Diyala Journal of Engineering Sciences, Vol. 11, No. 2, June 2018, pages 67-73 ISSN 1999-8716
DOI: 10.24237/djes.2018.11209 eISSN 2616-6909

Video Steganography Using LSB Substitution and Sobel Edge Detection

Ashty M. Aaref
Department of Software Engineering, Technical College of Kirkuk, Northern Technical University

Abstract

The procedure that involves the in closure of information without altering its intuitive standard is called data embedding. In this paper the secret message (English text) is hidden in the edge of the frames of the .AVI video without changing the details of frames. MATLAB R2013a is utilized to execute this algorithm. The secret message was embedded in the frames 38,39,40,41 and 42 and the reason of selecting those frames is that these frames have sufficient edge point details in them. High embedding and superior quality of encoded secret messages have been accomplished by this design. Additionally, in this project the cover frame image is represented by a 120 frames size 120* 160, and the secret message has been represented with a message comprised of 300 characters Both Peak Signal to Noise Ratio (PSNR) and the Mean Square Error (MSE) have been taken into consideration while analyzing the suggested method as we calculated both PSNR and MSE between the cover frames and the embedded ones. The results obtained were objectively good as the PSNR value ranges from 74.5293dB to 75.9123 dB.

Keywords: EdgeDetection; Video; Edges; mask; threshold; Sobel Operator; Frames.

Paper History: Received: (5/1/2017), Accepted: (20/2/2017)

1. Introduction

Concealing exclusive messages or specific details via a transporter using a hidden method can be called Steganography. This has been derived from two different terms which as steganos, and the graphy [1]. The first one means being secret and not be shared with any (individual), and the second one means to be done by writing or drawing [1]. A text, image, video or an audio file can simply represents the cover medium which is meant to the channel where the data is going be hidden to be private or secret. The superfluous bits in the cover media and placing the secret data into the space can always be substituted by the stego algorithm. The more superfluous bits of superior standard of videos and sounds are available for being hiding. Moreover, there are many applications of steganography such as military, industrial sectors, copyright and intellectual property rights (IPR). This kind of techniques which meant by steganography can be used to send and receive the messages safely [2].It is important to highlight that the idea of this method was basically to hide any details or information throughout image files. However, the recent studies propose using video files as well in hiding the crucial messages [3]. This will add more security to the data needed to be private and not to be shared with any because of the sophisticated structure of the video files which will preserve the data from any external attack or the hackers [4]. The classification of the video in term of stenographic tactics is temporal domain and spatial domain. FFT, DCT or DWT are utilized to transform messaged in the frequency domain. Then the whole or some of the transformed coefficients will be used to include the transformed messages. The LSB tactic is suggested to be utilized in spatial domain in this paper, along with AVI video files has been provided as a concealed and cover medium has been suggested to be used in this paper to present the algorithm. The reasons of using LSB technique in this paper are:

1. There is less chance for degradation of the original image.

2. Hiding capacity is more i.e. more information can be stored in an image.

In [5] Raid R. Omar Al-Nima and Sarah B. Ali Al-Nima used video file type (move) to hide an English text. This method offered high accuracy and secure for data transitions. Experimental results demonstrated success of hiding process. In [6] Uma Sahu et al is used a combination of cryptography and steganography for data hiding in video clips. Random frame selection, pixel swapping and encryption of message have been done to enhance the security of the secret information which goes under the cover of video clips. The method is also able to accommodate large amount of data in video. In [7] P. R. Deshmukh and Bhagyashri Rahangdale used technique will be applied to the AVI file. The Proposed a technique is Hash based least significant bit technique for video steganography. Least Significant Bit insertion method embeds data in the lower bits of RGB pixel of video and these changes will be minimal. Data hiding is the process of embedding information in a video without changing its perceptual quality and also keeps away from knowledge of existence of message. A hash function is used to select the position of insertion in LSB bits.

2. Sobel Edge Detection Operator

Sobel Edge Detection is covered two kinds of masks which simply distinguish between the horizontal edge and the vertical edge. These masks are shown in Figure 1. The calculation of gradient in both horizontal and vertical tends are highly influenced by both masks. The gradient in i and j directions are wrapped around with featureless image and represent these Sobel masks, all are given by [8]:

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

Equation 1 shows convolution of input image with horizontal mask and Equation 2 shows Convolution of image with vertical mask [9].

