Pertanika J. Sci. & Technol. 24 (2): 411 - 422 (2016)

SCIENCE & TECHNOLOGY

Journal homepage: http://www.pertanika.upm.edu.my/

High Capacity Video Steganography Technique in Transform Domain

Hemalatha, S.*, U. Dinesh Acharya and Renuka, A.

Department of Computer Science and Engineering, Manipal Institute of Technology, Manipal University, Manipal - 576104, India

ABSTRACT

Steganography is one of the techniques used for secure transmission of secret information. The secret information is concealed in a carrier and transmitted. Video steganography uses video signals to hide the secret information. The objective of this paper is to hide large volumes of secret data in video files. In the proposed technique, AVI video files are used as the carrier. Video files containing audio are split to get video and audio frames. Video frames are like still images and so, can be used for image steganography. When audio is extracted from the video, it is like an audio file and it can also be used for steganography. This leads to the high capacity steganography, since both video and audio frames are used as the carrier. When a small video clip is read, thousands of video and audio frames will also become available. The secret data can be image, audio or text. In this proposed work, secret image and audio signals are hidden in the video file.

Keywords: Capacity, Integer Wavelet Transform, Peak Signal to Noise Ratio, Structural Similarity Index Metric, Video Quality Metric (VQM), Video steganography

INTRODUCTION

Steganography is the science of secret communication. The secret information to be communicated is hidden in another carrier and transmitted. In video steganography a video clip can be a carrier. A video stream consists of many images and audio frames. All these frames can be used to hide the secret information, so, any image and/or audio steganography techniques can be used with video steganography too. Video steganography gives flexibility of selective frame steganography to increase the security of the system or it can use the whole video for hiding huge amounts of data so that the hiding capacity is much higher in the case of video. Compressed video formats

Article history:
Received: 17 February 2016
Accepted: 22 April 2016

E-mail addresses:
hema.shama@manipal.edu (Hemalatha, S.),
dinesh.acharya@manipal.edu (U. Dinesh Acharya),
renuka.prabhu@manipal.edu (Renuka, A.)
*Corresponding Author

ISSN: 0128-7680 © 2016 Universiti Putra Malaysia Press.

however, are challenging because during compression the hidden information may be lost. The most commonly used codec for video is H.264/AVC. The underlying coding structure can be modified to hide data but in this case, only a small amount of data can be hidden securely. Other possible ways of data hiding in video are hiding in motion vectors or macro blocks (Sadek et al., 2014; Tew & KokSheik, 2014).

Compared with image steganography, video steganography is harder to be tested by the attacker because random frames or audio tracks can be selected for the hiding process and for the attacker it is difficult to guess the sequence of frames or audio tracks. There are many attacks for videos such as lossy compression, change of frame rate, frames interchanging and addition or deletion of frames during video processing among others. The challenge in video steganography is to make the steganography technique resistant to these attacks.

Three important characteristics of steganography are invisibility or imperceptibility, robustness and security. Invisibility or imperceptibility is the ability to be unseen by human eye. Robustness is the aptitude of the steganography to resist operations such as filtering, cropping, rotation, and compression. If the hidden information is not detected by third party, then the system is said to be secure. Even if it is suspected, it should be impossible to detect the hidden information by the third party.

Capacity is another important factor to be considered in steganography. It is the amount of information that can be hidden relative to the size of the cover object without reducing the quality of the cover object. Usually, it is specified as the number of bits per bytes or kilobytes. The more information the cover can carry the better it is. Large embedded information however, degrades the quality of the steganography (referred to as stego henceforth) significantly. This means that when capacity is increased, the security decreases and vice versa

Steganography can be in spatial domain or in transform domain. In spatial domain techniques, actual sample values are modified to hide the secret information. These techniques are less resistant to signal processing operations. In transform domain, the actual sample values are transformed to some other domain like frequency domain or time-frequency domain. The transformed coefficients are then modified to hide the secret information. Transform domain techniques are robust against signal processing operations (Sadek et al., 2014).

