ISSN: 0067-2904

Video Steganography Method Based DWT

Asmaa Hasan Mohsen*, Shaimaa Hameed Shaker
Department of Computer Sciences, University of Technology, Baghdad, Iraq.

Abstract

The video steganography is a technique to hide information inside video file.Whereas video Steganography is a very important task in real life where the users want to keep data, so the steganography process used for the secure data transmission from the sender to receiver through the internet. Least significant bit (LSB) insertion technique operates on LSB bit of the media file to hide the information bit. In this paper steganography technique used to hide the information inside compressed video as development of a standard method in order to benefit from the advantages of the compression process, which added to the video, these features are reduce storage size of video, and reduce bandwidth to transfer data in faster way with save time that requires transferring video during the network. The video compressed by using discrete wavelet transform DWT because it an efficient method that can be used to perform an efficient compression technique. The video contains number of frames played over a period of time. By using the proposed method can hide information in any number of these frames for various video sizes and in specific location of the pixels. And has been verified from performance of proposed method by retrieval original information without any lost, and by using several quality measures performance to measure the results of proposed method found it's a good, acceptable compare with results of other methods used in more researches.

Keywords: Compressed video, Steganography, Least significant bit (LSB).

إخفاء نص داخل فيديومضغوط

أسماء حسن محسن، شيماء حميد شاكر
قسم علوم الحاسوب ، الجامعة التكنولوجية، بغداد، العراق.

الخلاصة

ألاخفاء هو تقنية لاخفاء المعلومات داخل ملف الفيديو.حيث إن استخدام الأخفاء السري للبيانات في الفيديوجدامهم في حياتنا عندما يريد المستخدم حفظ بياناته بشكل سري، لذلك نحن نستخدم عملية الاخفاء لتامين نقل البيانات من المرسل الى المتلقي عبر الانترنت. LSB هي تقنية تعمل على اخذ اقل Bit ملف الفيديو لاخفاء bit من المعلومات فيه. في هذا البحث تم استخدام تقنية الاخفاء لاخفاء المعلومات داخل الفيديو المضغوط كتطوير على الطريقةbit ملف الفيديو لاخفاء القياسية وللاستفادة من مميزات عملية الضغط التي تضاف الى الفيديو هذه المميزات هي لنقل البيانات بشكل اسرع مع حفظ الوقت المطلوب لنقل Bandwidth و تقليل المساحة الخزنية وتقليل لانها طريقة فعالة يمكن استخدامها لأداء تقنية DWT الفيديو يضغط بواسطة استخدام الفيديو عبر الشبكة ضغط كفوئة الفيديو يتكون من عدد من اللقطات التي تم التقاطها على مدى فترة من الزمن. بواسطة استخدام الطريقة المقترحة يمكن اخفاء المعلومات بهذه

*Email: asmaa.hasan92@yahoo.com

اللقطات لمختلف احجام الفيديووبتخصيص موقع عناصرالصورة. وقدتم التأكد من اداءالطريقة المقترحة بواسطة استرجاع الرسالةالاصليةوبدون اي فقدان ، وباستخدام العديد من طرق قياس معايير الجودة لقياس نتائج الطريقة المقترحة وجد انها جيدة ومقبولة مقارنة مع نتائج الطرق الاخرى المستخدمة في اكثر البحوث.

Introduction

Video Steganography is the process of embedding text in video such that its existence cannot be detected by Human Visual System (HVS) and only the sender and receiver can it known. Steganography comes from the Greek word ***steganos*** which literally means "covered" and ***graphia*** which means "writing", i.e. covered writing. The most common use of steganography is to hide a file inside another file. Generally, in data hiding, the actual information is not maintained in its original format. The format is converted into an alternative equivalent multimedia files like images, video or audio. Which in turn is being hidden within another object [1].

The huge usage of digital multimedia such as videos via communications, wireless communications, internet, leads to incurable growth of data flow through these media. The problem occurs when traditional text and image based steganography techniques is not plentiful .They are able to carry only small files. So there is a problem, how to get much enough files to hide our message. This becomes a very tedious task for carry large amount of data. Here comes the need to use these videos for hiding text inside it. The use of video as a carrier cover for the secure message to overcame the capacity problem. Information can be hidden in any frame of video. Video has a large capacity to store information [2].

Cryptography was created as a technique for securing the secrecy of communication and many different methods have been developed to encrypt and decrypt data in order to keep the message secret. Unfortunately it is sometimes not enough to keep the contents of a message secret, it may also be necessary to keep the existence of the message secret. The technique used to implement this, is called steganography. The main objective of Steganography is mainly concerned with the protection of contents of the hidden information. Videos are ideal for information hiding. The Least Significant Bit (LSB) embedding technique suggests that data can be hidden in the least significant bits of the cover image and the human eye would be unable to notice the hidden image in the cover file. Video is converted into a number of frames, and then convert each frame in to an image. This technique can be used for hiding letter (in other words the 8 bit) in 8 byte of images [3].

In this paper Section 2 related works, Section 3 the proposed method which contains compression method, embedding secret data inside video, extraction secret data, decompression method, Section 4 describes experimental result and discussion, and Section 5 describes the conclusion.

Related Works

Dey N, Roy A.B, Dey S, 2011, [4], Introduced steganographic technique for the secret color image is hidden in the different sub bands of the cover image's, the cover image and secret image separate into three color planes are R, G and B, each plane of the images is decomposed into four sub bands using DWT, each color plane of the secret image is hidden by alpha blending technique in sub bands of cover image that mean secret image is distributed within the original image depending on the alpha value.

