Adaptive Steganography Scheme Using More Surrounding Pixels

Masoud Afrakhteh
Dept. of Computer Science and Information Systems
Universiti Teknologi Malaysia
Johor, Malaysia
e-mail: masoudafrakhteh@gmail.com

Subariah Ibrahim
Dept. of Computer Science and Information Systems
Universiti Teknologi Malaysia
Johor, Malaysia
e-mail: subariah@utm.com

Abstract— **Information hiding has several branches. One is called as steganography. Steganography has a wide variety of techniques whereby a certain amount of secret information is embedded inside a medium. The medium format can be of any kind. However, the simplest file format is an 8-bit gray scale image. The original image is called a cover image. Likewise, once some secret bits are embedded, it is called stego-image. The most challenging issue is how capable a technique is to conceal the very existence of secret information in the stego-image. Steganographic techniques are evaluated in terms of imperceptibility and capacity. Imperceptibility means how visible secret information is to the human eyes in the stego-image. On the other hand, capacity means the maximum amount of secret information embedded. Conventional Least Significant Bit (LSB) insertion method is the basic concept for steganographic techniques. LSB method embeds secret bits in LSB(s) of the cover image. A pixel which carries a fraction of secret data is called a target pixel. The more surrounding pixels around a target pixel are utilized to find the most appropriate capacity value, the more precise capacity value is achieved. This paper proves that discovering the best capacity value brings about an improvement in terms of imperceptibility. Most of the steganographic techniques use either three or four adjacent pixels around a target pixel whereas the proposed technique is able to utilize at most all eight adjacent neighbors so that imperceptibility value grows bigger. The proposed method is called A-MSPU which stands for Adaptive More Surrounding Pixels Using.**

Keywords- adaptive; image's distortion; LSB method; more surrounding pixel

I. INTRODUCTION

Nowadays information hiding is a challenging area. Steganography is considered one of the branches in information hiding and it is the art of hiding secret information within the cover image [1][2]. The simplest steganographic method is carried out by embedding secret information in the least significant bit(s) (LSB) of the cover image. Two main issues are considered when using the algorithm. The first issue is imperceptibility which is the visibility of the secret information to the human eyes. The second issue is capacity, i.e., amount of secret information that can be concealed in the cover image [3].

Most of the steganographic methods consider human vision sensitivity as an essential factor [4] so that the very existence of the secret bits could not be revealed by any third party. One of the methods to achieve this is hiding the secret information in the edges rather than the smooth areas of a cover image. A method namely Multiple Base Notational Systems (MBNS) considers the edge areas of a cover image while re-expressing the secret bits in multiple base notational systems [3]. Using this technique MBNS adds more security. Furthermore MBNS proposes a walk path through the cover image which is not a row by row selection of target pixels. MBNS method owns the best values in terms of performance and quality metrics in comparison with bit-plane complexity segmentation method (BPCS) [5] and pixel value differencing (PVD) [3][6]. PVD replaces certain blocks of the cover image with the secret data and embeds the amount of difference between every two corresponding blocks so that a larger amount of difference makes a greater modification and vice versa. A method called optimal LSB method applies the same process of embedding used by simple LSB method but in the end, every target pixel value is to be adjusted [7][8]. To decrease the image distortion, the adjustment process discovers the closest candidate to be replaced with modified value of a target pixel. The best candidate value is the closest value to the original value of the target pixel so that the error is decreased and the quality metric values go higher. To add more security and avoid lumping the secret information, A-MELSBR method recommends using a probability parameter in order to scatter the secret information all over the cover image [9].

In this paper the main focus is to present an improvement made in terms of imperceptibility in comparison with MBNS method. The proposed method utilizes the same probability parameter to scatter secret bits as in [9] and considers the maximum number of surrounding pixels to achieve the capacity of every target pixel.

The adjustment technique is discussed in Section II. The proposed algorithm is explained step by step in Section III. In Section IV, the experimental results are presented. Finally in Section V, the desired conclusion is made based on the results achieved.