In this paper, a transform domain video steganography technique is proposed. Integer Wavelet Transform (IWT) is used to transform the video and audio frames into time-frequency domain. The IWT is a type of wavelet transform which maps integer to integers and can be used to transform digital signals.

MATERIALS AND METHODS

A study (Su et al., 2013) proposed high volume of data embedding in compressed video files in H.264/AVC format. The process is too complex and it achieved only 10% of the capacity of the video file size. Since in this process the size of the stego file increases, it is prone to suspicions about hidden data. Some researchers (Gujjunoori & Amberker, 2013) have proposed a data embedding technique for MPEG-4 video using HVS characteristics in DCT domain to improve the capacity while maintaining good visual quality. Tamer Shanableh (2014) proposed two approaches. The message bits are hidden using modulation of the quantisation scale of a constant bitrate video in the first approach. A payload of one message bit per macro-block is achieved. In the second approach, a second order multivariate regression is used. The regression model is then used by the decoder to predict the values of the hidden message bits. This also did not improve the capacity.

A few researchers (Xu et al., 2014) have attempted to hide data in the encrypted H.264/AVC video, but it will not increase the security level of data hidden. A study (Yao et al., 2014) proposed a motion vector based steganography by defining embedding distortion. But it increased the bit rate which is not appreciated in steganography. Other researchers (Gaj et al., 2015), proposed a watermarking technique for H.264/AVC video which resists rotation, scaling and translation attacks by embedding the watermark in the moving objects of the video. Quality of a video is measured using many metrics. Peak Signal to Noise Ratio is the most widely used metric. Pinson and Wolf (2004, 2011) discovered a new standardised method for objectively measuring video quality, namely Video Quality Metric or VQM. Winkler and Wang, 2009) and Tew & KokSheik (2014) also suggested that VQM is one of the best metrics to measure the video quality.

All the above compressed video steganography papers did not achieve good payload capacity. Compared with the size of the video, the capacity was too low. The capacity can be improved if the information is hidden in many frames. But selection of frames which can resist compression is the challenge. The proposed method is explained below.

Video files containing audio are split to get video and audio frames. Video frames are like still images and so can be used for image steganography. When audio is extracted from the video it is like an audio file and it can also be used for steganography. When a small video clip is read, thousands of video and audio frames become available. All these frames can be used for steganography. The stego video file however, cannot resist compression, even though the other attacks such as RST attack, addition and deletion of frames can be handled efficiently. When the RST attack is applied, the secret data is not lost since it is applied to the entire frame. To handle frame deletion or addition, random frames are selected as per random number generator and secret information is hidden in duplicates. Pseudo Random Number Generator (PRNG) principle is used to select the frames. Only the seed has to be exchanged between sender and receiver. The first two prime factors of the seed are discovered. Sum of these two prime factors gives the first frame number where the secret data is hidden; then for the next two continuous frames, same secret data are hidden. This procedure is repeated three times in the frames at an offset of sum of the prime factors. For example, if 10 is the seed then 2 and 5 are its first two prime factors. Their sum is 7. Secret data is hidden in the continuous frames 7, 8 and 9. Then in the frames 16, 17 and18 (because 9+7=16) and in frames 25, 26 and 27 (because 18+7=25) same data are hidden. At the receiver, data is extracted from all these frames and checked for majority similarity. There is no chance of losing all such frames. Thus, any addition or deletion of frames can be handled.

The audio frame size is small and the first frame always contains silent period. Hence, except for the first frame, all other frames are combined to hide secret audio. The secret audio is hidden in duplicate to resist frame loss.

Secret information is hidden in the video frames using Integer Wavelet Transform (IWT). The IWT decomposes the image into four sub bands LL, LH, HL and HH. The low frequency

LL sub band contains the most significant features and the high frequency sub bands contain less significant features. Thus, if the secret information is an image, only LL sub band will be hidden and during extraction, the secret image is constructed using only LL sub band. When IWT is applied to audio signals, it is converted into approximate or low frequency components and detail or high frequency components. Detail components are not significant. So to hide secret audio in audio frames, only approximate components are hidden and it is possible to reconstruct the secret audio using only approximate components.

