Proposed Method of Text Hiding in Image Edges

Nasseer M. Basheer
Ph.D., Lecturer, Technical College/Mosul, Iraq

Ashty M. Aaref
Ph.D., Software Engineering Department, College Technology/Kirkuk, Iraq

Dhafer J. Ayyed
M. Tech Student, Computer Engineering Department Technical College/Mosul, Iraq

ABSTRACT
Nowadays, information hiding techniques have been beneficial in a many application areas, there are many techniques to achieve hiding data, and hiding text inside image is one field of them. This work shows how the edges of the images will be detected by scanning method using 3x3 window and then the secret message is canceled in edges of a gray scale images (Lena and goldhill) which acts as a cover images using Least Significant Bit (LSB) based Sobel edge detection algorithm. The algorithms are implemented using MATLAB R2014a. The design achieved high embedding capacity and high quality of encoded image. A Four Gray Scale input images with size 1024×1024 is used as a cover image and massage with length 500 character as a secret massage in this work.

Keywords
Information Hiding, Steganography, Cover image, secret message, LSB.

1. INTRODUCTION
The rapid expansion of the Internet and the overall development of digital technologies in the past years have sharply increased the availability of digital images. It is, therefore, very important to have the capabilities to detect copyright violations, and to control access to digital media when transmitting sensitive data over an insecure channel. Hence appear information hiding, one of the image processing techniques to hide the transmission of confidential data and remove doubt in the existence of hidden information, concealment techniques have been used for thousands of years as a means to achieve the secret connection .And one of the famous hiding techniques is Steganography and Watermark [1].The Figure (1) illustrates the classification of information hiding.

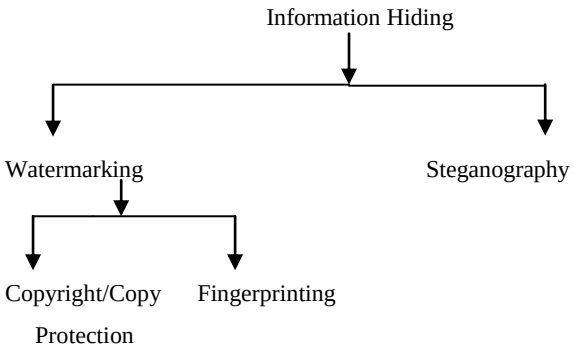

Figure 1 :Information Hiding

2. STEGANOGRAPHY
The word steganography comes from the Greek Steganos, which mean covered or secret and –graphy mean writing or drawing. Therefore, steganography means, literally, covered writing. Steganography is the art and science of hiding information such that its presence cannot be detected [2]. and a communication is happening [3]. A secret information is encoding in a manner such that the very existence of the information is concealed. Paired with existing communication methods, steganography can be used to carry out hidden exchanges. The main goal of steganography is covert communication to hide a message from a third party. The basic model of steganography consists of cover message (is the carrier of the message such as image, audio, text, video or some other digital media), Message (is the information which needs to be concealed in the suitable digital media) and stego-message (which is the cover message with message embedded inside it) [4]. All steganography mechanism consist of two stages, namely embedding stage and the recovery stage (also called extraction stage) which are shown in Figure (2) and Figure (3) respectively.

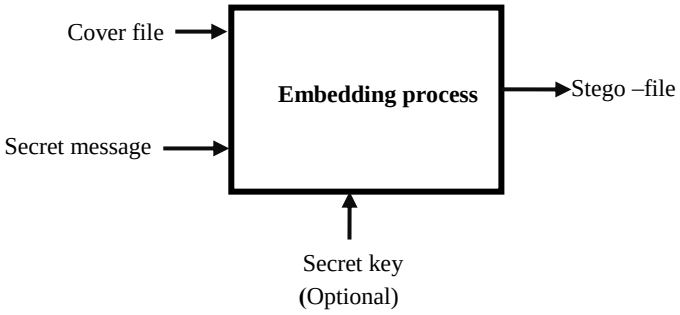

Figure 2 : A Generic Scheme of Steganography Embedding Model

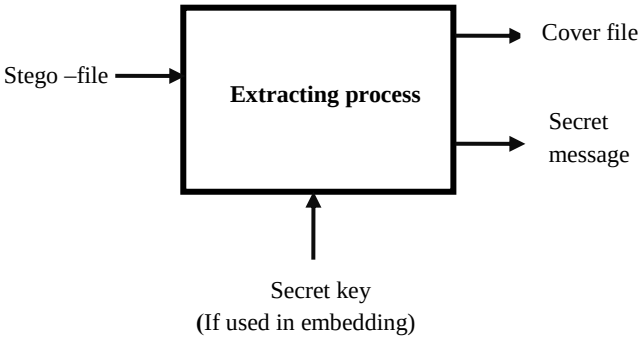

Figure 3 : A Generic Scheme of Steganography Recovery Model

In this paper edge pixels were selected in order to hide the data. One common way to hide the data is "Least Significant Bit Insertion". This method modifies the low order bit of each pixel to match the message to hide. The selection of pixels in which the message will be embedded is very important because modified pixels in areas of the image where there are pixels that are most like their neighbors are much more noticeable to the naked eye. A single modified pixel stands out among its uniform neighbor pixels thus making the image suspicious. One possible solution for this problem is to select the edge-pixels of the image to hide the message. It is not noticeable when a single pixel is modified when its surrounding pixels are least like it [5].