II. ADJUSTMENT TECHNIQUE

Once a fraction of secret information is embedded, the target pixel value is to be adjusted using a kind of process whose duty is to find the closest gray-scale value to the original value of the target pixel so that the difference between the original and modified value of a target pixel becomes smaller. Smaller amount of difference makes image distortion become smaller as well. In terms of imperceptibility, the value of a famous quality metric called Peak Signal to Noise Ratio (PSNR) grows larger as image distortion gets smaller. A higher PSNR value is regarded as an enhancement in imperceptibility and vice versa. Therefore the adjustment procedure can improve the quality of the stego-image. The adjustment procedure is as follows:

1. If a target pixel value is changed after embedding, two other pixel values will be generated by flipping the K+1th bit of the pixel P_i' as follows:

$$(P_-' \ , \ P_+') = \begin{cases} P_-' = P_i' + 2^k \\ P_+' = P_i' - 2^k \end{cases} \quad (1)$$

   where variable P_i' is the modified value of a target pixel i called P_i. P_i is already allowed to carry K bits of secret information in the first K least significant bits (K > 0). In case of finding either a value more than 255 for P_+' or a negative value for P_-', they should be replaced with P_i'.

2. Replace P_i' with the best candidate P_i'' as follows:

$$P_i'' = \begin{cases} P_i' & ,if\ |P_i - P_i'| \le |P_i - P_-'| \le |P_i - P_+'| \\ P_+' & ,if\ |P_i - P_+'| \le |P_i - P_i'| \le |P_i - P_-'| \\ P_-' & ,if\ |P_i - P_-'| \le |P_i - P_i'| \le |P_i - P_+'| \end{cases} \quad (2)$$

III. Steganographic Scheme

The step by step embedding phase is as follows:

1. Input a secret key SK in order to find the desired target pixels in a pseudo random way in the secret area. A cover image consists of two areas such as secret area and embedding area. The secret area comprises of the top-most row and the left-most column of the cover image while the embedding area consists of all the pixels other than the reserved pixels located in the secret area (Fig. 1). Secret information will be embedded in embedding area.
2. Input SN as a pseudo random number generator seed number to make a pseudo random selection of the target pixels located in the embedding area. Note that both SK and SN are seed numbers that initialize two pseudo random number generators (PRNG).
3. Let P, PL, M and N be the cover image, secret information size (in bits) and the number of rows and columns of the cover image respectively.

Reserved	**Reserved**	**Reserved**	**Reserved**	**Reserved**
Reserved	1		2	
Reserved	3			4
Reserved		5	6	

Figure 1. An example of how target pixels are chosen in order (1,2, …, 6). The reserved pixels are located in the secret area and the rest of pixels are regarded as the embedding area.

4. Assign the best value to the variable Mb, which is the maximum number of bits allowed to be embedded in a target pixel. The initial value for Mb will be the minimum Mb value using the following formula:

$$PL < M \times N \times Mb - M - N + 1 \quad (3)$$

5. Compute the maximum capacity of the cover image using the local variation formula as follows:

$$\sum_{i=2,j=2}^{M*N} \left| Min\ \left(8 \times \log_2\left(\sqrt{2 * |\sigma(\{SPS\})|}\right)\ ,\ Mb\right) \right| \quad (4)$$

   where σ is the local variance value and SPS is regarded as the set of all surrounding pixels of a pixel and contains at most eight adjacent pixels.

6. If PL is greater than maximum capacity, increase Mb by one and repeat step 5.
7. Calculate the probability value and the desired pseudo random numbers as below:

$$\begin{cases} Probability = PL\ /\ MaximumCapacity; \\ Rand('twister', SN); \\ RandMatrix = rand(M, N); \end{cases} \quad (5)$$

   The above equation is a simple PRNG and it is implemented using Mersenne Twister algorithm. This algorithm is the default built in PRNG method in MATLAB Versions 7.4 and later. Finally, RandMatrix is assigned by pseudo random numbers (0 < RandMatrix < 1).