Muhammad K, et al, 2014, [5], they apply steganography technique for hiding data inside the video. The Least Significant bit (LSB) method was used, which is considering a good technique for hiding all letters of the message with pixels of frames of the video in a randomized cyclic manner.

Patil S, Jagtap S.K, 2016, [6], they applied Discrete Wavelet Transform (DWT) to compressed the video during the transfer or download operations over the network. The obtained results reduce both the video size and transfer time. DWT is an efficient method that can be used to perform an efficient compression technique.

The Proposed method

The proposed method is extraction video frames then applied compression method on each frame respectively, and after this applied steganography to hide information in any or number of frames for various video sizes and in specific location of the pixels by using LSB technique. Figure-1 (a) shows a block diagram of sending process. Figure-1 (b) shows a block diagram of receiving processes.

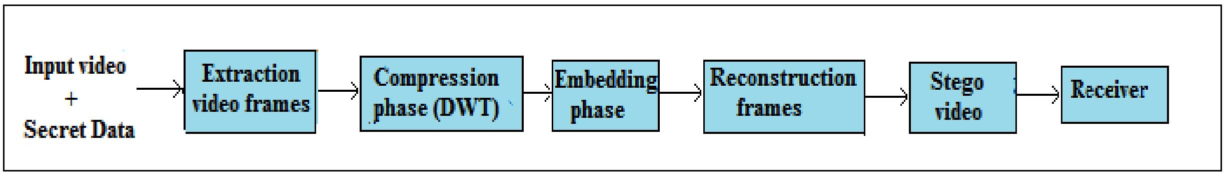


Figure1- (a) Block Diagram of Sending Processes.

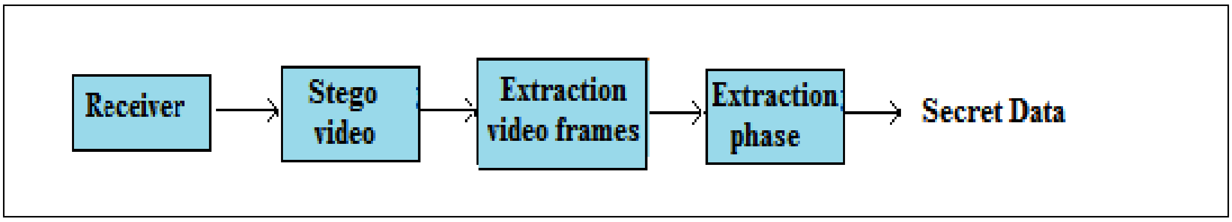


Figure 1- (b) Block Diagram of Receiver Processes.

Extraction video frames phase

The first phase of proposed method is extract video frames to can apply compression method on each frames and to can apply steganography technique, Algorithm (1) explains extraction video frames.

```
Algorithm1: Extraction video frames.
Input:  Digital video.
Output: Frames of video.
Step1: Start.
Step2: Read video, file name= name of video file and path= location of video file.
Step3: Extract frames from video and store in array of one dimension is [N] // N=number of frame.
Step4: End.
```

Compression video phase

The compression phase is very important to reduce the size of video, since the uncompressed videos require a huge space in memory and take huge time to upload or download videos from the internet or during transfer into the receiver. So the video size always affects on the efficiency of video transmission over network, and on storage space on PC hard disk [7].

The video having large areas of redundancies (unnecessary information's).The objective of video compression is to reduce redundancy of the video data in order to be able to store or transmit data in an efficient form. Data compression is achieved when one or more of these redundancies are reduced or eliminated. In lossy compression reduces a file by permanently eliminating certain information, especially redundant information. When the file is uncompressed, only a part of the original information is still there. The Haar wavelet transform with one iteration can be used to perform lossy compression for various video sizes with retains the quality of the compressed video.The Haar algorithm used to compressed video because it is popular and simple transform for data compression and high computation speed [8]. The process of Haar algorithm is deal with frames width and height must be power of 2 such as (128×128, 256×256, 512×512, etc). The proposed system deal with various video sizes such as (640×480),(480×360),(320×240), therefore suggested resize frames of video width and height into power2 to Haar transform an be applied on frame, after this the frame is decomposed on different level using the pyramidal algorithm architecture. The decomposition are the vertical and horizontal directions, and by using the Haar wavelet algorithm can compress the frame to half. The frame is generated by low pass filter (LPF), while the detailed frame are produced using a high pass filter (HPF), the compression process by using Haar wavelet algorithm divided the frame into a series of the frequency bands the low low (LL) band, the low high (LH) band, the high low (HL) band, and the high high (HH) band that shows in Figure -2 [9]. This process apply on each frame of video .The results as compressed frames from compression phase give to embedding phase.

