Hiding Image in Image using Modified method in Least Significant Bit (LSB) substitution based Sobel Edge Detection

Dr. Ashty M. Aaref

Software Engineering Department College of Technology / Kirkuk, Iraq

ABSTRACT

In the last decade, transfer of multimedia data on the internet has been developed quickly. Interaction using digital media has become easier and convenient to people as compared to traditional methods. In this paper we hiding our secret image in the edge of cover image without changing the details of image. This work shows how the edges of the images will be detected by scanning method using 3x3 window and then the secret binary image is canceled in edges of a gray scale images (Lena) 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 512×512 is used as a cover image and 32×32 binary image as a secret image in this work.

Keywords: Information Hiding, Steganography, Cover image.

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.
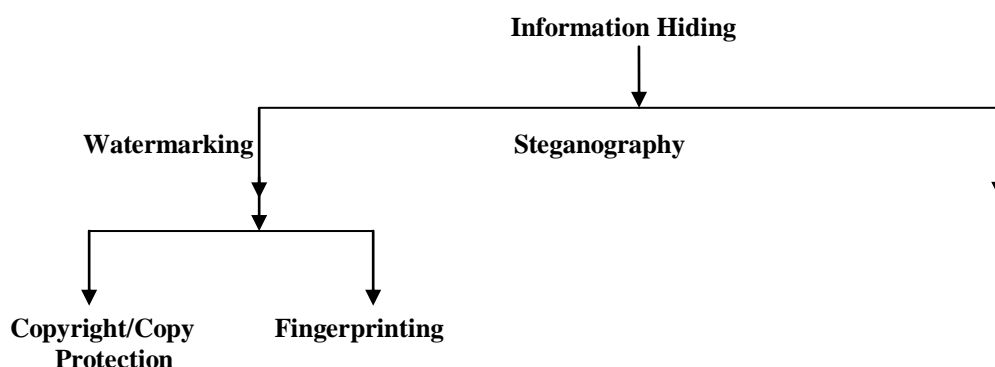
Figure (1): bnformation Hiding

2. STEGANOGRAPHY

Since the dawn of human communication, there has been a need to protect messages that travel between multiple parties. One of the earliest forms of protection is known as steganography. Digital steganography is the science of hiding secret messages within digital media, such as digital images, audio files, or video files [1]. The purpose of steganography is covert communication to hide a message from a third party. The different parameters used to evaluate the process of Steganography are capacity, robustness and imperceptibility [2].

A steganography system consists of three elements: 1) cover message (is the carrier of the message such as image, audio, text, video or some other digital media), 2) the secret message (is the information which needs to be concealed in the suitable digital media),and 3) the stego-message (which is the cover message with message embedded inside it). The embedding algorithm is the way used to embed the secret information in the cover message. The secret key is used to embed the message depending on the hiding algorithms [3]. A digital image a [m, n] described in 2D discrete space is derived from analog images a(x, y) in a 2D continuous space through a sampling process that is frequently referred to as digitization [4]. These digital images can be monochrome (bi-tone), grayscale or color depending upon the permissible intensity levels of each pixel i.e. whether each pixel is represented by only one bit, 8-bits or 24-bits. Generally speaking, a monochrome image can have only one bit plane whereas there are 8-bit planes in a gray scale image and 24 bit planes (8-bits each, with respect to the 3 channels R, G and B) in a color image. Figure (2) shows the bit planes of a 7×8 pixel gray scale image. The least significant bit plane (LSB plane) is the plane that consists of bits with minimum positional value ($2^0 = 1$) and the MSB plane (most significant bit plane) consists of bits with highest positional values i.e. $2^7 = 128$.
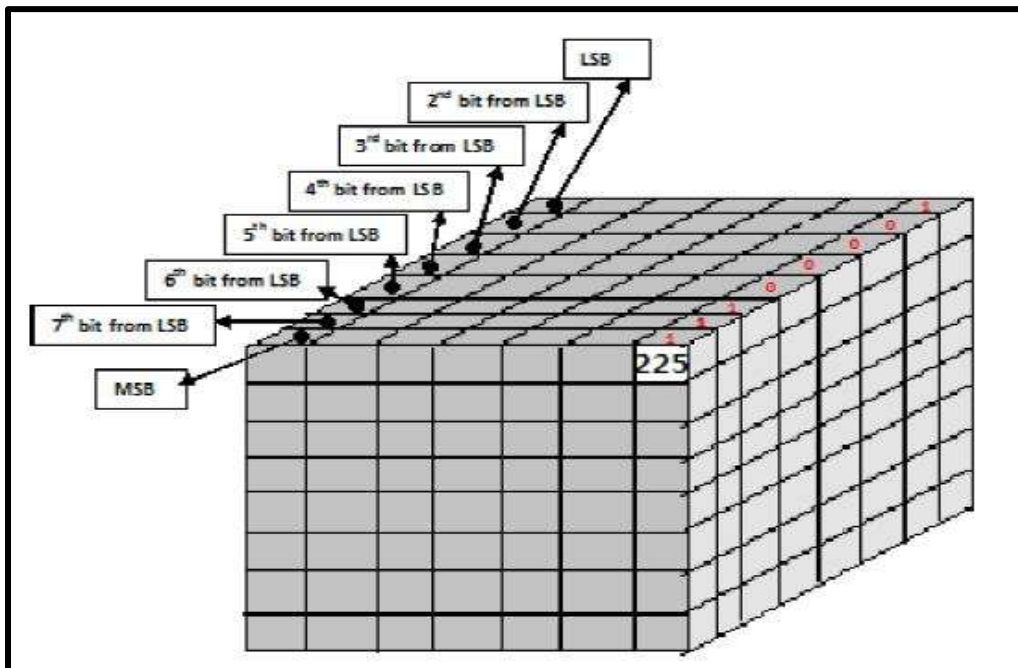

