Analysis of Secure Text Embedding using Steganography

Rupinder Kaur
Department of Computer Science and Engineering
BBSBEC, Fatehgarh Sahib, Punjab, India

Deepak Aggarwal
Department of Computer Science and Engineering
BBSBEC, Fatehgarh Sahib, Punjab, India

Abstract - Information hiding techniques play a vital role in the recent years. Steganography is one of the important information hiding technique which hides the existence of the message in the cover file. The cover file can be text, image, audio or video file. Image steganography is the emerging trend in the communication field. The amount of information hidden inside the image depends upon the resolution of an image. But with the increased capacity the quality of image should not be degrade. The aim of this thesis is to analyze the secure data hidden inside the image using steganography. It is based on spatial domain LSB technique. Information is embedded within the images using 1-bit, 2-bits and 3-bits LSB of each RGB component using 3 different cases. The original image is not required within the decoding process. The experimental results are compared using MSE and WPSNR parameters. The proposed scheme is secure because the intruder cannot directly tell the existence of data hidden inside the image.

Keywords – Steganography, LSB, MSE, WPSNR

I. INTRODUCTION

In the past years, several information hiding techniques are used which hides the data inside other object. These days, all the information is stored in a digital form. The other objects can be data, image, audio or video. Images are one of the most important carrying media which can be used for hiding the information. This process of hiding information into other object is referred to as a steganography. There are two techniques available for transmitting the secret information between the communicating parties. One is cryptography, in which the structure of the message is scrambled to make it meaningless and can be reconstructed only by the holder of a key. It offers the ability of transmitting the information between communicating parties in a way that prevents the third party from reading it. When secret message is transmitted, it is observable by anyone means it does not attempt to hide the fact that a message exists. The second technique is steganography, which hides the secret message into other object. It does not alter the structure of the secret message, but hides it inside a cover object so that it cannot be seen by any observer. Steganography system involves a vast array of secret communication methods that conceal the message’s very existence. The steganography techniques are similar to that of digital watermarking; however there is still difference between them. In digital watermarking, the information is embedded into a digital signal in a way that is difficult to remove. On the other hand, steganography focuses on making the fact that the secret message does not exist within the system.

The paper explores the analysis of secure data hidden inside the image using steganography. Proposed embedding and extraction algorithms are explained in section III. Experimental results are presented in section IV. Concluding remarks are given in section V.

II. STEGANOGRAPHY

2.1 Steganography

Steganography is derived from the Greek for covered writing and essentially means “to hide in plain sight”. It is the art and science of communicating in such a way that the presence of a message cannot be detected. Simple steganographic techniques have been in use for hundreds of years, but with the increasing use of files in an electronic format new techniques for information hiding have become possible.

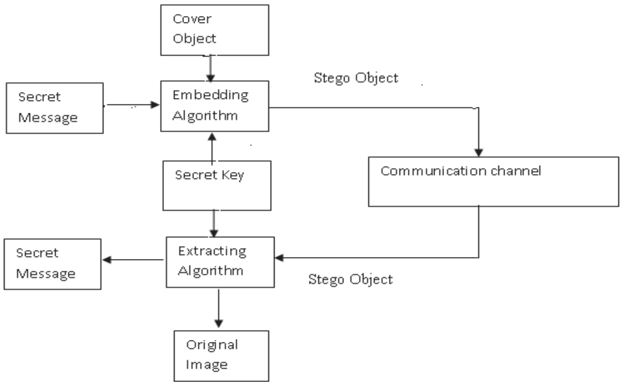


Figure 1. General Steganography System

2.2 LSB Technique

This is one of the simplest and easiest method of hiding the data in images. In this technique, the data in the binary form is to be hidden into the LSBs of the carrier bytes or in pixels of image. The overall change to the image is so small that human eye would not be able to discover**.** In 24-bit images each 8-bit value refers to the red, green and blue color. But in 8–bit images each pixel is of 8-bits, so each pixel stores maximum 256 colors.

For example, suppose we have to hide a message in 24-bit color image whose RGB components are separated and the original image pixels are:

(00100111 11101001 11001000) (00100111 11001000 11101001) (11001000 00100111 11101001)

Suppose we want to hide character “A” within the image whose binary representation is 01000001 within the image using LSB of each RGB component. After hiding the resulting image pixels are:

(00100110 11101001 11001000) (00100110 11001000 11101000) (11001000 00100111 11101001)

In each pixel only 3-bits of character are hidden by changing the LSB of each RGB component.