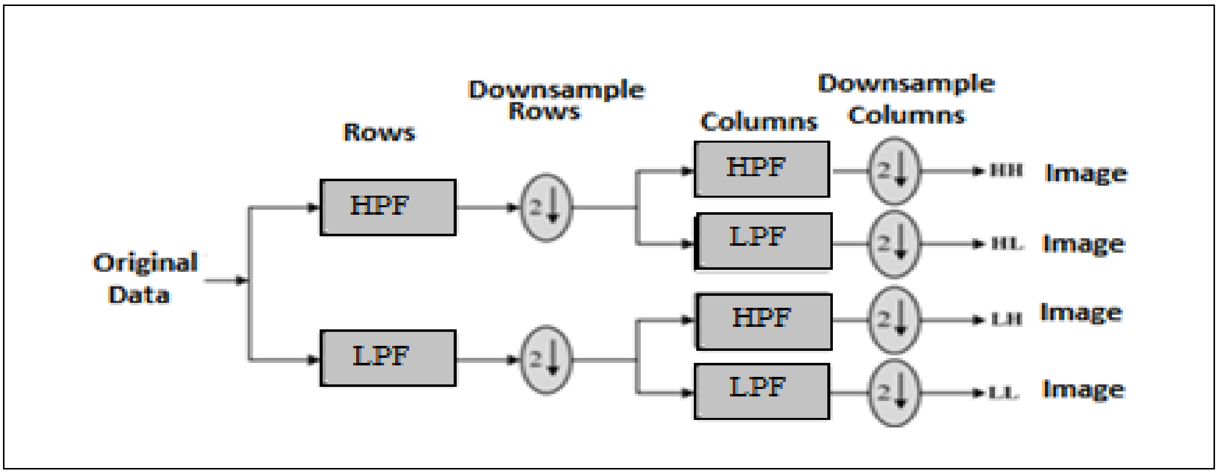


Figure 2- One Stage of Forward DWT [9].

Embedding phase

All the communicating techniques aim to achieve the confidentiality, integrity and authenticity of their secret information. Different approaches are used to cope with these security issues like digital certificate, digital signature and cryptography. But these methods alone cannot be compromised. Steganography is the best solution to these problems as it hides the existence of secret data. The proposed method used video from type (AVI) to hide information then the steganography used is called video steganography. The video considered to be the best cover objects for hiding information because it contains large amount of redundant bits. The most of the methods that hide information by steganography used LSB technique that can hide information by the eighth bit inside frame is changed to a bit of the secret message, for example hide letter A inside image shows below[10].

The letter 'A' has an ASCII code of 65 (decimal), which is 1000001 in binary number. It will need consecutive 7 pixels from frame to store an 'A':

Let's say that the pixels before the insertion are:

10000000.10100100.10110101. 10110101.11110011.10110111. 11100111

Then their values after the insertion of an 'A' will be:

1000000**1**.10100100.1011010**0**, 1011010**0**.1111001**0**.1011011**0**, 11100111

(The values in bold are the ones that were modified by the transformation).

In this paper after extract video frame will separate each color frame to RGB frames to store larger information by use a 24- bit frame, one pixel can store 3 bits by changing a bit of each of the red, green and blue color components, since they are each represented by a byte. When the number 300 the representation of binary number is 100101100 embedded into LSB of this part of the frame. If we overlay these 9 bits over the LSB of the 9 bytes as show below.

Let's say that the pixels before the insertion are:

10010101 00001101 11001001 10010110 00001111 11001011 10011111 00010000 11001011

Then their values after the insertion of 300 will be:

10010101 0000110**0** 1100100**0** 1001011**1** 0000111**0** 11001011 10011111 00010000 1100101**0**

We get the following (where bits in **bold** have been changed). Here the number 300 was embedded into the grid; only the 5 bits needed to be changed according to the embedded message. Finally result is stego frame and by this technique the human will not notice the change between original frame and stego frame.

The proposed method to increase security and development of a standard method will embeded information in any number of frames and in specific location of the frame that means if number of frame is even then hiding information will be in odd location of pixels of the frame, if number of frame is odd then the information will be hiding in even location of pixels of the frame, algorithm (2) and Figure-3 shows this. After hide all secret data inside frames now reconstruction frames as shows in Algorithm (3) to get stego video send to receiver.

Algorithm2: Hidden information inside specific frames of video

Input: Cover frames, Secret data
Output: Stego frames

```
Step1: Start.
Step2: Read letters of secret data
Step3: Find ASCII code of letters and convert into binary number and store  in array called bin
Step4: Read specific location of frames of video from textbox and store it in variable such as i
       // Where i represent number of frame, when send stego video to receiver i send with it.
         If (i mod 2 =  0) then // That means the number of frame is even
         calculate height(x) and width (y) of frame and read each pixel in this frame
         according to location of pixels(x, y).
          For x=0 to height
            For y=0 to width
               P= read pixel of color frame [x, y]
                  If (x mod 2! = 0&y mod 2! = 0) // That means taken the location of pixel is odd only
                     If (n<size of bin[n]) then // Where n represent counter that use to compare with size of
                                              bin array that store secret data in it.
                     Convert P to three pixels are Red, Green, and Blue that give (RGB) pixels to can store
                     3 bit from secret data inside one pixel from specific frame.
                    Else
                    Go to step9
        Else
          If (i mod 2! =  0) then // That means the number of frame is odd
           calculate height(x) and width (y) of frame  and  read  each  pixel in  this  frame  according to
           location of pixels(x, y).
             For x=0 to height
                For y=0 to width
                 P= read pixel of color frame [x, y]
                  If (x mod 2 = 0&y mod 2 = 0) // That means taken the location of pixel is even only.
                    If (n<size of bin[n]) then
                       Convert P to three pixels are Red, Green, and Blue that give (RGB) pixels to can store
                     3 bit from secret data inside one pixel from specific frame.
                    Else
                     Go to step9
Step5: Convert  pixel from Red frame to binary number and store in array called
        binimage and apply steps from (step6 to step8)
Step6: Read bit from (bin)  array by using for loop technique and hide it inside eight bit from array
       of Red (binimage)
        for (k = 0; k < =8; k++)
          if(K= =8) // Where K represent eight bit from binimage.
           binimage [k] =bin[n]
           Increase n// to read next bit of secret data that store in bin array.
          Else
           binimage[k]= binimage[k]// Remain the same values.
Step7:  if(k=8) then  convert values inside binimage that represent  one  byte  into  decimal  number and
       draw as new Red pixel.
Step8: K=0
       binimage="" // To store new values in binimage.
       Repeat (step5 to step8) to hide 2 bits from secret data inside Green and Blue pixel of frame
       Combined Red ,Green, and Blue pixels to give on color pixel is P
       Draw P[x][y] as new pixel of frame
        Repeat from step4 to step8 until hidden all letters of text with pixels of specific location of frame
Step9: Draw another pixels as new pixels of frame because all secret data hiding //That means remain the
                                                                             same values of frames.
        if (n >= size of bin[n])
        Draw p[x][y] as new frame //without apply any process on it.
Step10: Next y
Step11: Next x
Step12: Repeat steps (4 to 11) until hidden all letters of text inside pixels of frames of video if one frame
         not enough to store the text.
Step13: End.
```

