A Compressed Video Steganography using TPVD

Sherly A P and Amritha P P

TIFAC CORE in Cyber Security , Amrita Vishwa Vidyapeetham, Coimbatore

{sherlyram,ammuviju} @gmail.com

Abstract

Steganography is the art of hiding information in ways that avert the revealing of hiding messages. This paper proposes a new Compressed Video Steganographic scheme. In this algorithm, data hiding operations are executed entirely in the compressed domain. Here data are embedded in the macro blocks of I frame with maximum scene change and in block of P and B frames with maximum magnitude of motion vectors. To enlarge the capacity of the hidden secret information and to provide an imperceptible stego-image for human vision, a novel steganographic approach called tri-way pixel-value differencing (TPVD) is used for embedding. In this scheme all the processes are defined and executed in the compressed domain. Though decompression is not required. Experimental results demonstrate that the proposed algorithm has high imperceptibility and capacity.

Keywords: Video Steganography, MPEG, Tri-way PVD

1. Introduction

Text, image, audio, and video can be represented as digital data. The explosion of Internet applications leads people into the digital world, and communication via digital data becomes recurrent. However, new issues also arise and have been explored, such as data security in digital communications, copyright protection of digitized properties, invisible communication via digital media, etc. Steganography is the art of hiding information in ways that prevent the detection of hiding message whereas cryptographic techniques try to conceal the contents of a message. In steganography, the object of communication is the hidden message and the cover data are only the means of sending it. Secret information as well as cover data can be any multimedia data like text, image, audio, video etc The objective of this work is to develop a Compressed Video Steganographic Scheme that can provide provable security with high computing speed, that embed secret messages into images without producing noticeable changes. Here we are embedding data in video frames. A video can be viewed as a sequence of

DOI : 10.5121/ijdms.2010.2307

still images. Data embedding in videos seems very similar to images. However, there are many differences between data hiding in images and videos, where the first important difference is the size of the host media. Since videos contain more sample number of pixels or the number of transform domain coefficients, a video has higher capacity than a still image and more data can be embedded in the video. Also, there are some characteristics in videos which cannot be found in images as perceptual redundancy in videos is due to their temporal features. Here data hiding operations are executed entirely in the compressed domain. On the other hand, when really higher amount of data must be embedded in the case of video sequences, there is a more demanding constraint on real-time effectiveness of the system. The method utilizes the characteristic of the human vision's sensitivity to color value variations. The aim is to offer safe exchange of color stego video across the internet that is resistant to all the steganalysis methods like statistical and visual analysis.

Image based and video based steganographic techniques are mainly classified into spatial domain and frequency domain based methods. The former embedding techniques are LSB, matrix embedding etc. Two important parameters for evaluating the performance of a steganographic system are capacity and imperceptibility. Capacity refers to the amount of data that can be hidden in the cover medium so that no perceptible distortion is introduced. Imperceptibility or transparency represents the invisibility of the hidden data in the cover media without degrading the perceptual quality by data embedding[4]. Security is the other parameter in the steganographic systems, which refers to an unauthorized person's inability to detect hidden data. In this Steganographic scheme secret data's are embedded the I frame with maximum scene change and macro blocks of P and B frames based on motion vectors with large magnitude. To enlarge the capacity of the hidden secret information and to provide an imperceptible stego-image for human vision, tri-way pixel-value differencing (TPVD) algorithm is used for embedding[3].

Steganography in video can be divided into two main classes. One is embedding data in uncompressed raw video, which is compressed later[1][2]. The other, tries to embed data directly in compressed video stream. The problem of the former is how to make the embedded message resist video compression. But because the video basically exists in the format of compression, the research of the latter is more significative.

2. Architecture

A steganographic algorithm for compressed video is introduced here, operating directly in compressed bit stream. In a GOP, secret data's are embedded in I frame, and in P frames and in B frames. This proposed secure compressed video Steganographic architecture taking account of video statistical invisibility .The frame work is shown in the Figure 1