Experimentation is done in MATLAB 8.2. The embedding and extracting procedures are explained in the following sections. The MATLAB functions used in this proposal are listed in the Appendix.

Embedding Procedure:

Input:

- Cover video *C.avi*
- Secret images *S1.jpg and S2.jpg.*
- Secret audio *S3.wav.*

Output: Stego video, *G.avi.*

Method:

1. Read cover video *C.avi* and get video and audio frames. Store audio frames in '*a*' and video frames in '*im*'.
2. Get the frame numbers as per the PRNG and hide the secret image in duplicates. Separate out Red Green and Blue components from the selected frames. Two secret images are hidden in one frame: one in Blue component and the other in Green component. Hiding procedure in the frame is explained in the following steps.
3. Obtain IWT of Blue and Green components to get four sub bands in each as BLL, BHL, BLH and BHH, GLL, GHL, GLH and GHH.
4. Obtain IWT of the secret images *S1* and *S2* to get four sub bands of each as sLL1, sHL1, sLH1 and sHH1, sLL2, sHL2, sLH2 and sHH2.
5. Hide the low frequency sub band of the secret image *S1* by replacing the coefficients in BHH band. Also, hide the number of coefficients hidden.

```
for i=1 to size(sLL1) do
begin
BHH(i) = sLL1(i)
end.
```

6. Hide the low frequency sub band of the secret image *S2* by replacing the coefficients in GHH band along with the number of coefficients hidden.

```
for i=1 to size(sLL2) do
```

```
begin
GHH(i) = sLL2(i)
end.
```

7. Obtain the inverse IWT to get Blue and Green components
8. Integrate the R, G and B components into a single image frame
9. Add the frame to the video in the correct location.

The secret audio is hidden as follows:

10. Except for the first frame, all other audio frames are combined.
11. Obtain IWT of the combined audio frames and secret audio *S3* to get approximation (CA) and detail coefficients (CD).
12. Obtain the binary of approximation coefficients of the secret audio and duplicate the bits 3 times.
13. Hide the duplicated secret bits in the 3rd, 4th and 5th bit planes of the detailed coefficients of the cover.
14. Obtain the inverse IWT of the cover to get the stego audio samples and then convert into frames. Add the frames to the audio in the correct location.
15. Store the video and audio frames in the AVI format in the uncompressed form to get stego video *G.avi*.

Extracting Procedure:

Input: Stego video *G.avi*

Output:

- Secret images *S1.jpg.* and *S2.jpg*.
- Secret audio *S3.wav*.

Method:

1. Get the frame numbers as per the PRNG and extract the required video and audio frames from the stego video. The hidden secret images are extracted as follows:
2. Decompose the extracted video frames into Blue and Green components.
3. Obtain IWT of Blue and Green components and get the secret image sizes.
4. Extract the low frequency sub bands of the secret images hidden in the high frequency sub bands of Blue and Green components of all the selected frames. The size of the low frequency sub band is one fourth of the secret image size since ‘haar’ transform is used.

```
for i=1 to (secret size)/4
begin
newSLL1(i) = gHH1(i)
```

newSLL2(i) = gHH2(i)

end// gHH1 and gHH2 are the HH sub band of the Blue and Green components of the stego frame respectively. The size of the two secret images are taken as same.

5. For the extracted low frequency sub bands of the secret images, obtain the inverse IWT by considering zeroes as high frequency components to get secret images
6. Extract all the duplicates of the secret images similarly. The corresponding secret images must be compared for majority similarity and that will be selected as the secret image. Thus obtain the secret images *S1* and *S2*.