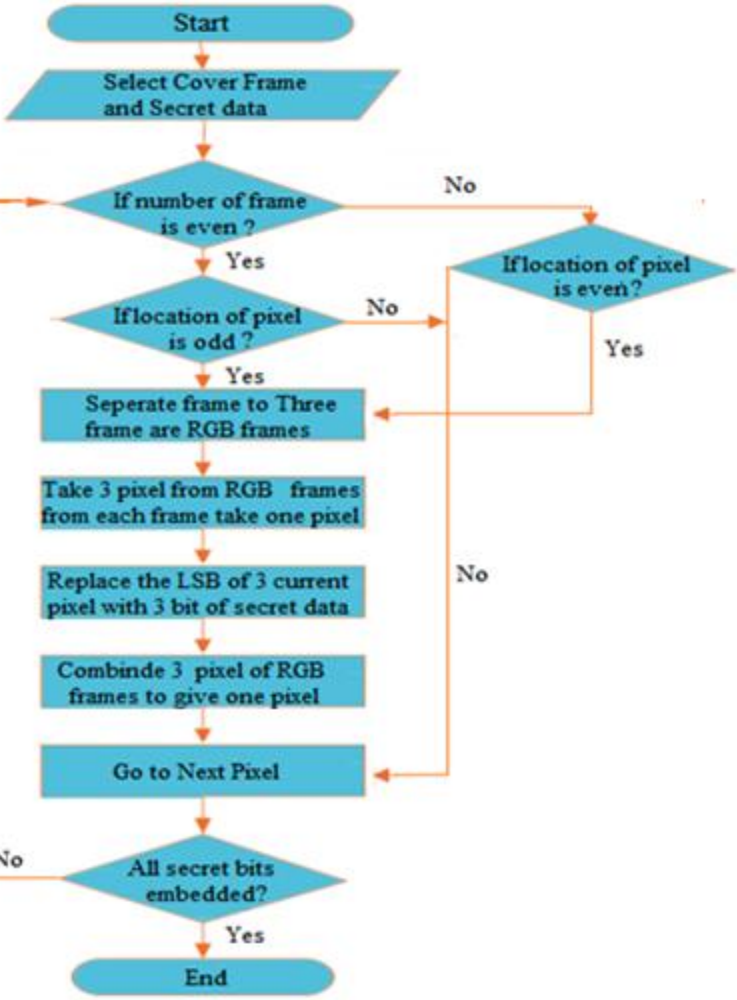

Start
Select Cover Frame and Secret data
If number of frame is even ?
No
Yes
If location of pixel is even?
If location of pixel is odd ?
No
Yes
Yes
Seperate frame to Three frame are RGB frames
Take 3 pixel from RGB frames from each frame take one pixel
Replace the LSB of 3 current pixel with 3 bit of secret data
No
Combinde 3 pixel of RGB frames to give one pixel
Go to Next Pixel
No
All secret bits embedded?
Yes
End

Figure 3-Embedding Algorithm Flowchart.

```
Algorithm(3): Reconstruction frames
Input: stego frames.
Output: Secret video
Step1: Start.
Step2: FM=array of stego frames
Step3: FM rate = 25 // Determine the frame rate in the per-second
Step4: Video write (video name, path).
         For i=1 to N of FM // where N = number of frames from FM array.
Step5: Write video (video, N of FM (i), FM rate) // starting write video with name of video, number of
        frame and frame rate.
Step6: Return (i).
Step7: End.
```

Extraction phase

The first step in extraction phase when the video access to receiver is extraction video frames and this explained in algorithm (1), now extraction secret data from specific frames according to algorithm (4) and Figure-4 that show below.

Algorithm4: Extraction Secret Data

Input: Stego frames, Number of stego frame (i)
Output: Secret data

```
Step1: Start.
Step2: Read frames of video by using for loop technique.
          For j=0 to N// Where N represent last frame of video.
          If (j=i)
            If (i mod 2 = 0) then calculate height(x) and width (y) of frame and read each Pixel  in this
            frame according to location of pixels(x, y).
              For x=0 to height
               For y=0 to width
                P= read pixel of specific frame [x, y]
                   If (x mod 2! = 0&y mod 2! = 0) then  //That means taken the location of pixel is odd only.
                       Convert P to three pixels are Red, Green, and Blue that give (RGB) pixels and read
                        eight bit of each pixel from RGB pixels and store in bin array.
                   Else
                        Go to step5
           Else
            If (i mod 2! =  0) then // That means the number of frame is odd
             calculate height(x) and width (y) of frame and read each pixel in this frame
             according to location of pixels(x, y).
               For x=0 to height
                  For y=0 to width
                   P= read pixel of color frame [x, y]
                    If (x mod 2 = 0&y mod 2 = 0) // That means taken the location of pixel is even only
                          Convert P to three pixels are Red, Green, and Blue that give (RGB) pixels and read
                         eight bit of each pixel from RGB pixels and store in bin array.
                    Else
                     Go to step5
Step3: Repeat step2 until n equal to 8 // where n represent count of bin array.
         If(n = 8)
          Convert values(binary numbers) that store in bin array into decimal number that represent
          ASSCI code from it get on letter from secret data.
          bin= ""  // To store new values in bin array
          n=0
Step4: Repeat steps (step2 and step3) Until Get on All secret data.
Step5: Next y
Step6: Next x
Step7:End
```