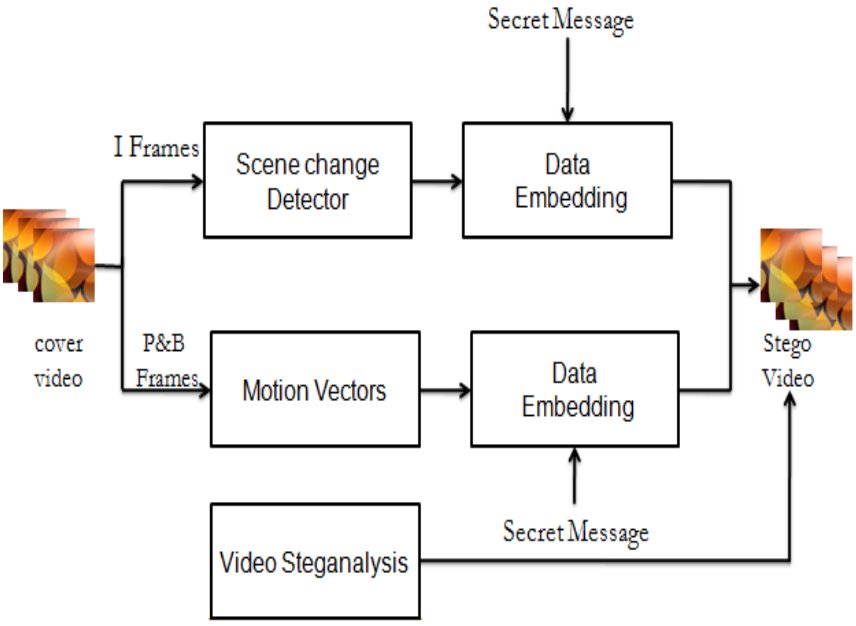

Fig.1, Block diagram of the proposed System

This architecture consists of four functions: I P and B frame extraction, the scene change detector, motion vectors calculation and the data embedder and steganalysis. The first section explains the extraction of I P and B frames from MPEG video. In the next section, scene change detector analyzes the frames with maximum scene change.[8][9] I frames in MPEG standard is coded in intra frame manner, we can obtain the DC picture with abstracting the DC coefficients from the DCT coefficient codes. Eq 4.1 describes the compare method between two conjoint I frames.

$$HD(I_i, I_{i+1}) = \sum \frac{(H_i(k) - H_{i+1}k))^2}{(H_i(k) + H_{i+1}(k)^2} \quad (1)$$

where $Ii, Ii+1$ means the i^{th} and $i+1^{th}$ I frames, Hi and $Hi+1$ are histograms of DC pictures from the i^{th} and $i+1^{th}$ I frames. $HD(Ii, Ii+1)$ is the peak value the two I frames are from different scenes, therefore the scene change point is found. Also the variances var (i) of each DC picture from I frame will be calculated. Then, data embedder, secret message is hidden into the compressed video sequence without bringing perceptive distortion. Motion vectors in P and B can be utilized for data hiding. In this proposed method data's are embedded in blocks based on motion vectors with large magnitude, Since human visual system is less sensitive to distortion in regions that are temporally near to features of high-luminance intensity , this feature can be utilized for data hiding., data are not embedded in all blocks but only in motion vectors with a magnitude above a threshold. Larger magnitude illustrates faster temporal changes and less visible degradation due to data hiding. The details of data embedding in P and B frames are as follows:

1. For each P and B frames, motion vectors are extracted from the bitstream.
2. The magnitude of each motion vector is calculated as follows:

$$| MV_j | = \sqrt{H_j^{\,2} + V_j^{\,2}} \qquad (2)$$

   where MV_j the motion vector of the j^{th} macroblock, and H_i is horizontal and V_j is the vertical components of the MV respectively.
3. This magnitude is compared with a threshold
4. Select the block with maximum magnitude and embed the data using PVD method

To increase the capacity of the hidden secret information and to provide an imperceptible stego-image for human vision, here pixel-value differencing (PVD) is used for embedding. With these four sections, we can obtain the final stego-video.PVD method is explained in next chapter

3 Compressed Video Steganographic Algorithm

Here a novel steganographic approach called tri-way pixel-value differencing with pseudo-random dithering (TPVDD) is used for embedding.[3][4] TPVDD enlarges the capacity of the hidden secret information and provide an imperceptible stego-image for human vision with enhanced security. A small difference value of consecutive pixels can be located on a smooth area and the large one is located on an edged area. According to the properties of human vision, eyes can tolerate more changes in sharp-edge blocks than in smooth blocks. That is, more data can be embedded into the edge areas than into smooth areas. This capability is made used in this

approach which leads to good imperceptibility with a high embedding rate. The Tri-way Differencing Scheme is explained as follows. In general, the edges in an image are roughly classified into vertical, horizontal, and two kinds of diagonal directions. Motivated from the PVD method, using two-pixel pairs on one directional edge can work efficiently for information hiding. This should accomplish more efficiency while considering four directions from four two-pixel pairs. This can be implemented by dividing the image into 2×2 blocks and one example block is shown in Figure 2