3. SOBEL EDGE DETECTION OPERATOR

In case of Sobel Edge Detection, there are two masks, one mask identifies the horizontal edges and the other mask identifies the vertical edges. Each of the masks has the effect of calculating the gradient in both vertical and horizontal direction. These Sobel masks are convolved with smoothed image and giving gradients in i and j directions is given by [6]:

$$Gi = Gx * F(i,j) \quad \text{and} \quad Gj = Gy * F(i,j)$$

Sobel masks are showing in Figure (4).

-1	-2	-1
0	0	0
1	2	1

-1	0	1
-2	0	2
-1	0	1

Figure 4 : Horizontal operator and Vertical operator [6]

Equation (1) shows convolution of input image with horizontal mask and Equation (2) shows convolution of image with vertical mask [8].

$$Gx = \{f(x+1,y-1) + 2f(x+1,y) + f(x+1,y+1)\} - \begin{Bmatrix} f(x-1,y-1) + 2f(x-1,y) + \\ f(x-1,y-1) \end{Bmatrix} \quad ..(1)$$

$$Gy = \{f(x-1,y-1) + 2f(x,y-1) + f(x+1,y-1)\} - \begin{Bmatrix} f(x-1,y+1) + 2f(x,y+1) + \\ f(x+1,y+1) \end{Bmatrix} \quad ..(2)$$

These masks can then be combined together to find the absolute magnitude of the gradient at each point. The gradient magnitude is given by [6]:

$$G = \sqrt{Gx^2 + Gy^2} \quad ..(3)$$

4. THRESHOLDING

Thresholding is a relatively simple approach of image segmentation [7]. Thresholding becomes a simple but effective tool to separate objects from the background [8]. The way to extract the object from the background is to select a threshold T. Then, any point (x,y) in the image at which f (x,y) >T is called an object point; otherwise the point is called a background. Segmented image f (x,y) can be represented as below [9]:

$$f(x,y) = \begin{cases} 1 & \text{if } f(x,y) \geq T \\ 0 & \text{other wise} \end{cases} \quad ..(4)$$

The simplest methods used to determine the threshold value and that have been applied in this thesis. (Mean image data values are calculated as follows:

$$T = \frac{1}{H*W} \int_{i=1}^{H} \int_{j=1}^{W} f(i,j) \quad ...(5)$$

H=high of image. W=width of image.

5. PROPOSED WORK

5.1 LEAST SIGNIFICANT BIT EMBEDDING

Embedding technique in the algorithm is based on replacing the LSB of the pixel (f (i,j)) with the message bits one by one. Hence if the message is equivalent to m-bits there are m-pixels to deal with, whose least significant bits will be replaced by the m-message bits. The embedding procedure can be described using the equation as follows:

$$Is\,(f(i,j)) = \begin{cases} f(i,j) - 1 & LSB=(f(i,j))=1 \text{ and } m=0 \\ f(i,j) & LSB=(f(i,j))=m \\ f(i,j) + 1 & LSB=(f(i,j)) \neq 0 \text{ and } m=1 \end{cases}$$

In general, a p-by-q image is simply a p-by-q matrix, where each entry in the matrix is a positive integer called the pixel value, which determine the color of that pixel. For an n-bit image, these pixel values range from 0 to $2n - 1$. In other words, the possible color values for each pixel in an n-bit image are the colors corresponding to the bit strings of length n. Unless there is a specific need to use the bit string representations of pixel values, we will typically use the decimal representations. In this work, primarily an 8-bit grayscale images are used. These images are thus p-by-q matrices of integers ranging from 0 to 255, where 0 corresponds to black, 255 to white, and the values in between form a spectrum of varying shades of grey (i.e., darker shades nearer 0 and lighter shades nearer 255) The least significant bit (LSB) is the bit corresponding to 20, that is, the bit that makes a value even or odd. Since these grayscale values form a spectrum ranging in order from dark to light, each grey value varies little from the values on either side of it. For example, the grey value 100 varies little from the grey values 99 or 101. Therefore, changing the LSB creates an imperceptible change in the image.