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

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

These masks can then be combined together to find the absolute magnitude of the gradient at each point. The gradient magnitude is given by [10]:

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

There are several reasons behind choosing the Sobe edge. One of them has been exposed by which is the exact direction of the edge. Likewise, the operator which is taken smoothly. These two points are together to expose the edge direction and then minimize the imprecision in any conditions [11].

3. Thresholding

Splitting images into different segments is called Thresholding [12]. Thresholding is a crucial and effective method which can be used to segment any object from the background. Selecting a threshold T can be taken from extracting any object from the background. Likewise, the object point can be calculated from the image's points (x,y) where f(x,y)>T. It is important to mention that this point will be then called a background [13]. The splitting image can be represented as below:

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

Extracting the threshold in this paper has been done by using the simplest method (Mean image data values) is calculated as follows:

$$T=\frac{1}{H*W}\sum_{i=1}^{H}\sum_{j=1}^{W}f(i,j) \qquad (5)$$

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

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

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

Figure 1: Horizontal operator and Vertical operator [8]

4. Proposed Work

The flow diagram is shown in Figure 2 represent Encoding (steganography) and Decoding (De_ steganography).

The flow diagram of encoding steganography above is explained in detail in the following points:

1. Reading Cover video (AVI) file: The original video (AVI) file that is used in this paper is read. Then the video file is transform into number of frames all with size of (120×160) frame. These frame are of different frequency and convergent. We consider each frame as an image. Here we set the counter value to frames.

2. Selecting Cover frame: After converting the video file to the frames we will take the frames form (38 to 48) to embed the secret message then we have 10 frames for embedding the secret message. While other frame is kept without embedding.

3. Detecting edge for the cover frames : This stage is done by the following point:

a. Applying the convolution mask i and j on the input frame: a standard convolution operation is implemented by sliding a window of odd size (3x3 window) for the horizontal or vertical mask over an frame at (8-bit pixel in the gray scale made frame window), where each pixel in input frame window is multiplied by the contrast pixel value in the mask. When all the pixels values for frame window with the mask window are multiplied, those multiplication results are added together. The resultant value from addition operation is the pixel value that is stored in Gx variable if horizontal mask or in Gy variable if vertical mask is used. The result matrix after this operation is got the identical magnitude of gradients Matrix Gx and Gy as the original frame.

b. Determining the gradient magnitude (Gr): The gradient magnitude is calculated by computing Equation 3 and determined by squaring the pixels values of each filtered image (Gx and Gy), Then the two results are added and their root is computed to get the total gradient value (Gr).

c. Comparing the Gradient Magnitude with threshold value and find true edges: The edges can be exposed by putting into the threshold to the total gradient (Gr). The identification of the pixel as an edge, completely depends on the (Gr) whether it greater than the threshold. Else it's not identified as an edge as calculated in Equation 4.

4. Hiding video frame by frame: The text data can then transformed into the binary. Acquiring the ASCII code from each character and then transforming those ASCII code into binary helps in getting the Binary format. The LSB bit of the frame pixel is replaced by the binary data. During the comparing operation between the gradient magnitude (Gr) and the threshold value the pixels are determined that will hiding the secret message in it according (the pixels that have values greater than the threshold). This encoded frame called as stego video is ready for transmission through the internet.