Figure (2): Bit planes of a 7×8 pixel gray scale image [5]

Embedding data, which is to be hidden, into an image requires two files. The first is the image (that will hold the hidden information), called the cover image. The second file is the message (the information to be hidden). A message may be plain text, ciphertext, other images, or anything that can be embedded in a bit stream. When combined, the cover image and the embedded message make a stego-image. A stego-key (a type of password) may also be used to hide, then later decode, the message.

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 [6].

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 [7]:

$$Gi=Gx*F(i,j) \qquad and \qquad Gj=Gy*F(i,j)$$

Sobel masks are showing in Figure (3).

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

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

Figure (3): Horizontal operator and Vertical operator [7]

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

$$Gx = \{f(x+1, y-1) + 2f(x+1, y) + f(x+1, y+1)\}$$
$$- \begin{cases} f(x-1, y-1) + 2f(x-1, y) + \\ f(x-1, y-1) \end{cases} \qquad \dots(1)$$

$$Gy = \{f(x-1, y-1) + 2f(x, y-1) + f(x+1, y-1)\}$$
$$- \begin{cases} f(x-1, y+1) + 2f(x, y+1) + \\ f(x+1, y+1) \end{cases} \qquad \dots(2)$$

These masks can then be combined together to find the absolute magnitude of the gradient at each point. The gradient magnitude is given by [9]:

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

The choice of Sobel edge detection operator is motivated by the fact that they incorporate both the edge detection as well as smoothing operator so that they have good edge detection capability in noisy conditions [10].

4. THRESHOLDING

Thresholding is a relatively simple approach of image segmentation [11]. Thresholding becomes a simple but effective tool to separate objects from the background [12]. 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 [13]:

$$f(x, y) = \begin{cases} 1 \ if \ f(x, y) \geq T \\ 0 \ other \ wise \end{cases} \qquad \dots(4)$$

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:

$$T = \frac{1}{H*W} \int_{i=1}^{H} \int_{j=1}^{W} f(i, j) \qquad \dots\dots (5)$$
H=high of image.   W=width of image.

5. PROPOSED WORK

A- 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:

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

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) Read the input image and secret image :
At first, four input images are used as input images ,all of them are (Lena) 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 image is a binary image with size of (64×64) where it divides into four blocks each block has size (32*32).

2) Apply Sobel edge detection on gray scale image:
The Sobel edge detection algorithm is applied to the gray scale input images (Lena) 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) Calculate the secret image size: Calculate the size of the secret image to be useful to the receiver for extract the secret image. Where the size of secret image pixel must be hide in the last 16 pixels in Lena images because the size of secret image consist of number of rows(8 bits) and number of columns (8 bits) so that the first 8 pixels from last 16 pixels of Lena images hide number of rows (8 bits in 8 pixels) then the second pixels of last 16 pixels of Lena images hide number of columns(8 bits in 8 pixels).