III. PROPOSED WORK

3.1 Embedding Algorithm

The proposed algorithm includes the hiding of encrypted data within the binary image using LSB technique. The data is hidden inside the binary image using LSB in 3 different cases i.e.1-bit, 2-bits and 3-bits of each pixel. Steps to embed data are as follows:

Inputs: Image file and text to embed.

Output: Text embedded Image.

Procedure:

Step 1: The original color image and text is taken as an input. From this image, the RGB components of each pixel will be separated and these extracted pixels will be stored within the pixel-array.

Step 2: Extract all the characters from the text file for encryption and store them in a character-array.

Step 3: Convert the encrypted text to the number system using ASCII code to generate the binary matrix.

Step 4: For embedding data inside the image, the RGB component of each pixel having certain intensity value is converted into binary form and stored within the binary matrix.

Step 5: Choose the first pixel from the binary matrix and pick the data bits of the corresponding character. Replace the LSB of each RGB component with data bits from the binary matrix using 3 different cases i.e. 1-bit, 2-bits and 3-bits of each pixel.

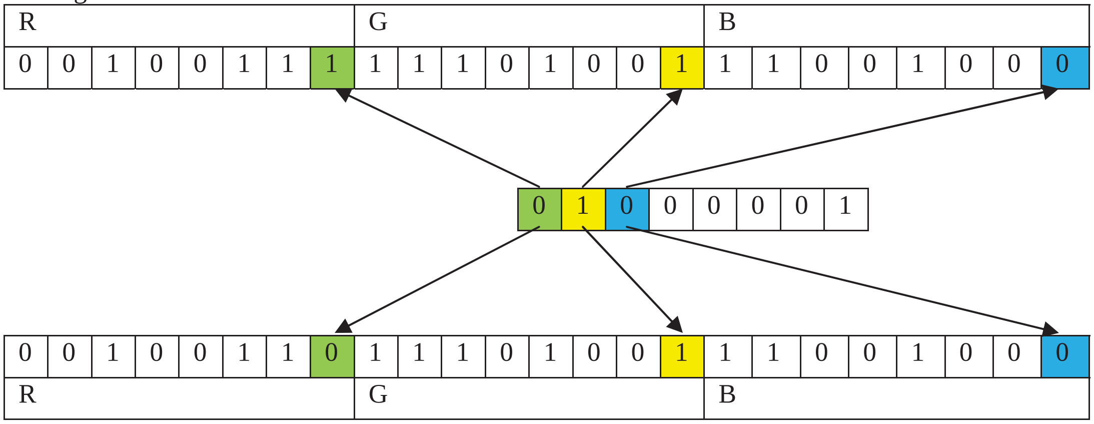
Suppose we have to hide a message in 24-bit color image whose RGB components are separated and the original image pixels are:

R								G								B							
0	0	1	0	0	1	1	1	1	1	1	0	1	0	0	1	1	1	0	0	1	0	0	0

Figure 2. RGB Representation of pixel

Suppose we want to hide character "A" within the image whose binary representation is 01000001.

Case 1: To replace the 1-bit of each RGB component with the data bits. In this case only 3-bits per pixel are used for hiding the data.



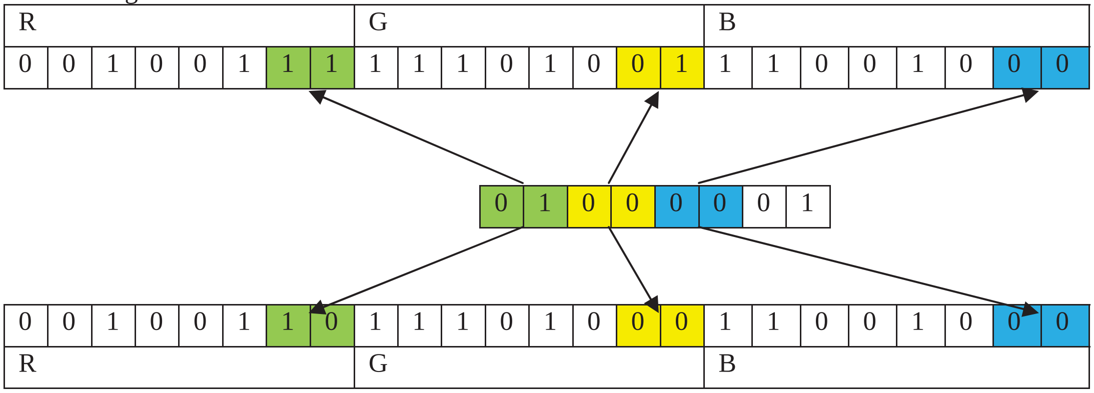
R								G								B							
0	0	1	0	0	1	1	1	1	1	1	0	1	0	0	1	1	1	0	0	1	0	0	0

0	1	0	0	0	0	0	1

0	0	1	0	0	1	1	0	1	1	1	0	1	0	0	1	1	1	0	0	1	0	0	0
R								G								B							

Figure 3. Data hiding using 3-bits per pixel

Case 2: To replace the 2-bits of each RGB component with the data bits. In this case, 6-bits per pixel are used for data hiding.