The secret audio is extracted as follows:

7. Except for the first frame, combine all the audio frames and obtain the IWT to get approximate and detail coefficients.
8. Extract the 3rd, 4th and 5th bits of detail coefficients as per the number of secret bits hidden.
9. Form the bits into three groups and perform the majority evaluation and obtain the secret coefficients. These are the approximate coefficients of the secret audio.
10. Perform inverse IWT for these approximate coefficients, considering detailed coefficients as zeros to get the secret audio *S3*.

RESULTS AND DISCUSSION

Figure 1 shows one of the frames. The frame size is 240 X 320. Figure 2 shows the secret images. The stego frame and the extracted secret images are shown in Figure 3.

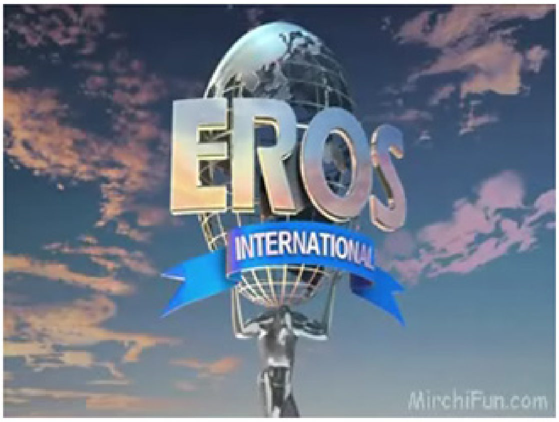

Figure 1. One of the frames from the video
Source: downloaded from MirchiFun.com in January 2015.

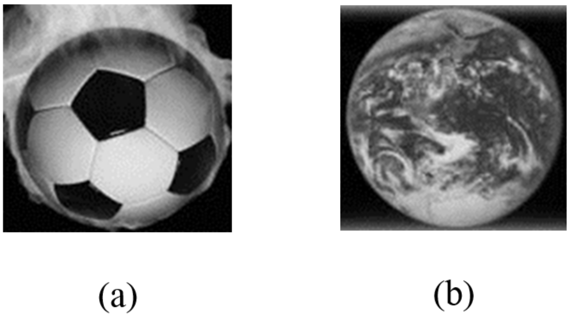

Figure 2. Secret images : (a) football, (b) earth
Source: downloaded from Google in June 2013

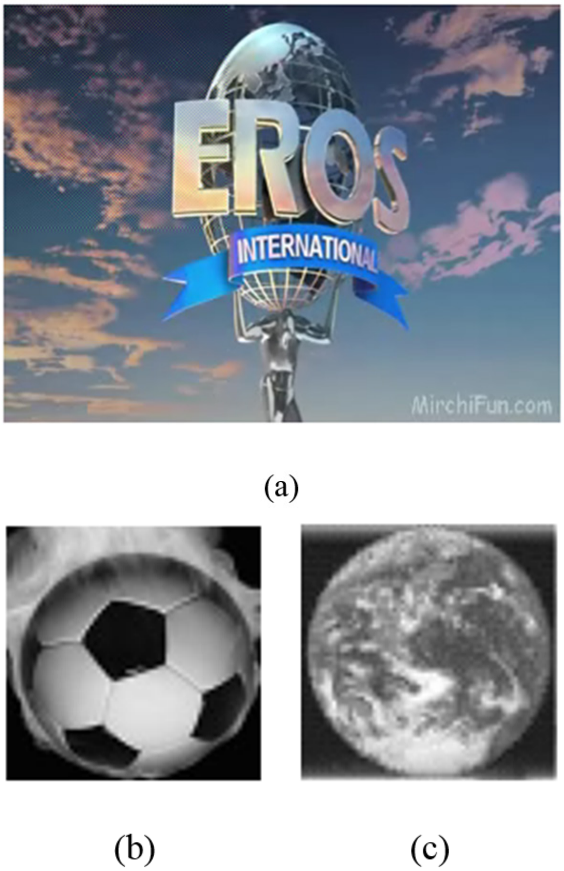

Figure 3. Stego frame and Extracted images: (a) stego frame (b) extracted football (c) extracted earth

The quality of the stego video is measured using Video Quality Metric (VQM). The VQM (Pinson & Wolf, 2004) is developed by the Institute for Telecommunication Science (ITS) which is used as an objective measurement to evaluate the quality of the video. It measures the perceptual effects such as blurring, unnatural motion, noise and colour distortion among others and combines them into a single metric. The VQM has a high correlation with subjective video quality assessment and it has been accepted by American National Standards Institute (ANSI) as an objective video quality standard. Hence, VQM is a better quality metric than Peak Signal to Noise Ratio (PSNR).