8. Choose the desired target pixels located in the embedding area as follows:

$$\begin{cases} If\ Randmatrix(i,j) \le probability\ ,\ Chosen \\ Else \qquad Not\ chosen \\ \quad where\ i = 1..M\ and\ j = 1..N \end{cases} \quad (6)$$

9. Let $P(i,j)$ and $P'(i,j)$ be the original and modified values of a target pixel located in i_{th} row and j_{th} column of the cover image and stego image respectively.
10. Determine DynamicCapacity for $P(i,j)$ that is the number of bits can be embedded in a target pixel using the available surrounding neighbors. The pixels $P(i+1,j-1)$, $P(i+1,j)$, $P(i+1,j+1)$, $P(i,j+1)$ are considered as surrounding neighbors as long as they are not already chosen as in (6), otherwise they are ignored. If they were chosen, it is expected that a wrong value would be computed for the capacity of the target pixel. The reason is that the chosen adjacent pixels do not have the same values that they used to carry before their modification. However, there is no limitation for pixels such as $P(i,j-1)$, $P(i-1,j-1)$, $P(i-1,j)$ and $P(i-1,j+1)$. All these pixels are considered as fixed neighbors, i.e., they are always used for estimating the capacity of a target pixel (Fig. 2). Thus, the

DynamicCapacity for a target pixel can be calculated as follows:

$$Min\left(\left\lfloor \frac{log_2\left(\sqrt[2]{2*|\sigma(\{SPS\}-\{CSP\})|}\right)}{BestDelta} \right\rfloor , Mb \right) \qquad (7)$$

where total DynamicCapacity is achieved by making a summation of every target pixel DynamicCapacity. BestDelta is a regulation factor whose value ranges between zero and Mb. CSP is the set of chosen target pixels including any of the pixels $P(i+1, j-1)$, $P(i+1, j)$, $P(i+1, j+1)$ and $P(i, j+1)$.

11. Embed desired number of secret information based on the achieved DynamicCapacity value. Once embedding is done for a target pixel, its gray-scale value is called P_i' while original value is called P_i. For example, if "010" is a portion of secret information and P_i contains "01100110", P_i' will be "01100010". However there is a probability that a target pixel carries no secret information when the DynamicCapacity value equals to zero.
12. Adjust the value of each target pixel using adjustment technique in order to find the closest candidate to the original value of the target pixel. By doing this, the target pixel value will be replaced with the closest candidate and it results in an increase in PSNR value.
13. If there are any secret bits left not embedded, roll back changes made by steps 11 to 12, afterwards decrease BestDelta and repeat steps 10 to 12. Otherwise, correct the Probability, roll back changes made by steps 11 to 12 and go to step 8.

BestDelta and Probability are achieved by a simple divide and conquer searching algorithm. BestDelta ranges between zero and Mb and for correcting the Probability just decrease the initial value achieved in (5) as much as possible until all message bits are embedded. In other words, once BestDelta value is determined, the procedure is repeated as long as the smallest probability value embeds all the message bits using the BestDelta. By doing this, the algorithm scatters message bits all over the cover image and it keeps the algorithm from embedding the message bits at the beginning of the cover image.

To explain further, suppose that the Probability is equal to one. In this case, the respective target pixels of the cover image will be chosen sequentially row by row as in (6). On the contrary, if the value of the initial probability is smaller than a target pixel random value provided that a full embedment of the secret bits is achieved, the minimum probability should be chosen. In this case, message bits are scattered as much as possible and the selection is no longer sequential.

Always use	Always use	Always use
Always use	**Target pixel**	*Use if not chosen*
Use if not chosen	*Use if not chosen*	*Use if not chosen*

Figure 2. Figure shows how to choose surrounding neighbors of a target pixel for estimating the maximum capacity (bits) that can be embedded in every target pixel.