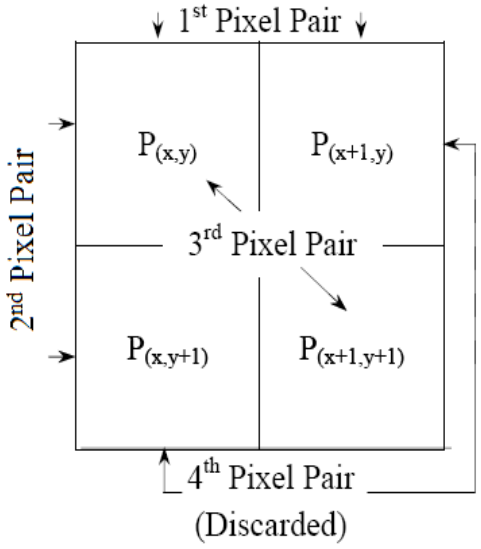

Fig.2 An Example of four pixel pair

However, since the changing of pixel values for the fourth pixel pair affects the first and the second pairs, the fourth pair is useless and has to be discarded. Therefore, we propose that three pairs are used to embed the secret data. Before introducing the proposed algorithm, the pre-procedure is to partition the cover image into non overlapping 2×2 blocks with 4 pixels. In this scheme, each 2×2 block includes four pixels of $p(x, y)$, $p(x+1, y)$, $p(x, y+1)$, and $p(x+1, y+1)$ where x and y are the pixel location in the image. Let $p(x, y)$ be the starting point, then three pixel pairs can be found by grouping $p(x, y)$ with the right, the lower, and the lower right neighboring pixels. Those three pairs are named by P_0, P_1 and P_2 where $P_0 = (p(x, y), p(x+1, y))$, $P_1 = (p(x, y), p(x, y+1))$ and $P_2 = (p(x, y), p(x+1, y+1))$ respectively. When using the tri-way PVD method to embed the secret data, each pair has its modified P'_i and a new difference value d'_i for $i = 0, 1, 2$. Now, the new pixel values in each pair are different from their original ones. That is, we have three different values for the starting point $p(x, y)$ named $p'_0(x, y)$, $p'_1(x, y)$ and $p'_2(x, y)$ from P_0, P_1, and P_2 respectively. However, only one value for $p'_i(x, y)$ can exist after finishing the embedding procedures. Therefore, one of $p'_i(x, y)$ is selected as the reference point to offset

the other two pixel values. That is, two pixel values of one pair are used to adjust the other two pairs and construct a new 2×2 block. Selecting different reference points results in varied distortion to the stego-image. Here, we propose an optimal selection approach to achieve minimum Mean-Square-Error (MSE). Suppose that $m_i=d'_i-d_i$, d_i and d'_i are the difference values of pixel pair i before and after embedding procedures. The rules that can exactly determine one optimal reference pair without really estimating MSE are introduced as follows.

1) If all values of m_i are great than 1 or smaller than −1, the optimal pixel pair $i_{optimal}$ is the pair with the greatest $|m|$.

2) If all m_i have the same sign and only one $m_i \in \{0,1,-1\}$, then the optimal pixel pair $i_{optimal}$ is selected from the other two pairs with the smallest $|m|$.

3) If only one m_i has a different sign from the other two pairs, the optimal pixel pair $i_{optimal}$ is selected from the other two pairs with the smallest $|m|$.

4) If only one $m_i \in \{0, 1, -1\}$ and the other two m_i has different signs, the optimal pixel pair $i_{optimal}$ is the pair with $m_i \in \{0, 1, -1\}$.

5) If there exists more than one pair with $m_i \in \{0, 1, 1\}$, the optimal pixel pair $i_{optimal}$ can be selected as any one pair with $m_i \in \{0, 1, 1\}$.

By following those selection rules described above, we can skip the calculation steps of MSE estimation to obtain the optimal reference pairs. Thus, the total computational complexity can be greatly reduced.