The ITS has developed a video quality metric (BVQM) software tool that performs automated batch processing of multiple video clips to objectively assess their video quality (Pinson & Wolf, 2011). The processed video clips are calibrated and the VQM is calculated. This tool is used to measure VQM. The VQM ranges between '0' and '1'. If it is '0', that indicates no difference between the original and the stego video meaning it is of good quality. '1' indicates a fully distorted video.

The extracted secret image quality is measured using PSNR and Structural Similarity Index Metric (SSIM). The PSNR is a simple statistics error metric which objectively quantifies the error signal. With PSNR, greater values indicate greater similarity. It is given by equation [1].

$$PSNR = 10 \times log_{10}\left(\frac{MAX^2}{MSE}\right) \qquad [1]$$

where *MAX* = maximum value, MSE = Mean Square Error as given by equation [2].

$$MSE = \frac{1}{M \times N} \sum_{i=1}^{M} \sum_{j=1}^{N} \| O(i,j) - D(i,j) \|^2 \qquad [2]$$

O(i,j) is original pixel and *D(i,j)* is stego pixel. *M X N* is the size of image. It is expressed in decibels (dB).

The SSIM is an objective image quality metric and is superior to traditional PSNR which estimates the perceived errors, whereas SSIM considers image degradation as perceived change in structural information. The SSIM is given by equation [3].

$$SSIM = \frac{(2 \times \bar{x} \times \bar{y} + C1)(2 \times \sigma_{xy} + C2)}{({\sigma_x}^2 + {\sigma_y}^2 + C2) \times (\bar{x}^2 + \bar{y}^2 + C1)} \qquad [3]$$

where C1 = $(k_1L)^2$, and C2 = $(k_2L)^2$ are two constants used to avoid null denominator. L is the dynamic range of the pixel values (typically this is $2^{\#\text{ bits per pixel}}$ -1). k_1 = 0.01 and k_2 = 0.03 by default.

The dynamic range of SSIM is between -1 and 1. The maximum value of 1 will be obtained for identical images. Equation [3] can be written as the product of three terms: M1, M2 and M3.

$$M1 = \frac{2 \times \bar{x} \times \bar{y} + C1}{\bar{x}^2 + \bar{y}^2 + C1}\,. \qquad [4]$$

$$M2 = \frac{2 \times \sigma_x \times \sigma_y + C2}{{\sigma_x}^2 + {\sigma_y}^2 + C2} \qquad [5]$$

$$M3 = \frac{\sigma_{xy} + \mathrm{C3}}{\sigma_x * \sigma_y + \mathrm{C3}} \qquad [6]$$

where $C3 = \frac{C2}{2}$.

M1 indicates luminance distortion, M2 indicates contrast distortion and M3 indicates structural distortion (Wang et al., 2004); Sasi varnan et al., 2011).

The extracted secret audio is evaluated using Signal to Noise Ratio (SNR) and Squared Pearson Correlation Coefficient (SPCC). The SNR is a term that refers to the measurement of the level of an audio signal compared to the level of noise that is present in that signal. It is expressed in decibels (dB). A larger SNR value indicates a better quality. It is given by equation [7].

$$\mathrm{SNR} = 10\log_{10}\left(\frac{\frac{1}{N}\sum_{i=0}^{N} xi^{2}}{MSE}\right) \quad [7]$$

The recommended SNR for audio signal is above 30dB.