14. Embed the secret parameters in the secret area. Once the desired values for BestDelta and Probability are discovered, these secret parameters along with the rest of secret parameters such as SN, PL and Mb should be embedded in the secret area. The embedding process needs SK shared by both the sender and the receiver in order to find the coordinates of the target pixels in the secret area. Simple LSB insertion method is applied for embedding the secret parameters inside the secret area using one LSB of each target pixel selected using SK as in Step 1.

The modified cover-image is called stego image. In order to recover secret information from the stego image, the receiver is expected to know SK so as to extract the rest of secret parameters. Knowing SK, the secret parameters can be extracted easily. The extraction procedure is as follows:

1. Extract secret parameters using SK.
2. Generate random numbers as in (5).
3. Choose the desired target pixels as in (6).
4. Compute the DynamicCapacity number of a target pixel as in (7).
5. Extract the secret information from a target pixel using the corresponding DynamicCapacity.
6. If the amount of bits extracted equals to PL, the whole secret information is achieved by putting together all the extracted bits.

IV. Experimental Results

The algorithm is applied on different standard images such as Man, Lena and Finestones using different PL values. Several quality metrics are available to represent how much an image quality is decreased by embedding process. As described earlier, PSNR is the most famous quality metric. PSNR values above 36 dB represent excellent image quality [10] and make any changes in the stego-image look completely invisible to the human eyes.

The results of embedding secret information in Man are depicted in Table I. It shows how PSNR value decreases as PL value increases. A parameter called Embedding Rate is calculated with dividing PL value by the total number of bits available in a cover image, namely total number of bits in an 8-bit gray-scale image of size 512×512 equals to 512×512×8. In other words, embedding rate implies a kind of percentage of available bits utilized in a cover image so as to embed the corresponding value of PL. The stego-image generated for Man using BestDelta value 0.75 is shown in Fig. 3. The same experiments are repeated for various standard images to make a comparison with MBNS method. As shown in Table II, A-MSPU method has higher PSNR values than MBNS method.

By using human eyes, it is rather difficult to detect any changes in the quality of the cover image as it is supposed to be invisible to the human eyes and both cover image and stego-image look almost identical (Fig. 4). Error images can be useful to distinguish manipulated pixels from the rest of pixels in a stego-image. A target pixel may be modified in a few numbers of LSBs so that the gray scale values have a difference in value before and after embedding. The difference can be either zero or a bigger value. It will be zero merely in two cases, i.e., whenever either a target pixel LSB(s) is overwritten with the same bit(s) similar to the corresponding pixel in the cover image or there is nothing embedded in a target pixel as the respective DynamicCapacity for a target pixel may equal to zero. The more difference value, the lesser quality and imperceptibility. As in Table I, the maximum number of Mb equals to four that is by using five bits the PSNR value drops below 36 dB which is regarded as being visible to the human eyes. Thus the largest value which can be represented by four bits is 15 or $2^4 - 1$. As the cover image format is an 8-bit gray scale image, the error image has the same format. In an 8-bit gray scale image 256 colors can be shown. To map all 16 colors to 256 colors, number 15 should be multiplied by 17. To represent the manipulated target pixels with dark pixels as depicted in Fig. 4, the desired error image can be achieved as follows:

$$ErrorImage = 255 - 17 \times |p(i,j) - p'(i,j)| \quad (8)$$

The error images facilitate distinguishing embedded areas (edges) from the areas that carry either no secret bits or a lesser amount of bits (smooth areas). The darker a pixel looks, the more bits are embedded in that pixel and vice versa. In Fig. 4, error image (c) does not preserve edge areas as much as error image (b). The reason is that error image (c) holds more secret information and is limited to utilize at most three bits (PL=610000, Mb=3) of each target pixel (Table I). On the contrary, edges in error image (d) look clearer than error image (c) as error image (d) utilizes four bits.

