embedding, we will end up with having three different values for the pixel $p_{(x,y)}$ since it is a common pixel among the pixels-pair. Therefore, we choose the optimal reference point $p_{(x,y)}$ with the minimum MSE, and then we offset the other two pixels-pair.

11. If the two pixel-pairs (P_0, P_4) are used for embedding the information, we calculate then t_i' of P_0' , P_1', P_2'. We check ift_i' is still satisfying the branch conditions. If not, we offset the values of P_4' to satisfy these conditions.

12. Construct a new 2 x 2 block and repeat the previous steps until all the information bits are embedded.

B. *Extracting Phase*

1. Partition the gray-scale image into 2 x 2 blocks of pixel pairs.
2. Calculate the difference for the pixels pairs as shown in (1).
3. Locate the range [L_i,U_i] in the designed region table for each $|di|$.
4. Compute the width for each range as in (2).
5. Compute the amount of bits (t) that can be embedded in each pixel pair as in (3).
6. Check whether the computed amount of bits (t) for P_0, P_1, P_2satisfies at least one of the branch conditions of (4). If it is satisfying then the P_0, P_1, P_2 pixel pairs are selected. Otherwise, the two independent pixel pairs P_0, P_4 are selected.
7. Subtract the lower bound Li from the |di| to obtain b. Then convert b into a binary sequence with ti bits.

III. THE PROPOSED ALGORITHM

The most challenging part of the steganography algorithms lies within choosing the appropriate pixels to embed the information. In our random-based algorithm that we denote by RTPVD we use a random factor to choose the proper combination of pixels to be used in the embedding phase. Four different combinations are used. Every combination starts with a different pixel based on a random sequence. This randomization limits the allocation of embedded information in the same direction of pixel pairs. This preserve the setgo-image quality. In addition, the information are encrypted before embedding using the well-known AES algorithm which uses 128 bits as a key and provides a high level of security. In addition, since we have a stego-key (i.e., the random generator seed) which is used to generate a random sequence for the starting pixel in every pixel-pair, it will be relatively easy to handle two keys as one (i.e., the stego key and the encryption-decryption key).

A. *Block Selection*

The embedding phase in the TPVD algorithm is performed using always the first pixel $P_{(x,y)}$ as a starting point. The most important thing is to have a combination with a common point included in each pixel pair to embed the information within the same direction. In the TPVD algorithm we have three possible pixel pairs where $P_{(x,y)}$ is included in each one of them. We can start by randomizing the pixel pairs into two different combinations as shown in figure 3. However, we found that the combination (i.e., figure 3a right side) is not useful to use and should then be discarded. In fact, the tri-way direction is based on the starting point and its associated pairs. To overcome this problem we choose four combinations as shown in figure 4. We have

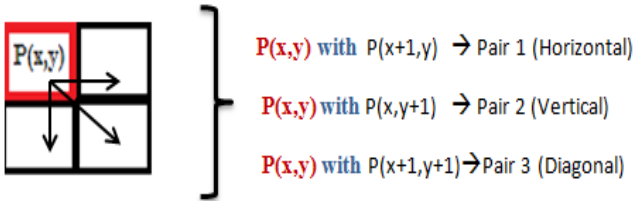


Fig.2. Pixels-pair combinations of the TPVD algorithm.

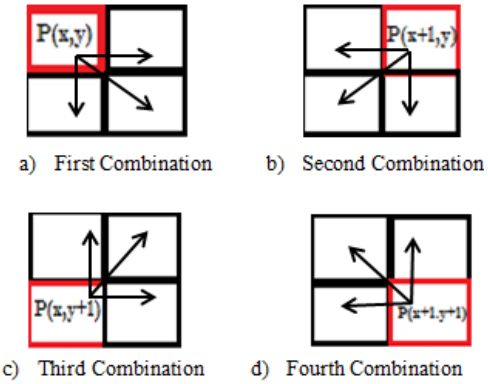


Fig. 3. All possible pixel pair combinations.

doi:10.15849/icit.2015.0045

taken into consideration that each combination should have its own starting point linked to its pairs and in all possible directions (i.e., vertical, horizontal and diagonal).

B. Embedding Phase

To embed the information, we apply the same steps used in TPVD algorithm except that we randomize between different pixel pair combinations.

1. The algorithm takes as input a digital key (i.e., an integer number or seed). This key is used to generate a sequence of pixel numbers that the algorithm uses in selecting the starting pixels of every block.
2. The information are encrypted using the AES algorithm. The same key can also be used in the encryption and the decryption phase.
3. Based on the digital key, one of the proposed four pixel pair combinations is used to embed the information.
4. The key is embedded into the first block of the stego-image. It is used in the information extraction phase.