The SPCC is based on correlation of samples. The high value of SPCC indicates a good quality of the output signal. Its range is between 0 and 1. It is given by equation [8].

$$SPCC = \left[\sum \frac{(x-\bar{x})(y-\bar{y})}{\sqrt{\sum(x-\bar{x})^{2}}\sqrt{\sum(y-\bar{y})^{2}}}\right]^{2} \quad [8]$$

where x, y, $\bar{x}$ and $\bar{y}$ are the cover signal, stego signal, average of the cover signal and average of the stego signal, respectively (Ballesteros & Moreno, 2013).

Table 1 shows the performance metrics for the stego video and extracted secret images and secret audio. When the payload is increased, the quality of the stego is reasonably good even with six secret images and 40000 secret audio samples. The quality of the extracted secret audio however, decreases and if the payload is further increased, the extracted secret audio will be distorted. Table 2 shows performance against RST attack. A rotation of 3 degrees is applied to all the stego frames and then scaled to the original size. It is possible to extract the secret image and audio with reasonable performance metrics. Table 3 compares the proposed video steganography algorithm with the research outcome of other video steganography. Since VQM is not calculated in those papers, the comparison is done in terms of PSNR with reference to payload capacity percentage. The results show that the proposed technique has better metrics than the others.

CONCLUSION

This paper proposes a high capacity video steganography technique using AVI video files as cover object. The secret information is hidden in the cover video frames, instead of hiding in the Meta data or macro block as found in the literature. The audio frames are also used for information hiding. The experimental results show that the technique is robust and secure.

Table 1
Performance Metrics

Cover video	Secret images (gray scale) and Secret audio	Stego	Extracted image		Extracted audio	
		Average VQM	PSNR in dB	SSIM	SNR in dB	SPCC
C.avi	Two images with size 128X128 and 20000 secret audio samples	0.0001	S1:36.2 S2:36.1	S1:0.9949 S2:0.9053	30.1	0.9235
	Four images with size 128X128 and 32768 secret audio samples	0.0002	S1:36.2 S2:36.1 S3:35.9 S4:32.4	S1:0.9949 S2:0.9053 S3:0.9894 S4:0.8812	28.5	0.8906
	Six images with size 128X128 and 40000 secret audio samples	0.0041	S1:36.2 S2:36.1 S3:35.9 S4:32.4 S5:36 S6:34.4	S1:0.9949 S2:0.9053 S3:0.9894 S4:0.8812 S5:0.9907 S6:0.9355	25	0.7536

Table 2
Performance Against RST Attack

Cover video	Secret images (gray scale)	Stego	Extracted image		Extracted audio	
		VQM	PSNR in dB	SSIM	SNR in dB	SPCC
C.avi	Two images with size 128X128 and 20000 secret audio samples	0.0591	S1:29 S2:30.5	S1:0.7940 S2:0.8105	29.5	0.7985

Tabel 3
Performance Comparison of Proposed Video Steganography Algorithm with That of Other Related Published Work

Algorithm	Payload capacity in %	PSNR in dB
Su et al. (2013)	14.5	36.5
Gujjunoori & Amberker (2013)	3.1	25.0
Liu et al. (2013)	2.0	36.0
Proposed	17.6	34.0

REFERENCES

Ballesteros, L. D. M., & Moreno, A. J. M. (2013). Real-time, speech-in-speech hiding scheme based on least significant bit substitution and adaptive key. *Computers and Electrical Engineering, 39*(4), 1192–1203.

Gaj, S., Patel, A. S., & Sur, A. (2015, January 23). Object based watermarking for H.264/AVC video resistant to rst attacks. *Multimedia Tools and Applications,* 1-28. doi: 10.1007/s11042-014-2422-3.

Gujjunoori, S., & Amberker, B. B. (2013). DCT based reversible data embedding for MPEG-4 video using HVS characteristics. *Journal of information security and applications, 18*(4), 157 -166.