V. CONCLUSION

A-MSPU improves imperceptibility in comparison with MBNS method for two reasons. The first reason is that A-MSPU discovers the optimum value for Mb while MBNS method merely uses a fixed number such as four to limit the usage of bits in a target pixel, i.e., when it is possible to embed a typical amount of bits using two bits, there is no need to use four bits. In this regard, PSNR value drops nearly about 4 dB for 260000, 540000 and 740000 bits using one more bit for Mb (Table I). The second reason is that A-MSPU discovers more surrounding pixels rather than utilizing four adjacent neighbors (Known as Fixed neighbors in step 9). Fixed neighbors are modified in previous steps so that they cause an increasingly image distortion whereas the rest of unselected neighbors still keep their original gray scale values by the end of embedding process. By using unselected neighbors as well as fixed neighbors, every DynamicCapacity value can be estimated more precisely and also image distortion will be decreased. Lesser image distortion results in getting higher quality levels as higher PSNR values are achieved in Fig. 5.

ACKNOWLEDGMENT

This work is supported by the Ministry of Science & Technology and Innovation (MOSTI), Malaysia and Research Management Center, Universiti Teknologi Malaysia under the VOT: 79309.

REFERENCES

[1] F. A. P. Petitcolas, R. J. Anderson and M. G. Kuhn, “Information hiding—A survey,” Proc. IEEE, vol. 87, pp. 1062–1078, 1999.

[2] H. Wang and S. Wang, “Cyber warfare—Steganography vs. Steganalysis,” Commun. ACM, vol. 47, no. 10, pp. 76–82, 2004.

[3] X. Zhang and S. Wang, “Steganography using multiple-base notational system and human vision sensitivity,” IEEE Signal Processing Letters, vol. 12, pp. 67–70, Jan. 2005.

[4] M. Kutter and S. Winkler, “A vision-based masking model for spread- spectrum image watermarking,” IEEE Trans. Image Processing, vol. 11,pp. 16–25, Jan. 2002.

[5] H. Noda, J. Spaulding, M. N. Shirazi, and E. Kawaguchi, “Application of bit-plane decomposition steganography to JPEG2000 encoded images,” IEEE Signal Processing Lett., vol. 9, no. 12, pp. 410–413, Dec. 2002.

[6] D.-C. Wu and W.-H. Tsai, “A steganographic method for images by pixel-value differencing,” Pattern Recognit. Lett., vol. 24, pp. 1613–1626, 2003.

[7] C. K. Chan and L. M. Cheng, “Hiding data in images by simple LSB substitution,” Pattern Recognition, pp. 469–474, Mar. 2004.

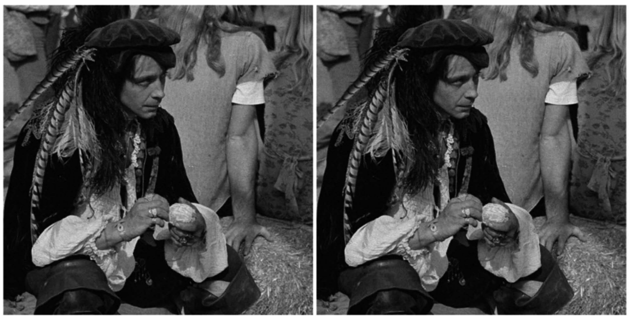
[8] N.I. Wu and M.S. Hwang,”Data Hiding: Current Status and Key Issues,” International Journal of Network Security, Vol.4, No.1, PP. 1–9, 2007.

[9] Y.K. Lee and L. h. Chen, “An Adaptive Image Steganographic Model Based on Minimum-Error LSB Replacement,” In Proceedings of the Ninth National Conference on Information Security. Taichung, Taiwan, 14-15, pp.8-15, 1999.

[10] D. B. Russakoff, T. Rohlfing, K. Mori, et al., “Fast generation of digitally reconstructed radiographs using attenuation fields with application to 2D-3D image registration,” IEEE Transactions on Medical Imaging, vol. 24, no. 11, pp. 1441–1454, 2005.