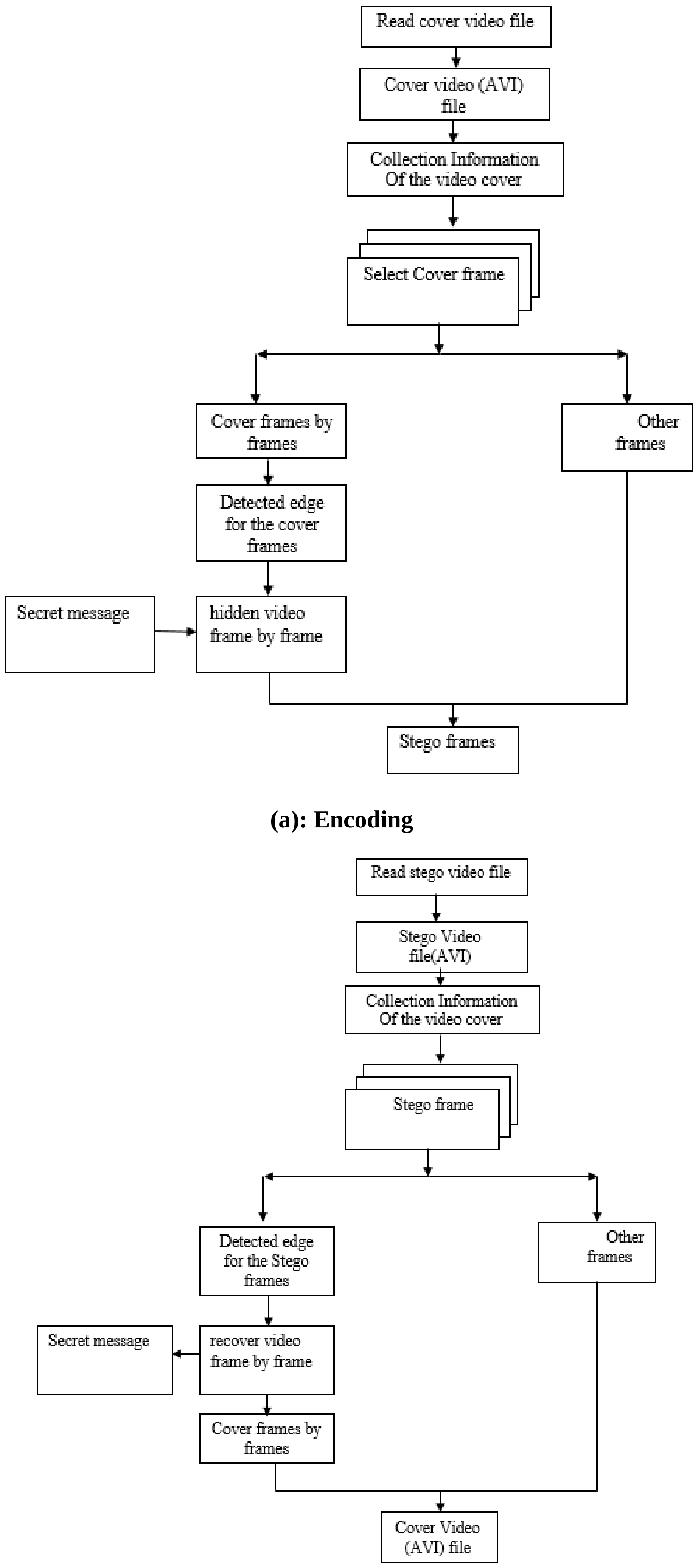
The flow diagram of decoding steganography above is explained in detail in the following points:

1. Reading stego video (AVI) file: The stego-fram can be produced from the resulting of the inclusive procedure then it will be read by the receiver.

2. Selecting Cover frame : After converting the video file to the frames we will take the frames form (38 to 48) to extract the secret message then we have 10 frames for extracting the secret message. While other frame is kept without extracting.

3. Detecting edge for the stego–frames: Exposing the edges in the stego-frames can be acquired by using the Sobel operator method. Hence, all the edges can be acquired, even the non-edge pixels.

4. Recovering video frame by frame: LSB method is used to acquire the private messages from edge pixels. Then any binary string will be transformed to an ASCII code, and by that will transform according to the related messaged from the whole message.



(a): Encoding

(b) Decoding

Figure 2: Steganography System

Least Significant Bit Embedding

Substituting the LSB of the pixel (f(I,j)) with the message bit one after another is the inclusive technique used in this algorithm. Therefore, the number of the pixels to be dealt with by the number of bits which is equal to the message. Additionally, any non-important message will be substituted by number of message bits. The equation below explains clearly the embedding process:

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

The color of the pixel can only be defined by the ingress of the matrix which always should be a positive integer, which is the value of the pixel. This can be extracted from the p-by-q image which is simply a p-by-q matrix. The pixels estimate is from 0 to 2n-1 for each n-bit-image. This can be clarified by the fact that the n-bit image's color can be extracted from n which represents the string length. However, the decimal exemplification will be used if there is a persistent need of using the bit string exemplification. In this project, a grayscale image has been utilized from 8-bit, and there are belong to p-by-q matrices of integers. The value of these integers is varied from 0 to 255. The 0 value appear to the black color and the 255 appear to the white color, the in between value appear to the shades of the different gray (the more close to 0 the darker shades, the more close to 255 the more lighter) The least significant bit (LSB) is the bit corresponding to 20, that is, the bit that makes a value even or odd. Since the slight change in the values does not necessarily show a different or change in the color, so a slight change in the LSB will only show unnoticeable alter in the color.