Liu, Y., Li, Z., Ma, X., & Liu, J. (2013). A robust data hiding algorithm for H.264/AVC video streams. *The Journal of Systems and Software, 86*(8), 2174– 2183

Pinson, M. H., & Wolf, S. (2011). *Batch Video Quality Metric (BVQM) User's Manual.* Handbook series, U. S. Department of Commerce, National Telecommunications and Information Administration.

Pinson, M. H., & Wolf, S. (2004). A New Standardized Method for Objectively Measuring Video Quality. *IEEE Transactions on Broadcasting, 50*(3), 312-322.

Sadek, M. M., Khalifa, A. S., & Mostafa, M. G. (2014, March 20). Video steganography: a comprehensive review. *Multimedia Tools and Applications, 74*(17), 7063-7094. doi: 10.1007/s11042-014-1952-z.

Sasi varnan, C., Jagan, A., Kaur, J., Jyoti, D., & Rao, D. S. (2011). Image Quality Assessment Techniques in Spatial Domain. *International Journal of Computer Science and Technology, 2*(3), 177-184.

Shanableh, T. (2014). Data Hiding in MPEG Video Files Using Multivariate Regression and Flexible Macroblock Ordering. *IEEE transactions on information forensics and security, 7*(2), 455-464.

Su, P. C., Lu, M., & Wu, C. (2013). A practical design of high-volume steganography in digital video files. *Multimedia Tools and Applications, 66*(2), 247–266.

Tew, Y., & KokSheik, W. (2014). An Overview of Information Hiding in H.264/AVC Compressed Video. *IEEE Transactions on Circuits and Systems for Video Technology, 24*(2), 305-319.

Wang, Y. (2006). Survey of Objective Video Quality Measurements. *EMC Corporation Hopkinton, MA 01748, USA*, pp.1-7.

Wang, Z., Bovik, A. C., Sheikh, H. R., & Simoncelli, E. P. (2004). Image Quality Assessment: From Error Visibility to Structure Similarity. *IEEE Transactions on Image Processing, 13*(4), 600-612.

Winkler, S. (2009). Video Quality Measurement Standards –Current Status and Trends. *Proc. 7th International Conference on Information, Communications & Signal Processing*, (pp. 1-5).

Xu, D., Wang, R., & Shi Yun, Q. (2014). Data Hiding in Encrypted H.264/AVC Video Streams by Codeword Substitution. *IEEE Transactions on Information Forensics and Security, 9*(4), 596-606.

Yao, Y., Zhang, W., Yu, N., & Zhao, X. (2014, September 3). Defining embedding distortion for motion vector-based video steganography. *Multimedia tools and Applications*. doi: 10.1007/s11042-014-2223-8

APPENDIX

Following MATLAB functions are used in the experimentation:

To read video file *C.avi* and to get number of video frames *n*:

```
VideoFreader=vision.VideoFileReader(‘C.avi’)
vid=VideoReader( ‘C.avi’) (Sadek, Khalifa, & Mostafa, 2014)
n= vid.NumberOfFrames
```

To extract audio and image frames (*a* and *im*) from the video:

```
for i=1 to n do
begin
[I, Audio] =step(VideoFreader)
a{i}=Audio
im{i}=I
end
```

To decompose an image *B* into four sub bands *BLL, BHL, BLH, BH* using IWT:

```
LS = liftwave(‘haar’,’Int2Int’)
[BLL,BHL,BLH,BHH] = lwt2(double(B), LS)
```

To get back the image *B,* inverse IWT is applied as follows:

```
B=ilwt2(BLL, BHL, BLH, BHH, LS)
```

To obtain the IWT of audio frame *a*:

```
[CAc, CDc] = lwt(double(a), LS)
```

To get back the audio frame, inverse IWT is applied as follows:

```
a= ilwt (CAc, CDc, LS)
```

To write video and audio frames (*im* and *a*) into a video file *G. avi* (*n* is the total no. of frames):

```
VideoFwriter=vision.VideoFileWriter(‘G.avi’)
for i=1to n do
begin
step(VideoFwriter,im{i},a{i})
end.
```