C. Extracting Phase

1. The algorithm starts by extracting the randomisation key from the first block of stego-image. Note that in this block no secret information are hidden.
2. Use the key to generate the random list of pixel pairs combinations that have been used in the embedding phase.
3. To extract the information from the stego-image, we use the same steps of the TPVD extraction algorithm except that the extraction will be done by using the random pixel pairs combinations generated.
4. The hidden encrypted information is extracted and decrypted using the AES algorithm.

IV. EXPERIMENTAL RESULTS

We have conducted two different experiments using three gray-scale images (i.e., Lena.bmp, Barbara.bmp, Girl.bmp)


Figure 5: Lena.bmp
Fig.4: Lena.bmp

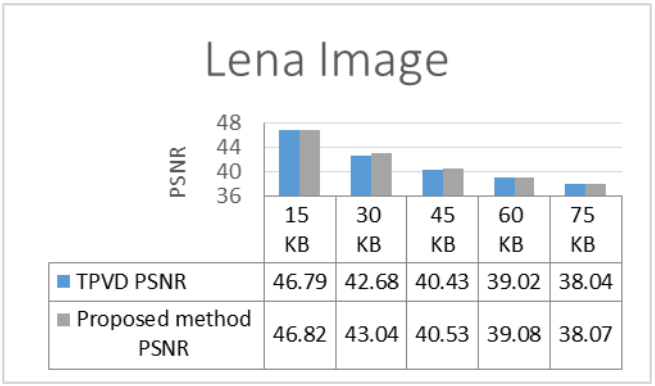

Figure 6 : Barbara.bmp
Fig.5: Barbara.bmp


Figure 6 : Girl.bmp
Fig.6: Girl.bmp

with 512x512 resolution. The experiments are performed with different file sizes. The text represents the information that should be hidden into the cover image.

A. First Experiment

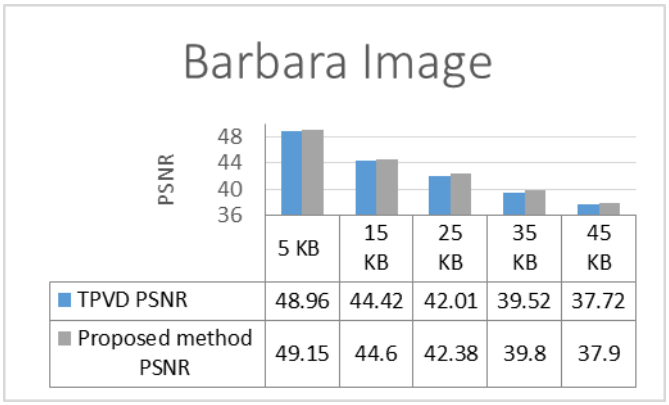
The first experiment is performed to compare the performances of the proposed algorithm RTPVD with the TPVD algorithm. The results are presented in figures 7-10.



	15 KB	30 KB	45 KB	60 KB	75 KB
TPVD PSNR	46.79	42.68	40.43	39.02	38.04
Proposed method PSNR	46.82	43.04	40.53	39.08	38.07

Fig. 7. First experiment for the Lena image.

The charts of the figures 7-12 represent the PSNR value with different file sizes using the TPVD and the RTPVD algorithms. Results show the PSNR values in the RTPVD algorithm are slightly better than those obtained by the TPVD algorithm for the given images of relatively small size.

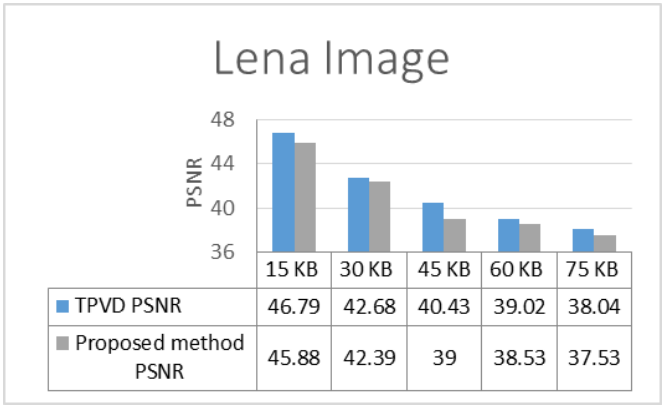


	5 KB	15 KB	25 KB	35 KB	45 KB
TPVD PSNR	48.96	44.42	42.01	39.52	37.72
Proposed method PSNR	49.15	44.6	42.38	39.8	37.9

Fig. 8. First experiment for Barbara image.

B. Second Experiment

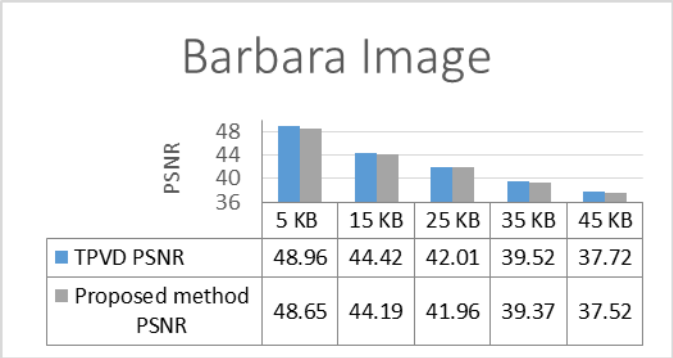
The second experiment consists of encrypting the information and then embed them into the cover image. This experiment is conducted to check whether the quality of the stego-image is preserved and to add another layer of security.