Algorithm:

Input: cover video (AVI), hidden video (AVI).

Output: Stego video (AVI).

Step 1: Inputting cover video file or stream.

Step 2: Reading required information of the cover video.

Step 3: Breaking the cover video into frames.

Step 4: Inputting message hidden video file or stream.

Step5: Executing the Sobel using the temporarily frame.

i. Pass the modified frame through filter.

iii. Applying Sobel mask on X-direction.

iv. Applying Sobel mask on Y-direction.

v. Computing gradient magnitude (Gr).

vi. Gradient magnitude (Gr) is compared it with Threshold values.

vii. Obtained edges frame.

Step 6: Reading secret Message: the secret message is read to be inserted into the input frame. The length of the messages that are used (300) character.

Step 7: Breaking secret message into 20 parts. Each one saved by a different parameter (text1, text2, text3, text 4……..text 20).

Step8: Converting the secret message into binary string.

Step 9: Finding edges of pixels of the cover frame. When the edges are detected, the edge pixels and their values greater than (threshold + key) are used for embedding the bits of secret message in their LSBs.

Step 10: saving stego-image: the output is four images containing secret images (four blocks each block represent as an independent image).

Least Significant bit Extraction

Algorithm:

Input**:** Stego video (AVI).

Output: Recover video (AVI).

Step 1: Inputting stego video file or stream.

Step 2: Reading required information from the stego video.

Step 3: Breaking the video into frames.

Step 4: Recovering the message hidden video file or stream.

Step5: Executing the Sobel using the temporarily frame.

i. Pass the modified frame through filter.

iii. Applying Sobel mask on X-direction.

iv. Applying Sobel mask on Y-direction.

v. Computing gradient magnitude (Gr).

vi. Gradient magnitude (Gr) is compared it with Threshold values.

vii. Obtained edges frame.

Step 6: Extracting the secret image: Extract the secret image from edge pixels using LSB Technique.

Step 7: Recovering of secret image: Retrieve bits from stego-frame and show the secret image.

5. Exprimental Results

The standard video (AVI) with the frame of size 120*160 pixels have been used in this work. The below equations clarify the application of the algorithm used by utilizing two parameters measurements:

1. Mean square error (MSE) is to indicate the diversity between an estimator and the estimated true value [13].

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

Where:

i, j: refer to the pixels positions in the frame.

M,N: refer to the number of rows and columns in the frame, respectively.

2. Peak Signal to Noise Ratio (PSNR): is used to illustrate the standard of the measurement between the premier and the restructured image. The quality if an image highly depends on the PNSR, as the higher it is, the good is the quality [13]. The PSNR is defined as:

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

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

The performance of the proposed technique is evaluated using a five frame of (.AVI) video and five parts of secret message (English text).The perceptual imperceptibility of the embedded data is indicated by comparing the cover frame with its stego frame. When we hide secret text in video there will be no loss in quality of video and even no one can guess the presence of data within a video.

The Figure 3 shows variation in PSNR according to capacity of frames video (38,39,40,41 and 42) with all according to the corresponding of the parts of secret message ('hi,my name' , 'is merko ' , 'fromkirkuk' , 'iam15years' , 'student***'). It gives different values of PSNR which deals with quality frame of video.

The Figure 4 shows variation in MSE according to capacity of frames video (38,39,40,41 and 42) with all according to the corresponding of the parts of secret message ('hi,my name' , 'is merko ' , 'fromkirkuk' , 'iam15years' , 'student***'). The two graphs in Figure 3 & Figure 4 show the variation in PSNR and MSE for different frames video depending on parts of secret message.