TABLE I. PARAMETERS ACHIEVED USING A-MSPU OVER STANDARD IMAGE “MAN” OF SIZE 512×512.

PL (bits)		**10000**	**50000**	**100000**	**150000**	**260000**	**420000**	**460000**	**520000**	**610000**	**740000**
A-MSPU Method	**Embedding Rate**	0.01	0.02	0.05	0.07	0.12	0.20	0.22	0.25	0.29	0.35
	Mb (bits)	1					2		3		4
	BestDelta	1.9375	1.9375	1.9375	1.9375	1.9375	1	0.75	1.0313	0.75	0.75
	PSNR (dB)	65.26	58.36	55.32	53.58	50.06	47.42	46.96	43.34	42.13	37.16



(a) (b)

Figure 3. (a) Cover image. (b) Stego-image with 610000 bits embedded using BestDelta 0.75.

(a) (b) (c) (d)

Figure 4. Corresponding PL values for error images (a),(b),(c) and (d) are 46×10^4 ,52×10^4 ,61×10^4 and 74×10^4 bits respectively.

TABLE II. AVERAGED PSNR VALUES ACHIEVED USING STANDARD IMAGES OF SIZE 512×512.

PL (bits)		10000	50000	100000	150000	260000	420000	460000	520000	610000	740000
Embedding Rate		0.01	0.02	0.05	0.07	0.12	0.20	0.22	0.25	0.29	0.35
PSNR (dB)	A-MSPU Method	64.33	57.4	54.38	52.56	49.97	46.48	45.63	42.62	41.3	38.44
	MBNS Method	65.146	57.746	53.856	51.211	47.214	43.143	42.441	41.17	39.758	37.477

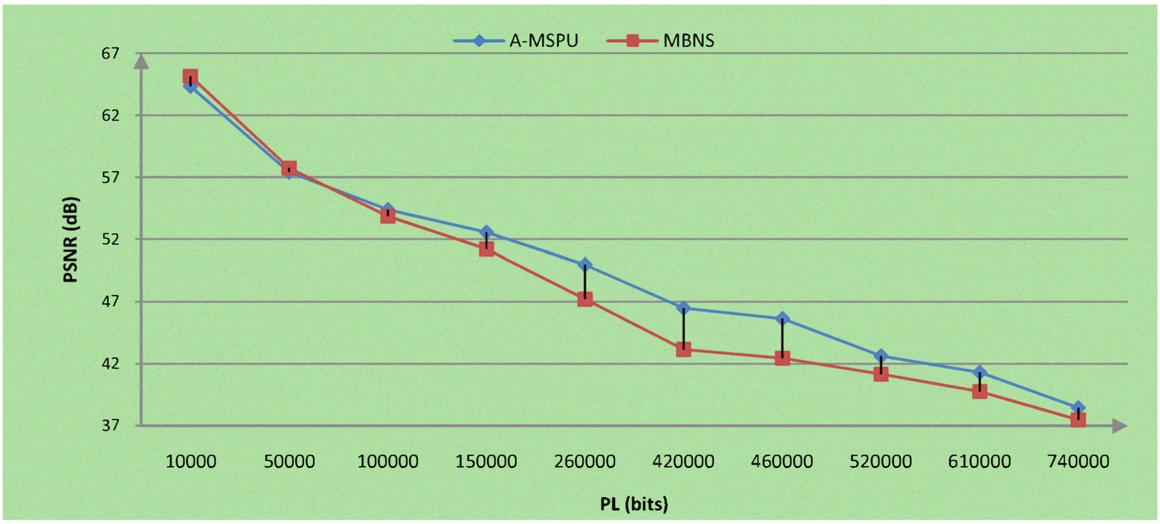


Figure 5. A-MSPU method quality performance in comparison with MBNS method (based on TABE II).