Algorithm:

1) Reading the input image and secret message :

At first, four input images are used as input images ,all of them are (Lena or goldhill) image and all with size of 512×512. Each one saved by a different parameter (Cover1,Cover2,Cover3,Cover4) ,each one of these images is gray scale which is an 8-bit image. Which acts as a cover images. The secret message is with length 500 character is divided into four parts even part has 125 character to be inserted in one image.

2) Applying Sobel edge detection on gray scale image:
The Sobel edge detection algorithm is applied to the gray scale input images (Lena and goldhill) to obtain edges in the images. Sobel operator is used here to detect the edges which are described further in this paper. The edge detected image is a binary image consisting of '1's at edge pixels and '0's at non-edge pixels. Detected the edges by 3x3 scanning window and store these edge pixels in an array (R).

3) Converting the secret message into binary string:
The secret message is first converted to its equivalent ASCII code. This ASCII code is converted into the binary string, i.e. in the form of '1's and '0's.

3) Calculating the total secret message length: The total of the secret message length is calculated to be useful to the receiver for extract the secret message. Where the size of secret image pixel must be hide in the last 12 pixels in the orginal images.

4) Embedding the secret image: The bits of secret images for each block are inserted in the LSBs of edge pixels , where each block has put in the one image . in the edge pixel of the edge detected image that has edge magnitude value for it more than threshold (Gr> threshold).

5) Writing stego-image : The output is four images containing secret message (four parts each part represent as a independent message).

5.2 LEAST SIGNIFICANT BIT EXTRACTION

Algorithm:

1) **Reading the Stego-Image:** The four stego-images generated at the output of the embedding procedure which is a gray scale image is read.
2) **Detecting edges in the stego image:** The Sobel operator method is applied to detect edges in the stego-images. Thus, getting the edge and non-edge pixels.
3) **Extracting size of secret message bits:** The total number of the secret message length bits is extracted from the last 12 pixels in the stego_ images.
4) **Extracting the secret message:** The secret message is extracted from edge pixels using LSB Technique
5) **Recovering of secret image:** Retrieving bits from stego-images and Convert the extracted binary string to an ASCII value. This ASCII code is converted to its corresponding character to form the complete message.

6. EXPERIMENTAL RESULTS

Experimental results using the algorithm described in proposed work section have been applied on standard (Lena) images of size 512x512 pixels are presented in this section. Two parameters measurements are applied in the presented work that is:

1. **Mean square error (MSE)** of an estimator is to quantify the difference between an estimator and the true value of the quantity being estimated [10].

$$MSE=\frac{1}{MN}\sum_{I=0}^{M-1}\sum_{J=0}^{N-1}\big(x(i,j)-y(i,j)\big)^2 \quad ..(5)$$

Where:
i, j: refer to the pixels positions in the image.
M,N: refer to the number of rows and columns in the input image, respectively.

2. **Peak Signal to Noise Ratio (PSNR):** The PSNR ratio is often used as a quality measurement between the original and reconstructed image. The higher the PSNR, the better is the quality of the compressed or reconstructed image [10]. The PSNR is
Defined as:

$$\text{PSNR} = 10 \log \frac{(R^2)}{\text{MSE}} \quad \ldots(6)$$

Where: R is the maximum pixel value in the input image data type.

Figure (5) shows original image of Lena. and figure (6) shows the edge detected of the Lena image and figure (7) shows the encoded images of Lena with secret message. Figure (8) shows original image of goldhill. and figure (9) shows the edge detected of the goldhill image and figure (10) shows the encoded images of goldhill with secret message. Results for proposed method are shown in table 1.