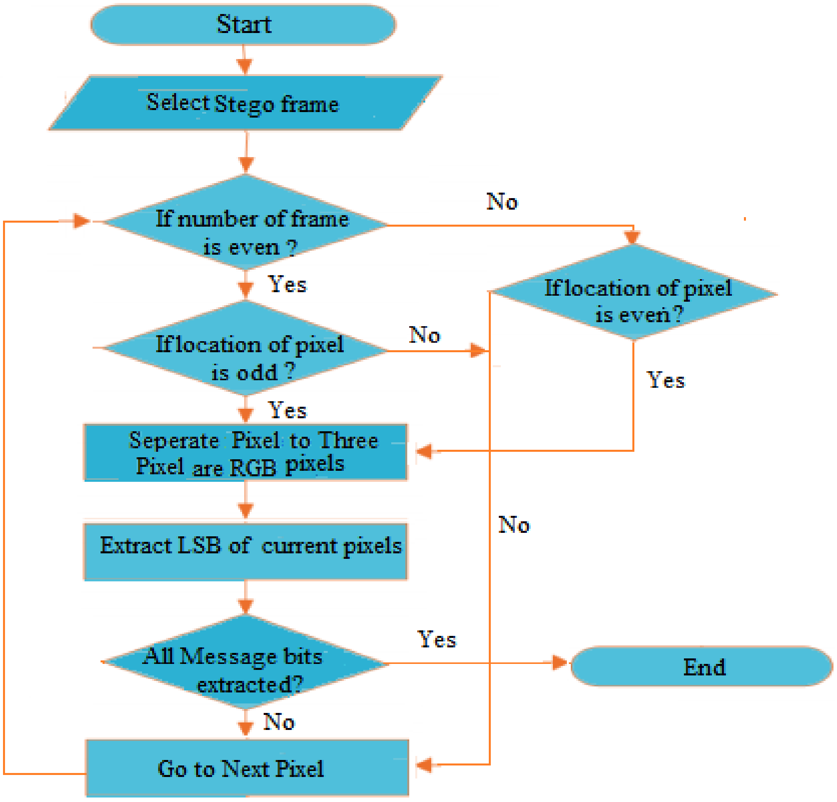

Start
Select Stego frame
If number of frame is even ?
No
Yes
If location of pixel is even?
If location of pixel is odd ?
No
Yes
Yes
Seperate Pixel to Three Pixel are RGB pixels
Extract LSB of current pixels
No
All Message bits extracted?
Yes
End
No
Go to Next Pixel

Figure 4- Extraction Algorithm Flowchart.

Decompression phase

Haar Wavelet Transform has feature of symmetric that mean both the forward and the inverse transform has the same complexity [11], that means decompression phase is same of compression phase but in inverse direction and the result is decompression frame, Figure-5 shows this. After decompression frames of video, now combined in the same way that mentioned earlier to get on target video.

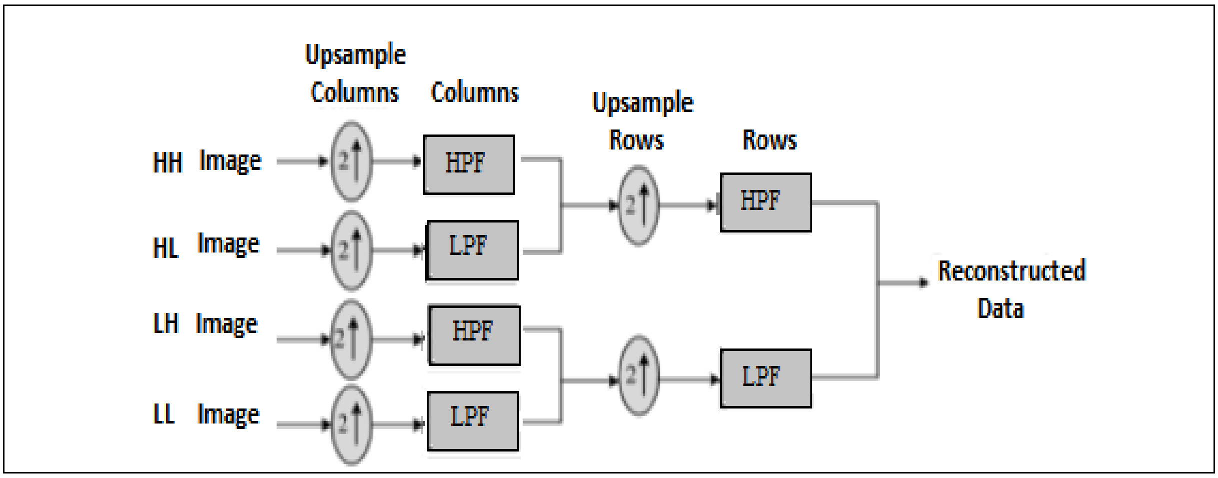


Figure 5- One Stage of Inverse DWT [9].