	15 KB	30 KB	45 KB	60 KB	75 KB
TPVD PSNR	46.79	42.68	40.43	39.02	38.04
Proposed method PSNR	45.88	42.39	39	38.53	37.53

Fig.9: Second experiment for lena image.

doi:10.15849/icit.2015.0045



	5 KB	15 KB	25 KB	35 KB	45 KB
TPVD PSNR	48.96	44.42	42.01	39.52	37.72
Proposed method PSNR	48.65	44.19	41.96	39.37	37.52

Fig. 10. Second experiment for the Barbara image.

Results show that, the PSNR value of the RTPVD is less than the PSNR of the TPVD algorithm. This due to the fact that the AES encryption algorithm is using the block cipher technique where the message is divided into blocks. Therefore, whenever the message is smaller than the block capacity, the AES algorithm will pad the message (i.e., add bits to the message) to fit into block size. The padding process affect slightly the quality of the image.

V. CONCLUSION

We have demonstrated that the TPVD steganography image-base algorithm can slightly be improved with a random selection of the starting pixels for every pixel-pair combination. This improvement can be significant with cover images of larger sizes and may increase the amount of secret information to be hidden. We showed also that the encryption of information will results on making the algorithm more secure but can lead to the deterioration of the stego-image quality. As future work, we plan to investigate on how to determine the maximum amount of encrypted information that can be embedded into a cover image of a given size without affecting its quality.

REFERENCES

[1] 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, pp. 37-44, 2008.

[2] N. Johnson and S. Jajodia, "Exploring steganography: seeing the unseen," IEEE Computer, pp. 26-24, 1998.

[3] T. Morkel, "Image steganography applications for secure communication," Master Thesis, University of Pretoria, 2012.

[4] K.G. Avinash, and M.S. Joshi, "An image steganography method with five pixel pair differencing and modulus function," International Journal of Computer Applications, vol. 68, no.1, pp. 20-26, 2013.

[5] W. Hong and T. Chen, "A novel data embedding method using adaptive pixel pair matching," IEEE Inf. Forensics Security, vol. 7, no. 1, pp. 176-184, 2012.

[6] A. Westfeld, "F5-A Steganographic algorithm: high capacity despite better steganalysis," In Proceedings of the Fourth International Workshop on Information Hiding, pp. 289-302, 2001.

[7] D.K. Sarmahand N. Bajpai, "Proposed system for data hiding using cryptography and steganography," Int. Journal of Computer Application, vol. 8, no. 9, pp. 7-10, 2010.

[8] R. Rivest, A. Shamir and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," Communication of the ACM, vol. 21, no. 2, pp. 120-126, 1978.

[9] J. Daemen and V. Rijmen,"The design of Rijndael: AES The advanced encryption standard," ISBN 3-540-42580-2 Springer-Verlag, New York, 2002.

[10] N.I. Wu and M.S. Hwang, "Data hiding: current status and key issues," International Journal of Network Security, vol. 4, no. 1, pp. 1-9, 2007.

[11] J. Korhonenand and J. You, "Peak signal-to-noise ratio revisited: is simple beautiful?," Fourth International Workshop on Quality of Multimedia Experience, pp. 37-38, 2012.

[12] H. Zhang, G. Geng, and C. Xiong, "Image steganography using pixel-value differencing," Second International Symposium on Electronic Commerce and Security, 2009.

[13] R. Ahirwal, D. Ahirwal, and Y. K. Jain, "A high capacitive and confidentiality based Image steganography using private stego-key," in Proceedings of the International Conference on Information Science and Applications (ISBN 978-81-907677-9-8), pp. 1-5, 2010.

[14] H.W. Tseng and H.S.Leng, "A steganographic method based on pixel-value differencing and the perfect square number," Journal of Applied Mathematics, vol. 2013, pp.1-8, 2013.

[15] W. Luo, F. Huang and J. Huang, "A more secure steganography based on adaptive pixel-value differencing scheme," Multimedia Tools and Applications, vol. 52, issue 2-3, pp 407-430.

[16] D.C. Wu DC, T. WH, "A steganographic method for images by pixel-value differencing," Pattern Recognition Letter, vol. 24, pp. 1613-1626, 2003.

[17] X. Zhang and S. Wang, "Vulnerability of pixel-value differencing steganography to histogram analysis and modification for enhanced security," Pattern Recognition Letter, vol. 25, pp. 331-339, 2004.