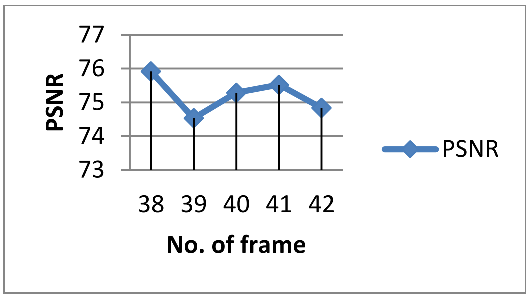


Figure 3: PSNR VS No. of frame of Video

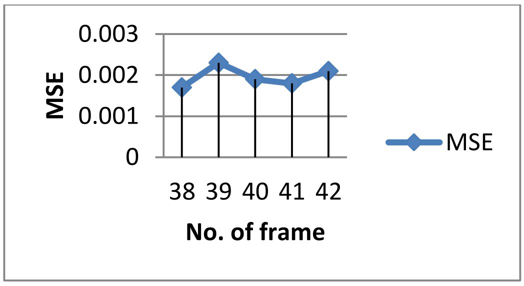


Figure 4: MSE VS No. of frame of Video

Table 1 Optimum Algorithm PSNR in dB is compression vs. Other Publication

#	Algorithm	PSNR in dB
1	Optimum Algorithm (video AVI)	75.9123
2	Raid R. Omar Al-Nima[5] (videomov)	50.7761
3	Uma Sahu et al[6] (video AVI)	66.2903
4	P. R. Deshmukh and Bhagyashri Rahangdale [7] (video AVI)	60.21

6. Conclusion

Despite the fact that the AVI files have been used in this work, other formats can be utilized with a slight practical amendment. Video images can provide a variety of methods in securing and converting any data by concealing messages throughout video images. The future domain seems to be great in the area of Software based Stenographic Engine. Experimental outcomes show the importance of this work not only in the area of accomplishing considerable embedding ability but also in acquiring a stego-image with a great standard. Furthermore, it is noticed that the size of the secret image plays a noticeable role in the resolution. Table 1 shows how this algorithm differs from the work in the literature review.

References:

[1] E. Cole and R.D. Krutz, (2003), Hiding in Plain Sight: Steganography and the Art of Covert Communication, Wiley Publishing, Inc., ISBN 0-471-44449-9.

[2] S.K. beisser and A. P.F. Petitcolas, (1999), Information Hiding Techniques for Steganography and Digital Watermarking, Artech House Books, ISBN 1-58053-035-41999.

[3] D. Stanescu, M. Stratulat, B. Ciubotaru, D.Chiciudean, R. Cioarga and M. Micea, (2007), Embedding Data inVideo Stream using Steganography, in 4th International Symposium on Applied Computational Intelligence and Informatics, SACI-2001, pp. 241-244, IEEE.

[4]. Kousik Dasgupta, J.K. Mandal and Paramartha Dutta, Hash Based Least Significant Bit Technique For Video Steganography(HLSB), International Journal of Security, Privacy and Trust Management (IJSPTM), Vol. 1, No 2, April 2012.

[5]. Raid R. Omar Al-Nima and Sarah B. Ali Al-Nima, Design and Implementation of Steganographic Algorithm on Video File (Mov), third scientific in information technology, 29-30 Nov, 2010.

[6]. Uma Sahu and Saurabh Mitra, A Secure Data Hiding Technique Using Video Steganography, International Journal of Computer Science & Communication Networks, Vol. 5, pp. -357.

[7]. P. R. Deshmukh and Bhagyashri Rahangdale, (2014), Data Hiding using Video Steganography, International Journal of Engineering Research & Technology (IJERT), Vol. 3, pp. 856-860.

[8].Rafael C. Gonzalez, R.E. Woods, Digital Image Processing, third edition,ISBN :013168728X Publisher: Prentice Hall; 3 edition (August 31, 2007).

[9] Abdulsattar M. Khidhir and Nawal Younis Abdullah, (2013), FPGA Based Edge Detection Using Modified Sobel Filter, International Journal for Research and Development in Engineering (IJRDE), Vol.2: Issue.1, pp. 22-32.

[10]. Rajesh Mehra and Rupinder Verma , (2012), Area Efficient FPGA Implementation of Sobel Edge Detector for Image Processing Applications, International Journal of Computer Applications, Vol.5 (16) , pp. 7 –11.

[11]. Dhanabal R,Bharathi V And S.Kartika , (2013), Digital Image Processing Using Sobel Edge Detection Algorithm In FPGA, Journal of Theoretical and Applied Information Technology, Vol. 58 No.1.

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

[13].Nasseer M. Basheer, Ashty M. Aaref, Dhafer J. Ayyed, (2015), Digital Image Sobel Edge Detection Using FPGA, International Journal of Advanced Research in Computer Science and Software Engineering, Vol 5, Issue 7.