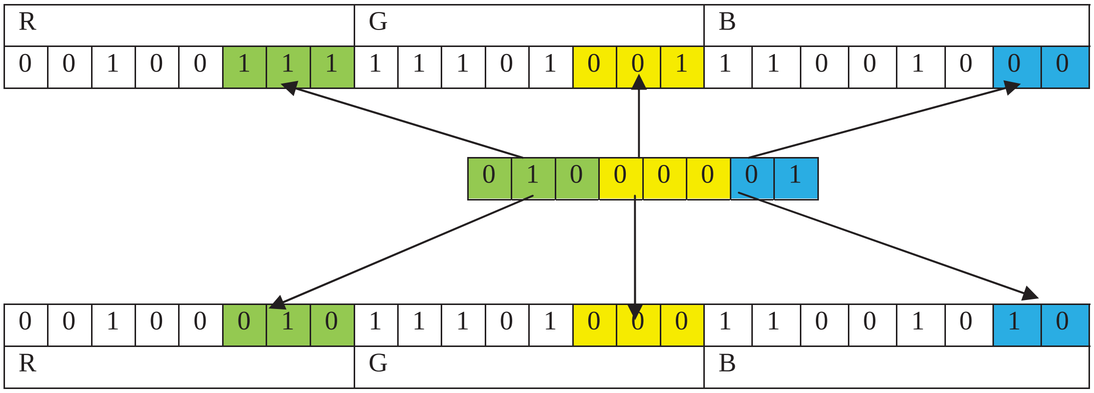
R								G								B							
0	0	1	0	0	1	1	1	1	1	1	0	1	0	0	1	1	1	0	0	1	0	0	0

0	1	0	0	0	0	0	1

0	0	1	0	0	1	1	0	1	1	1	0	1	0	0	0	1	1	0	0	1	0	0	0
R								G								B							

Figure 4. Data hiding using 6-bits per pixel

Case 3: To replace the 3-bits of each pixel with the data bits. In this case 9-bits per pixel are used to hide the data bits.



R								G								B							
0	0	1	0	0	1	1	1	1	1	1	0	1	0	0	1	1	1	0	0	1	0	0	0

0	1	0	0	0	0	0	1

0	0	1	0	0	0	1	0	1	1	1	0	1	0	0	0	1	1	0	0	1	0	1	0
R								G								B							

Figure 5. Data hiding using 9-bits per pixel

Repeat the same process till all the bits of A has been embedded.
Step 6: Repeat step 5 till all the characters have been embedded.
Step 7: Obtained the new image by combining the data bits and the pixels of the original image.
Step 8: After getting the final color image the original color image and the image with hidden data will be compared by using different parameters.

We have used performance parameters like WPSNR, MSE to check the impact on image, of replacement of different pair bits. The first term is the MSE. The MSE is the cumulative square error between the compressed and the original image. The MSE is often called quantization error variance σ_q^2 and its formula is given by

$$MSE=\sigma_q^2= \qquad {}^2$$

Where the sum over j, k denotes the sum over all pixels in the image and N is the total number of pixels in an image.

WPSNR (weighted peak signal to noise ratio):-Other important term related to this is the PSNR. PSNR is a measure of the peak error. It is used to test the change in the quality of image after applying various attacks. The mathematical formula is given by

$$PSNR = 20 * \log10$$

In WPSNR, some weight age value is assigned to the pixels of the image. It is calculated by the same formula of PSNR. The only difference is the weight age value assigned to the pixels. The weight age value is the random value that is assigned to all the pixels.

$$WPSNR = 10*\log10 (255*255/MSE)$$

Where n is the number of bits used to represent per pixel value and 255 represents the value of each pixel. A lower value for MSE means lesser error and this result in a high value of PSNR. Logically, a higher value of PSNR is good because it means that the ratio of signal to noise is higher. Here, the 'signal' is the original image, and the 'noise' is the error in reconstruction. So we can say that a compression scheme having a lower MSE (and a high PSNR) is a better compression scheme.