3.1 Adaptive Rules to Reduce Distortion

Although the proposed approach is feasible for embedding secret data, embedding large amount of bits can still cause serious image distortion easily. Since most distortion is generated from the offsetting process, the following two conditions are further designed to avoid too much offset described by

1) embed _ bit($P0$)≥5 and 1 embed _ bit($P1$)≥4

2) embed _ bit($P0$)<5 and 2 embed _ bit($P0$)≥6

Where embed_bit(Pi) represents the total embedding bits along the direction of Pi. If either one of above two conditions is satisfied, the current block being processed can probably result in higher distortion. Then we use two pixel pairs, $P0$ and$P3= P(x,y+1)$, $P(x+1,y+1)$and adopt the original PVD method to individually process those two pairs along one direction. If neither of the conditions is satisfied then PVD is applied to three pixel pairs $P0$, $P1$ and $P2$ in three directions. Here, we name those two conditions as “branch conditions”.

3.2 Pseudo-random Dithering

This section describes how pseudo-random dithering is applied the range of pixel differences and further modification for embedding and extraction of secret message.

Step 1: pseudo-randomly select a parameter β Є [0, 1], generated from an embedding key, for each block of two consecutive pixels, and calculate

$$l`k = lk + floor(\beta.wk) \quad (3)$$

$$u`k = lk+1 - 1 \quad (4)$$

where k is a range index. Thus, instead of the fixed ranges as used in the original PVD method, the new ranges are defined by the varied $l`k$ and $u`k$. In other words, the ranges corresponding to different blocks are differently defined according to a secret key. Because $wk \leq wk+1$ *u0*

$$u`k - l`k = lk+1+floor(\beta.wk+1)-1-lk-floor(\beta.wk) \geq wk-1 \quad (5)$$

Eqn. (5) indicates that the width of any varied range is no less than that of the original fixed range. If $l`k \leq |d| \leq u`k$, a total of log2(wk) secret bits are embedded into the corresponding block. Convert the secret bits into a decimal value b, and calculate

$$d' = \begin{cases} |e-d| \text{ for } d \geq 0 \text{ and mod } (e, wk) = d \\ -|e-d| \text{ for } d < 0 \text{ and mod } (e, wk) = -d \end{cases}$$

Where $l`k \leq e \leq u`k$

On the extraction side, b can be restored simply by

$$b=mod(d',wk) \quad (6)$$

Note that if b values in all the blocks are 0, the proposed approach degenerates to the original PVD method and the steps in pixel difference histogram will reveal the presence of hidden data. Nonetheless occurrence of such a case is highly unlikely.

3.3 The Embedding Algorithm

The details of data hiding steps are described as follows.

1)Calculate four difference values $di,(x,y)$ for four pixel pairs in each block given by

$$d_{0,(x,y)} = P_{(x+1,y)} - P_{(x,y)}$$

$$d_{1,(x,y)} = P_{(x,y+1)} - P_{(x,y)}$$

$$d_{2,(x,y)} = P_{(x+1,y+1)} - P_{(x,y)}$$

$$d_{3,(x,y)} = P_{(x+1,y+1)} - P_{(x,y+1)}$$

2) Using $|d_{i,(x,y)}|$ (i=0,1,2,3) to locate a suitable $R_{k,i}$ in the range table designed ,that is to compute j = min ($u_k - |d_{i,(x,y)}|$) where $u_k \geq d_i$ for all $1 \leq k \leq n$. Then $R_{k,i}$ is the located range.

3) Compute the amount of secret data bits t_i that can be embedded in each pair by $R_{j,i}$. The value t_i can be estimated from the width $w_{j,i}$ of $R_{j,i}$, this can be defined by $t_i = \log_2 w_{j,i}$.

4) If t_i of P_i (i=0,1,2,3) satisfies branch conditions, two pixel pairs P_0 and P_3 are processed using original PVD. But new difference d'_i is to calculate. Otherwise, the proposed tri-way scheme is used to process P_i.

5) Read t_i bits from the binary secret data and transform the bit sequence into a decimal value b_i.

6) Calculate the new difference value $d'_{i,(x,y)}$

7) Modify the values of p_n and p_{n+1} by the following formula:

$$(p'_n, p'_{n+1}) = (p_n - \text{ceil}(m), p_{n+1} + \text{floor}(m)) \quad (7)$$

Where (p_n, p_{n+1}) represent two pixels in P_i and $m = (d'_i - d_i)/2$

8) Using the selection rules to choose the optimal reference point $p'_{i(x,y)}$ with minimum MSE, then this selected point is used to offset the other two pixel pairs.

9) Now, the new block constructed from all pixel pairs and embedded with secret data is generated.