Experimental Results and Discussion

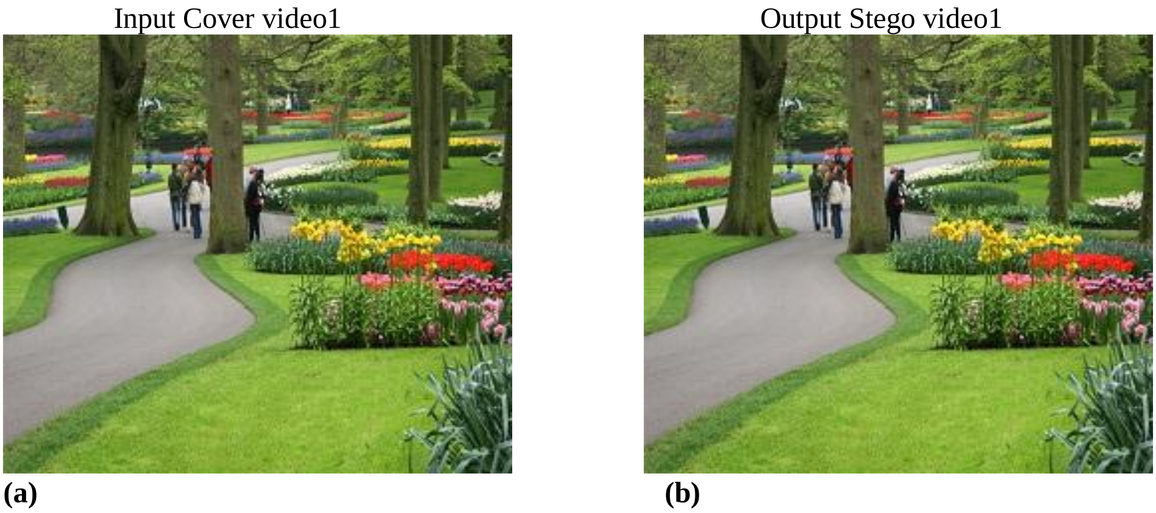
The proposed system is simulated using C# programming. For experiments it have been embedded variable amount of secret data of different dimensions of videos as shows in Figures-6, 7 and 8. To estimate the performance of the proposed technique by two quality measurements were used; MSE and PSNR; where PSNR analyzes the quality of stego frame with original frame [12, 13].

$$PSNR = 10\log_{10}\frac{(peakval)^2}{MES} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (1)$$

Where *peakval* is either specified by the user or taken from the range of the frame data type and this rang is (from 0 to 255) and MSE used to measure determination the difference between the original frame and the stego frame I', If the frame has a size of M * N then

$$MSE = \frac{1}{M \times N}\sum_{i=1}^{M}\sum_{j=1}^{N}[I(i,j) - I'(i,j)]^2 \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (2)$$

The results of PSNR, MSE, and the processing time in proposed system applied on three frames from three video, one frame from each video before and after embedding as examples, so to find final result of the video must calculate results of PSNR, MSE of all frames of each video and this take time and effort, Table-1 shows results of one frame of video1, Table -2 shows results of one frame of video2, Table-3 shows results of one frame of video3, these results prove success of proposed method compere with another results from another papers [14, 15], with take into consideration the time engrossed in to execution various operations is in nanosecond that means operations take few time in the execution.

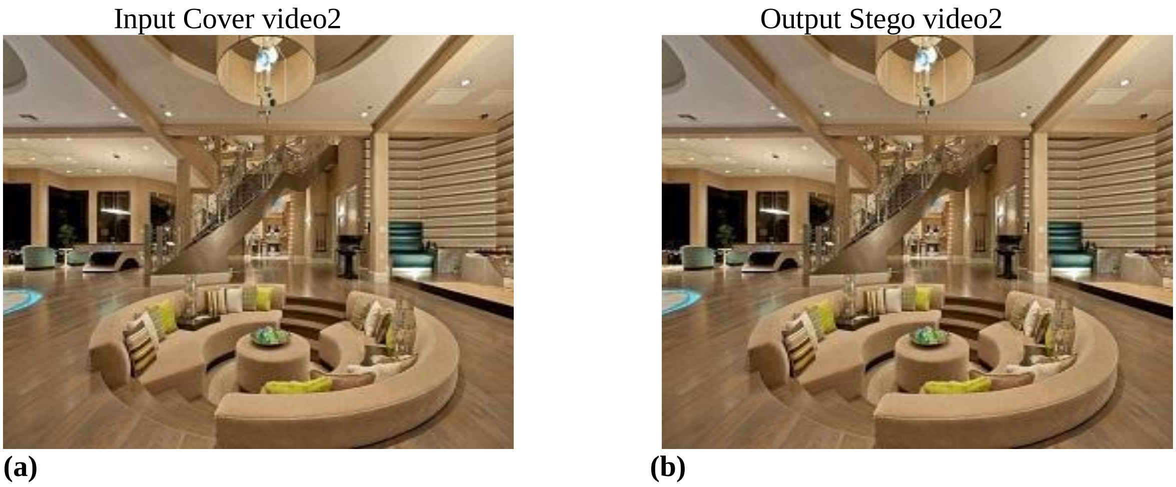

Input Cover video1

Output Stego video1

(a)

(b)

Figure 6- video1 cover and stego video1 with dimension (640×480).


Input Cover video2

Output Stego video2

(a)

(b)

Figure 7- video2 cover and stego video2 with dimension (480,360).