3.2 Extracting algorithm

In extraction algorithm the original image is not required. It requires only the output color image and the steps are:
Step 1: Consider two arrays and let these arrays are Character Array and pixel-array.
Step 2: Extract all the pixels of the output image. These extracted pixels are separated using RGB components and are stored within the pixel-array.
Step3: Start scanning from first pixel of the pixel-array and extract LSB of each RGB component. Store the extracted LSB in the character array.
Step 4: Generate the message by combining the bits of the character array.

We choose this method because it provides the secrecy as well as the privacy of information. To obtain privacy we have used the concept of cryptography and on the other hand to implement secrecy, we have used steganography.

IV.EXPERIMENT AND RESULT

The proposed scheme uses the color image to embed data in spatial domain. Secrecy is the important issue and to provide secrecy data is hidden inside the image using LSB technique. Data is embedded within the image using 1-bit, 2-bits and 3-bits of each RGB component of the pixel. The experimental results show the comparison between images using MSE and WPSNR value in 3 different cases. It also shows the results in graphs. It is observed that with the decreasing value of MSE the WPSNR value increases. It means the stego image is closer to the original image and less distortion occurs. Here the size of the image is 300*400.Figure 6-8 shows the comparison of WPSNR and MSE using three images.

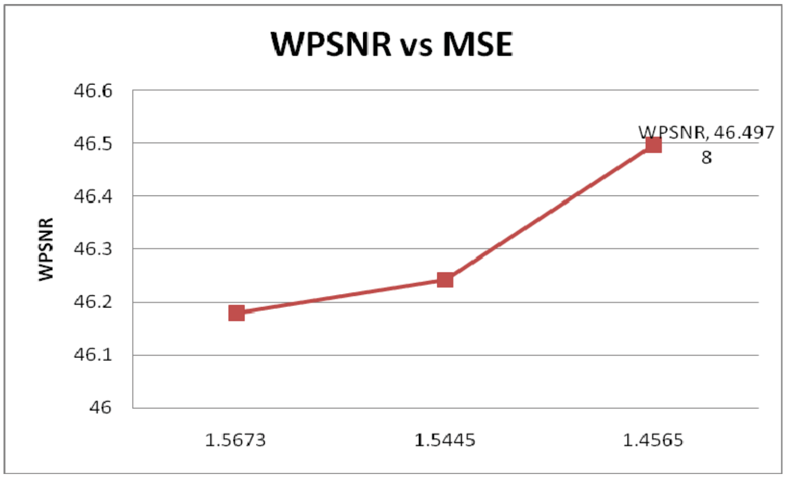


Figure 6. Result of WPSNR vs. MSE for hiding 3-bits in each pixel

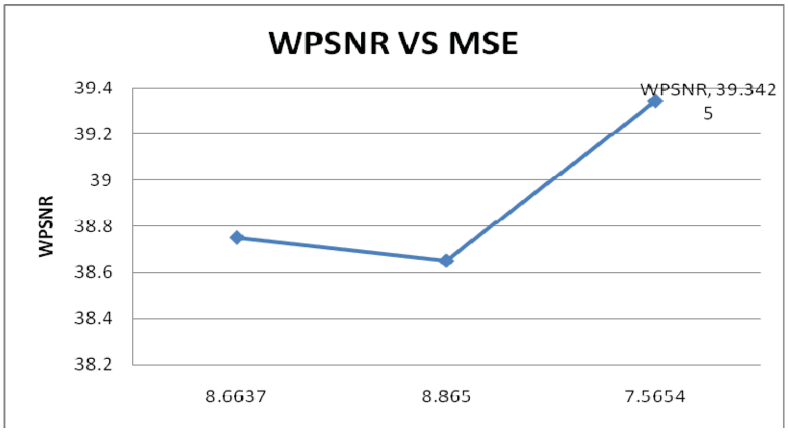


Figure 7. Result of WPSNR vs. MSE for hiding 6-bits in each pixel

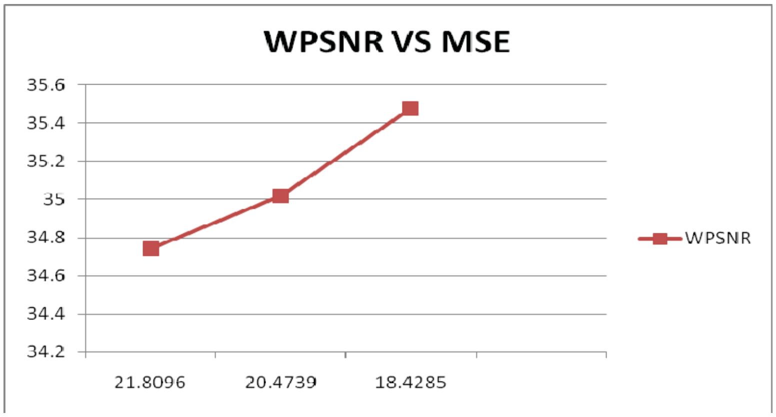


Figure 8. Result of WPSNR vs. MSE for hiding 9-bits in each pixel

Table 1 Comparison of images using 1-bit of each RGB component

Parameters used	Images Used		
	Animals Image	Car Image	Flower Image
MSE	1.5673	1.5445	1.4565
WPSNR	46.1793	46.2428	46.4978