4) Embed 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 plus key values (Gr> threshold+20).

5) Write stego-image : the output is four images containing secret images (four blocks each block represent as a independent image).

B- Least Significant bit Extraction

Algorithm:

1) Read the Stego Image: Read the four stego-images generated at the output of the embedding procedure which is a gray scale image.

2) Detect edges in the stego image: The Sobel operator method is applied to detect edges in the stego-images. Thus, we get the edge and non-edge pixels.

3) Extract size of secret image bits: Extract size number of secret image bits stored in the last 16 pixel in sego-images.

4) Extract the secret image: Extract the secret image from edge pixels using LSB Technique

5) Recovery of secret image: Retrieve bits from stego-image and show the secret image.

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 [16].

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

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 [16]. The PSNR is Defined as:

$$PSNR = 10 \log \frac{(R^2)}{MSE} \qquad …(6)$$

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

Figure (4) shows original image of Lena. and figure (5) shows the edge detected of the Lena image and figure (6) shows the encoded images of Lena with secret image and figure (7) shows the secret image .Results for proposed method are shown in table 1.

Figure (4) Original image of Lena

Figure (5) Edge detected image of Lena

Figure (6): Encoded image of Lena with secret image

Figure (6): The secret image

Table 1: Proposed method results

Image's type	PSNR	MSE
Cover1	80.7628	5.45501708984375
Cover2	80.7628	5.45501708984375
Cover3	81.2441	4.8828125
Cover4	80.9490	5.22613525390625

CONCLUSION

The results of proposed method and First Component Alteration Technique are compared in the Table 1.Experimental results show that the proposed work is successful in not only achieving a high embedding capacity but also in obtaining a stego-image of satisfactory quality. Also noticed that When increasing the size of input images or decrease secret image size then the least distortion.

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]   Alam S,Kumar V,Siddiqui and W.A,Ahmad M, "Key Dependent Image Steganography Using Edge Detection", IEEE, Fourth International Conference on Advanced Computing & Communication Technologies (ACCT),Rohtak, 8-9 Feb. 2014.
[3]   W, Peter," Disappearing Cryptography: Information Hiding: Steganography & Watermarking ",(third edition). San Francisco: Morgan Kaufmann, ISBN 978-0-12-374479-1, (2009).
[4]   Young. Ian Theodore et al, "Fundamentals of Image processing", version 2.2, 1998.
[5]   Minati Mishra, Priyadarsini Mishra and Flt. Lt. Dr. M.C. Adhikary," Digital Image Data Hiding Techniques: A Comparative Study", ANSVESA, 7(2), pp(105-115), 2012.
[6]   Saiful Islam, Mangat R Modi and Phalguni Gupta, "Edge-based image steganography",Islam et al. EURASIP Journal on Information Security 2014, 2014:8
[7]   Rafael C. Gonzalez, R.E. Woods, "Digital Image Processing", third edition,ISBN :013168728X Publisher: Prentice Hall; 3 edition (August 31, 2007).
[8]   Dr. Abdulsattar M. Khidhir and Nawal Younis Abdullah," FPGA Based Edge Detection Using Modified Sobel Filter", International Journal for Research and Development in Engineering (IJRDE), Vol.2: Issue.1, pp(22-32), 2013.
[9]   Rajesh Mehra and Rupinder Verma , "Area Efficient FPGA Implementation of Sobel Edge Detector for Image Processing Applications", International Journal of Computer Applications, Vol.5 (16) , pp(7 −11), 2012,
[10]   Dhanabal R,Bharathi V And S.Kartika , " Digital Image Processing Using Sobel Edge Detection Algorithm In FPGA", Journal of Theoretical and Applied Information Technology, Vol. 58 No.1, 2013.
[11]   James Clerk Maxwell, "Digital Image Processing Mathematical and Computational Methods", Horwood Publishing, vol: ISBN:1-898563-49-7, 2005.
[12]   Rashmi , Mukesh Kumar, and Rohini Saxena, "Algorithm And Technique On   Various Edge Detection: A Survey ", Signal & Image Processing: An International Journal (SIPIJ) Vol.4, No.3, pp (65-75), June 2013.
[13]   Fisher, R.; S. W. A. Perkin, and E. Wolfart, "Image Processing Learning Resources", HIPR2, Explore with JAVA, 2000.