International Conference on Communication and Information Processing (ICCIP-2019)
Available on: Elsevier-SSRN

Video Steganography using Edge Detection Techniques

Dipika Deshmukh[a],Dr.Gajanan Kurundkar [b]

[a]*Research Scholar,SRTMU,Nanded 431606,India*
[b]*Research Guide,GuruBuddhiswami College,Purna 431511,India*

Abstract

For transmitting secret information, security is very important because hackers may utilize weak link over communique network to take desired information. Video Steganography is the process of hiding some secret information inside a video. The addition of this information to the video is not perceptible by the human eye as the change of a pixel color is minor. In this paper, I have designed new algorithm to hide the data into video. Hidden data has been taken as text and hide it into frames of video.

Keywords- Canny, Edge Detection, Stego

1. Introduction

Digital video is a set of frames**,** which played at fixed frame rate. Frame rate depends on video standard. Digital Video Quality depends on parameters like the fps (frames per seconds), the number of pixels in a frame and frame size. The fps parameter standard is general video formats, its value lies between 24 and 30 fps but the other two parameters, the number of pixels in a frame and frame size present a number of improved from one video standard to another. Every image in a video called a frame which holds number of pixels having three or four color combinations like RGB (Red, Green, Blue) or CMYK (Cyan, Magenta, Yellow, Black). The remaining mediator colors are composed from a mixture of these primary colors. Because the human eye is mainly sensitive to green color, in few video standards the number of bits of every color combination may vary. In 24-bit RGB color standard, each red, green, and blue color containing 8 bits in length and has 256 alternatives in color density. On the other hand 32-bit CMYK color standard is required and this standard is generally used in modern computer displays.

2. Proposed Work

We have studied that there is lots of limitations in previous algorithms which are not sufficient for video steganography process. In previous research works sequential encoding is a good encryption method for getting the embedding process of video steganography but still it has some limitations as by using sequential

encoding, intruders can recognize the presence of hidden message by sequentially analyzing the video frames. So this technique is not much secure for steganography process. To preserve the security of secret information key based random frame selection algorithms are followed in our proposed work. For embedding process, LSB technique is used to embed data in video frames. But simple LSB technique is not resist to attacks and produces the poor value of peak signal to noise ratio and high value of mean square errors which are both the image quality parameters, so to avoid this problem we will use an improved techniques for hiding the data. Hiding in LSB bits change the resolution of pixels which may be easily observable to human eye during detection mechanisms and also they are very easy to attack by attackers. But proposed technique can embed large amount of data than LSB method. In proposed techniques, edge detection is carried out by different edge detectors.

Each operator follows different steps for edge detection as follows

2.1 Edge detection steps for Gradient based operator

1. Generate all the filters
2. Apply mean filter on gmap
3. Calculate Magnitude
4. Apply thresholds on gmap

2.2 Edge detection steps for LoG operator

1. Apply LoG filter
2. Check for zero crossings
3. Threshold based on gradient magnitude

2.3 Edge detection Steps for Canny operator

1. Smoothing
2. Compute Gradient
3. Non-maximum Suppression
4. Hysteresis Thresholding

2.4 Procedure for encoding and decoding data at edge pixels

Procedure for hiding data at edge pixels

1. Read and display video
2. Fragmentation of video
3. Select Random frames i. e images
4. Read selected images
5. Extract R, G, and B channels
6. Apply Mean Filter
7. Enhancement of images
8. Apply edge detection methods:

Group-1: Roberts, Prewitt, Sobel
Group-2: LOG
Group-3: Canny

9. Find the edge pixels from each channel and combine
   Following are the TH parameters used for edge detection:
   a. Group-1: TH = 0.2
   b. Group-2: TH = 0.007
   c. Group-3: TH = 0.15
10. Except for LOG, the edge pixels of each channel are combined using logical AND operation. For LOG, it is logical OR operation. After this step, we will have edge pixels location. Payload = total number of edge pixels
11. Read the text message and convert each character to its ASCII value store in an array. For e.g., the ASCII value of character 'a' is 97
12. The maximum number of characters that can be embedded equals the total number of edge pixels (payload). Now you have 2 options: embed the message only once (OR) embed the message multiple times in order to put almost full payload of characters.
    For example:
    Assume the message has 9 characters (each character is 1 byte or 8 bits) and the payload is 100 bytes (meaning 100 pixels, each pixel can store the ASCII value = 1 byte). So if we have to embed at almost full capacity, we will embed the message 11 times or 99 bytes leaving one byte or pixel free).
13. Select edge pixels randomly and store the locations in an array. This array is the 'key' and this has to be provided along with the stego video for decoding.
14. To embed or hide the message into the image, simply copy the ASCII values of each character into the edge pixels of Blue channel corresponding to the locations in the key. This will hide the data at the edge pixels.
15. Calculate different metrics to evaluate the performance of each edge detector.

Explanation:

i. An automatic thresholding methodology (e.g., Otsu's method, K-means etc) should not be used since they tend to give higher THs which will reduce the payload capacity i.e., decrease the number of edge pixels
   I have selected the THs in such a way so as to get maximum number of edge pixels while not compromising the edge i.e., non-edge pixels should not be identified as edge pixels
ii. TH for LOG detector is 0.007 which is much smaller than the THs used for other edge detectors. Also note that logical OR was used to combine edge pixels for LOG as opposed to logical AND used for other edge detectors.
iii. Even at this low TH, canny has higher payload. From this we can conclude that Canny is better than LOG
iv. At similar THs (Canny: 0.15; Sobel, Prewitt, Roberts: 0.2), Canny has higher payload and we can store more data in Canny. From this we can conclude that Canny is better than Roberts, Prewitt and Sobel
v. From 3 and 4, we can assert that canny edge detection shows superior performance in terms of

payload when compared to the other edge detectors

Procedure for decoding

1. Stego video (Stego frames and other frames)
2. Extraction of Stego frames
3. Decoding is not possible without the key. Key in our case is simply an array containing the pixel locations where we store the data.
4. To decode, use the key to go to the location of each pixel where a character has been hidden
5. Read the GL of the blue channel at this pixel. This GL is equivalent to the ASCII value of the character hidden at this pixel.
6. Convert this ASCII value to character and write to a text file
   If you want to hide a new message, must delete the old message and write the new message into notepad file
7. Display the message

3. Conclusion

To preserve the security of secret information, key based random frame detection algorithms are followed in our proposed work. For hiding the data blue channel is selected because blue channel is less perceptive to human eyes. We can hide data at almost full capacity in blue channel. LSB is common method for data hiding. In proposed improved method we can hide more data than LSB technique.

4. References

[1] Kamred ,Udham., Singh ,Int., 2014 " Video Steganography: Text Hiding In Video By LSB Substitution" Journal of Engineering Research and Applications,Open Access, p.105

[2] Shubhashree Savant, 2014 "A Review on Edge Detection Techniques for Image Segmentation" International Journal of Computer Science and Information Technologies, Academy Publisher, Tallinn,p.589

[3] Manasi N Patil, Brijesh Iyer, Rajeev Arya, "Performance evaluation of PCA and ICA algorithm for facial expression recognition application", Fifth International Conference on Soft Computing for Problem Solving, 965-976,2016.

[4] G.T. Shrivakshan, Dr.C. Chandrasekar, 2012, "A Comparison of various Edge Detection Techniques used in Image Processing" IJCSI International Journal of Computer Science , Software First Ltd, Doolar lane,Mahebourg, p.269

[5] Vanitha T, Anjalin D Souza, Rashmi B, Sweeta DSouzal , 2014 "A Review on Steganography – Least Significant Bit Algorithm and Discrete Wavelet Transform Algorithm" International Journal of Innovative Research in Computer and Communication Engineering, International Journal of Scientific and Research Publications jointly India and USA. p.89