Table 1 shows that in all the three images the WPSNR value increases with decrease in MSE value. In this case, 1-bit LSB of each RGB component is used and 3-bits per pixel are used for hiding the text. In 300*400 images, the total number of bytes that can be hidden inside the image are 45000bytes

Table 2 Comparison of images using 2-bits of each RGB component

Parameters used	Images Used		
	Animals Image	Car Image	Flower Image
MSE	8.6637	8.865	7.5654
WPSNR	38.7538	38.654	39.3425

Table 2 shows that in all the three images the WPSNR value increases with decrease in MSE value. But the WPSNR value is less as compared to the previous case because more bits are used for data hiding. In this case, 2-bits LSB of each RGB component is used and 6-bits per pixel are used for hiding the text. In 300*400 images, the total number of bytes that can be hidden inside the image are 90000bytes.

Table 3 Comparison of images using 3-bits of each RGB component

Parameters used	Images Used		
	Animals Image	Car Image	Flower Image
MSE	21.8096	20.4739	18.4285
WPSNR	34.7443	35.0188	35.4759

From table 3,it is observed that the value of WPSNR is less as compared to all the previous cases as 3-bits per RGB component are used for hiding the data. It uses 9-bits per pixel for hiding the data. In 300*400 images, the total number of bytes that can be hidden inside the image are 180,000bytes

V. CONCLUSION

The presented work is used to hide the data inside the image using spatial based domain technique. It hides the data using different number of LSB in three different cases. It shows the results of comparison in LSB of 1-bit, 2-bits and 3-bits using MSE and WPSNR parameters. Test is based on three images. The original image is not required for the decoding process. The experimental result shows that data hiding up to LSB of 3-bits is permissible. The permissible value of WPSNR is more than 20 decibel. It implies that hiding the secret message within the image up to LSB of 3-bits is less visible to human eye and the quality of image is better as compared to other cases. It fulfills the requirement of both security and quality of an image. It is secure because the intruder cannot directly tell there is a hidden message within the image by seeing it visually. It does not degrade the quality of image by embedding excessive amount of data.

REFERENCES

[1] A.A.A Gutub, M.M Fattani, “A Novel Arabic Text Steganography Method Using Points and Extensions”, Proceedings of WASET, 28-31, May 2007.

[2] N. N. Emam, “Hiding a large amount of data with high security using steganography algorithm”, Journal of Computer Science, 223 – 232, April 2007.

[3] Kh. M.Singh, S. B. Singh and L. S.S.Singh, “Hiding Encrypted Message in the Features of Images”, IJCSNS, vol. 7, No.4, April 2007

[4] G. Sahoo, R. K. Tiwari, “Designing an Embedded Algorithm for Data Hiding using Steganographic Technique by File Hybridization”, IJCSNS, vol. 8, No. 1, pp. 228-233, January 2008.

[5] D.Bhattacharyya, P.Das, S. K.Bandyopadhyay and T.H.Kim, “Text Steganography: A Novel Approach”, International Journal of Advanced Science and Technology vol. 3, February, 2009.

[6] S. K. Bandyopadhyay, “An Alternative Approach of Steganography using Reference Image”, International Journal of Advancements in Technology, ISSN 0976-4860, June, 2010.

[7] Akbas E. Ali, “A New Text Steganography Method by Using Non-Printing Unicode Characters”, Eng. & Tech. Journal, Vol.28, No.1, 2010.

[8] S.Narayana and G.Prasad, “Two new approaches for secured image steganography using cryptographic techniques and type conversions”, Signal & Image Processing: An International Journal (SIPIJ) vol.1, No.2, December 2010.

[9] Atallah M. Al-Shatnawi,”A New Method in Image Steganography with Improved Image Quality”, Applied Mathematical Sciences, Vol. 6, 2012, no. 79, 3907 - 3915

[10] Vijay Kumar Sharma, Vishal Shrivastava,” A Steganography Algorithm for hiding image in image by improved LSB substitution by Minimize detection”, Journal of Theoretical and Applied Information Technology, Vol. 36 No.1, February, 2012