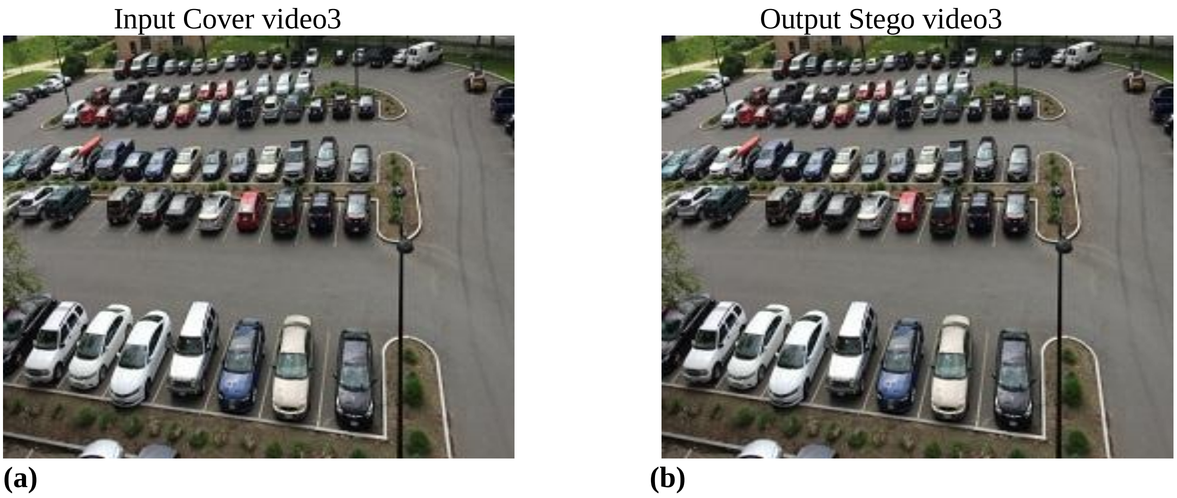


Figure 8- video3 cover and stego video3 with dimension (320,240).

Table 1- Result MSE, PSNR, and processing time of video1 with difference secret data size.

Video1 with Dimension 640×480	**Secret data size in (KBs)**	**PSNR(dB)**	**MSE**	**Processing Time Nano sec.**
	2	72.4886	7.8374	218431701
	4	82.9553	8.7483	239387627
	6	68.5992	9.9283	263873682

Table 2- Result MSE, PSNR, and processing time of video2 with difference secret data size.

Video2 with Dimension 480×360	**Secret data size in (KBs)**	**PSNR(dB)**	**MSE**	**Processing Time Nano sec.**
	2	65.8347	4.7362	227382745
	4	75.2483	5.6584	247329892
	6	59.1847	6.9873	272817489

Table 3- Result MSE, PSNR, and processing time of video3 with difference secret data size.

Video3 with Dimension 320×240	**Secret data size in (KBs)**	**PSNR(dB)**	**MSE**	**Processing Time Nano sec.**
	2	62.4387	1.3827	213847267
	4	69.7362	2.4928	228376474
	6	53.3851	3.8728	252817637

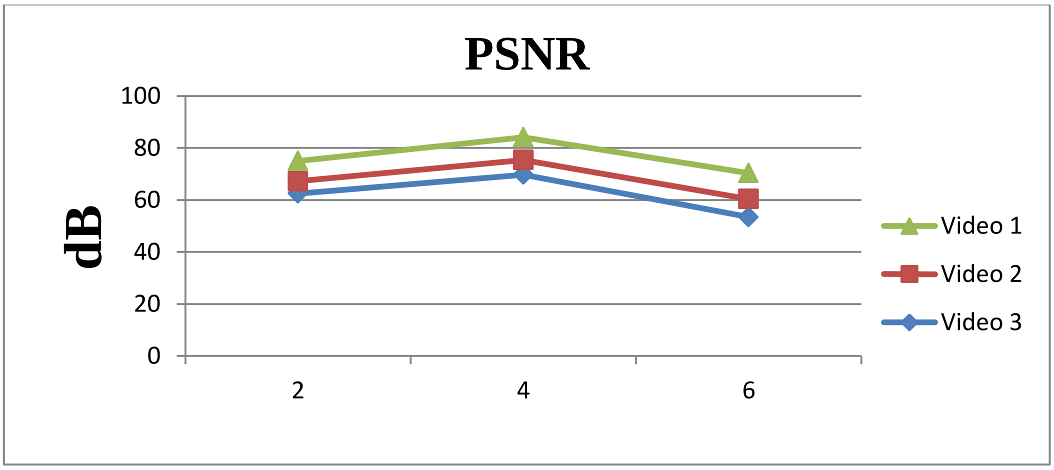


Figure 9- PSNR with three videos of different dimensions and variable amount of secret data.

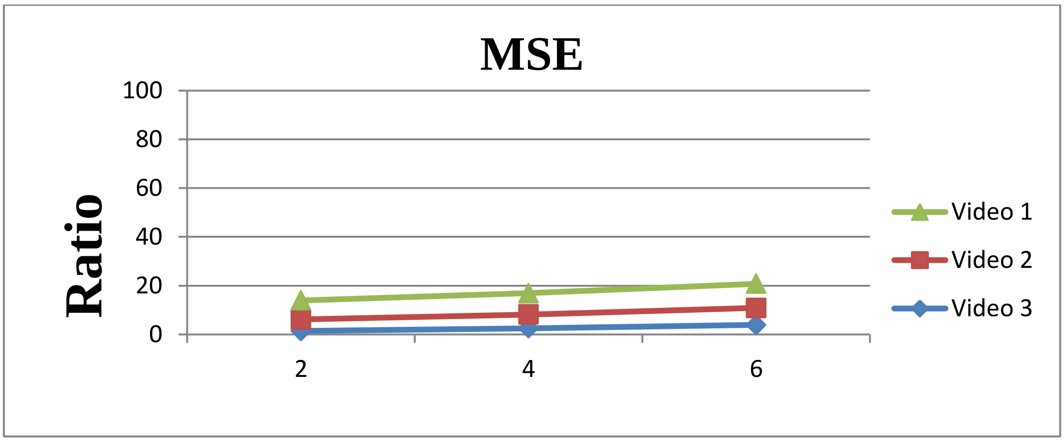


Figure 10- MSE with three videos of different dimensions and variable amount of secret data.