Figure 5 : Original image of Lena

Figure 6 : Edge detected image of Lena

Figure 7 : Encoded images of Lena with secret message

Figure 8 : Original image of goldhill

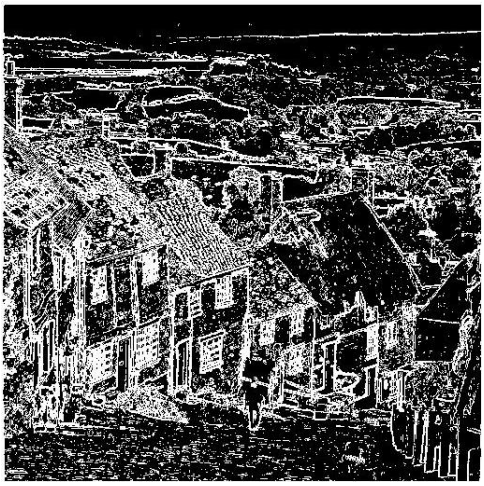

Figure 9 : Edge detected image of goldhill

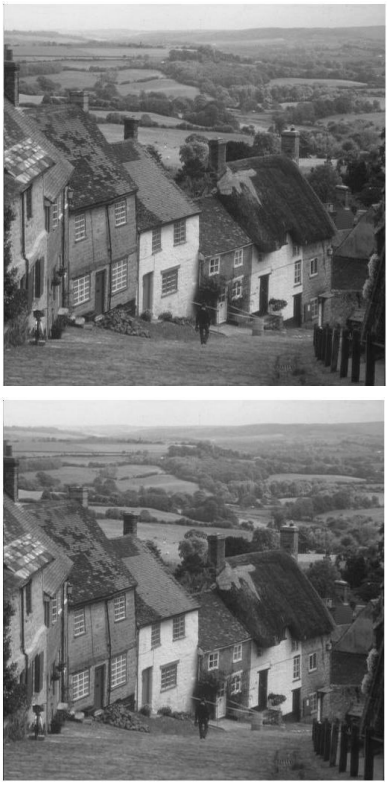

Figure (10):Encoded images of goldhill with secret message

Table 1: Proposed method results

Image's type	PSNR	MSE
Cover1(Lena)	81.2356	4.892349243164062
Cover2(Lena)	81.2356	4.892349243164062
Cover3(Lena)	81.1684	4.968643188476562
Cover4(Lena)	81.3125	4.8065185546875
Cover1(goldhill)	81.7187	4.377365112304688
Cover2(goldhil)	81.7567	4.339218139648438
Cover3(goldhill)	81.8143	4.281997680664063
Cover4(goldhill)	81.8047	4.291534423828125

7. CONCLUSION

The proposed method using the edges of image for hidden data . Advantage of edge detection technique is to be taken to increase capacity. because editing in edge areas cannot be detected well by human eye, but editing in smooth areas can be detected easily. Experimental results shows that the proposed work is successful in not only achieving a high embedding payload but also in obtaining a stego image of satisfactory quality. Also noticed that the number of hidden characters must be lower than the number of edge detected points due to the corruption of the extracted text after extraction. The maximum number of hidden characters in an image differs from one image to another depending on the number of edge detected points in used image, knowing that the number if edge detected points varies depending on the type of image used i.e.(low frequency image ,high frequency) and the used edge detection algorithm i.e.(Canny ,Sobel ,Robert …etc.).

The future work of this work : dealing with gray scale images is suggested with high dimension to avoid distortion to hide maximum number of character. Also using other operator to find edge detection like (Robert ,prewitt , canny …etc) for embedding secret message and draw comparison between them.

8. ACKNOWLEDGEMENTS

I wish to express my heartfelt gratitude to Dr. Nasseer M. Basheer and Dr. Ashty M. Aaref for bestowing on me the opportunity to undergo my thesis work under his guidance. I am also thankful to the authors whose works I have consulted and quoted in this work. Also special thanks to my family. Words cannot express how grateful I am to my mother, and father.

9. REFERENCES

[1] Frank Y. Shih, *Image Processing And Pattern Recognition Fundamentals And Techniques*, Published by John Wiley & Sons, Inc., Hoboken, New Jersey. ISBN 978-0-470-40461-4, 2010.

[2] Jamal A. Othman, "Steganographic scheme to avoid statistical Steganalysis", J. Of College Of Education For Women , vol. 25, pp. 249 -256, 2014.

[3] Neil F. Johnson and Sushil Jajodia ," Steganalysis: The Investigation of Hidden Information", IEEE, pp. 113-116, 1998.

[4] W, Peter, *Disappearing Cryptography: Information Hiding: Steganography & Watermarking* ,3rd edition. San Francisco: Morgan Kaufmann, ISBN 978-0-12-374479-1, 2009.

[5] Saiful Islam, Mangat R Modi and Phalguni Gupta, "Edge-based image steganography", Islam et al. EURASIP Journal on Information Security, pp. 2-14 2014.

[6] Nasseer M. Basheer, Ashty M. Aaref, Dhafer J. Ayyed ," Digital Image Sobel Edge Detection Using FPGA", International Journal of Advanced Research in Computer Science and Software Engineering , vol. 5, pp. 183-190, July 2015.

[7] James Clerk Maxwell, *Digital Image Processing Mathematical and Computational Methods*, Horwood Publishing, vol: ISBN:1-898563-49-7, 2005.

[8] Rashmi , Mukesh Kumar, and Rohini Saxena, "Algorithm And Technique On Various Edge Detection: A Survey ", Signal & Image Processing: An International Journal (SIPIJ), vol.. 4, pp. 65-75, June 2013.

[9] Fisher, R.; S. W. A. Perkin, and E. Wolfart, *Image Processing Learning Resources*, HIPR2, Explore with JAVA, 2000.

[10] The MathWorks, *Computer Vision System Toolbox™ Reference*, by MathWorks, Inc., 2013.