The video with difference dimensions can hide variable amount of secret data inside it frames, when the secret data is big then choose video with big dimensions to can hiding the larger amount of information within it. The MSE used for measure the difference (error) between cover video and stego video, so if MSE large that means the error is large. From results note the results of MSE in tables shows the amount of secret data in (KBs) with size 6 is larger than results of secret data with size 2, 4 because the amount of secret data larger size from other, that means the secret data that hide inside frames make small change between cover frames and stego frames, but this change not notice because the Least Significant Bit (LSB) embedding technique hiding the data in the least significant bits of the cover video, so the human eye would be unable to notice the hidden information in the cover file and this is the reason for efficiency LSB technique. The relation between MSE and PSNR is inversely, that means if MSE is low PSNR will be height [10]. MSE gives ratio is 5.7382 of average of three test videos and PSNR gives 67.7634 dB from this note the result of PSNR is higher than result of MSE, which means the results from proposed system are correct and acceptable.

Conclusion

The proposed method has been employed for applications that require high volume embedding. The steganography is a technique to hide information inside video file for the secure data transmission from the sender to receiver through the internet by using LSB embedding technique suggests that data can be hidden in the least significant bits of the cover video and the human eye would be unable to notice the hidden information in the cover file. The proposed method hides information in any or number of cover frames for various video sizes and in specific location of the pixels. When transfer or download big video will take more time and storage space, so used compression method with steganography to reduce storage size of video with reduce bandwidth to transfer data in faster away with save time that requires transferring video during the network. This method introduces and adds an extra security level barrier in the way of an attacker which makes the attack on this algorithm awful and misguides the process of steganalysis.

References

1. Hussain, M. and Hussain, M. **2013.** A Survey of Image Steganography Techniques. *International Journal of Advanced Science & Technology*, **54**(1):113 -124.
2. Kaur, J. and Kaur, J. **2016.** Hiding Text in Video Using Steganographic Technique - A Review . *An International Journal of Engineering Sciences*, **17**(1): 578 - 582.
3. Reddy,V. L.Subramanyam,A.Reddy,C.**2011.** Implementation of LSB Steganography and its evaluation for various file formats. *Int. J. Advanced Networking and Applications* 868, **2**(5): 868-872.
4. Dey, N. Roy, A .B. Dey, S. **2011.** A Novel approach of color image hiding using RGB color planes and DWT ". *International Journal of Computer Applications (0975 – 8887)*, **36**(5): 19-24.
5. Muhammad, K.Ahmad, J. Rehman,N.U. Jan, Z. Qureshi, R.G.**2014.** A Secure Cyclic Steganographic Technique for Color Images using Randomization. *Technical Journal, University of Engineering and Technology Taxila, Pakistan*, **19**(1): 57-64.

6. Patil, S. Jagtap, S.K. **2016.** Video Compression Exploiting Luminance Masking and DWT. *.International Journal of Advanced Research in Computer and Communication Engineering*, **5**(1): 358- 360.

7. Kaur, J. Sharma, R. **2012.** A Combined DWT - DCT approach to perform Video compression base of Frame Redundancy. *International Journal of Advanced Research in Computer Science and Software Engineering,* **2**(9): 325- 332.

8. Gupta,D. Choubey, S .**2015.** Discrete Wavelet Transform for Image Processing. *International Journal of Emerging Technology and Advanced Engineering*, **4**(3): 598-602.

9. Rathee, M. Vij, A **. 2014.** Image compression using discrete Haar wavelet transforms. *International Journal of Engineering and Innovative Technology (IJEIT)*, **3**(12): 47-51.

10. Laskar,S.A. Hemachandran, K. **2012.** High Capacity data hiding using LSB Steganography and Encryption. *International Journal of Database Management Systems (IJDMS),* **4**(6): 57-68.

11. Gupta, R. **2014.** Image compression using Haar wavelet transform and chaos-based encryption. *IJCSI International Journal of Computer Science Issues*, **11**, Issue 2(1): 174 -180.

12. Laskar,S. A. Hemachandran, K . **2012.** High capacity data hiding using LSB steganography and encryption. *International Journal of Database Management System (IJDMS)*, **4**(6): 57-68.

13. Reddy,V. R. Reddy,T. S .**2014.** Image encryption using fractional random wavelet transform . *International Journal of Advanced Research in Computer and Communication Engineering,* **3**(1): 4891-4893.

14. Poobathy, D . Chezian, R. M . **2014.** Edge Detection Operators : Peak Signal to Noise RatioBased Comparison. *I.J. Image, Graphics and Signal Processing*, 7(10): 55-61.

15. Reddy, V.L. Subramanyam, A. Reddy, P.C. **2011.** Implementation of LSB Steganography and its Evaluation for Various File Formats. *Int. J. Advanced Networking and Applications*, **2**(5): 868-872.