3.4 The Extraction Algorithm

To retrieve the embedded secret data from the stego-image, the extraction algorithm is described in the following steps.

1) Partition the stego-image into 2x2 pixel blocks, and the partition order is the same as that in the embedding stage.

2) Calculate four difference values $d^*_{i,(x,y)}$ for four pixel pairs in each block given by

$$d^*_{0,(x,y)} = P_{(x+1,y)} - P_{(x,y)}$$

$$d^*_{1,(x,y)} = P_{(x,y+1)} - P_{(x,y)}$$

$$d^*_{2,(x,y)} = P_{(x+1,y+1)} - P_{(x,y)}$$

$$d^*_{3,(x,y)} = P_{(x+1,y+1)} - P_{(x,y+1)}$$

3) Using $|d^*_{i,(x,y)}|$ (i=0,1,2,3) locate a suitable $R_{k,i}$. Also find the number of bits t_i that was embedded. If t_i satisfies the branch conditions, two independent pixel pairs are selected.Otherwise, three pixel pairs are used for further processing.

4) The secret message b^* is to calculate for stegoimage is not altered b^* is same as b. Finally b^* is converted to binary to obtain the original secret message.

4 Experimental Result

To demonstrate the accomplished performance of our proposed approach in capacity and security for hiding secret data in the stego-image, we have also conducted different experiments using different videos.According to the invisibility benchmark for the watermarked images , a minimum peak signal-to noise ratio (PSNR) value of 38 dB is adopted as the quality requirement for the stego-images The goal in objective image quality assessment is to develop quantitative measures that can automatically predict perceived frame quality. The simplest and most widely used full-reference quality metric is the mean squared error (MSE), computed by averaging the squared intensity differences of distorted and reference image pixels, along with the related quantity of peak signal-to-noise ratio (PSNR). These are appealing because they are simple to calculate, have clear physical meanings, and are mathematically convenient in the context of optimization. But they are not very well matched to perceived visual quality. In the last three decades, a great deal of effort has gone into the development of quality assessment methods that take advantage of known characteristics of the human visual system (HVS). The size of all cover frames is 358x288. Here, PSNR value is utilized to evaluate the invisibility of the stego-frames. steganography capacity is the maximum message size that can be embedded subject to certain constraints.

4.1 Text Embedding in video

In figure 3,4 and 5 we show the I P and B frames before and after embedding. Here Text data's are the secret information .In the table 2,3 and 4 shows PSNR values and capacity of I P and B frames .we have randomly taken I , P and B frames and after embedding in those frames the PSNR values are still above 40 db. Results shows that quality is preserved in stego I P and B frames. Table 6.1 shows the information about cover video .In I frames 20 to 25 % of the Cover I frames are embedded, in B frames 10 to 15 % of the Cover B frames are embedded and in P frames 15 to 20 % of the Cover P frames are embedded.

Table 1 Cover video file information

Name	resolution	Frame/sec	No of frames	Size(KB)
Boat.mpg	288×352	15	150	572

Figure 3 I frame before and after embedding

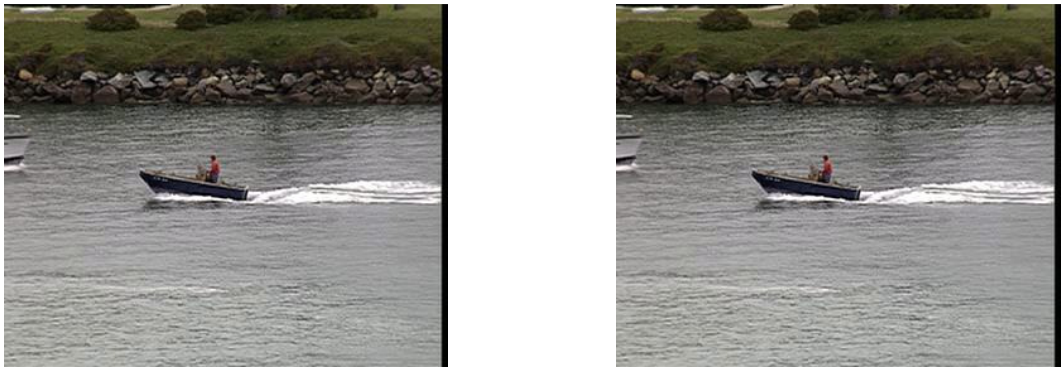

Figure 4 Bframe before and after embedding

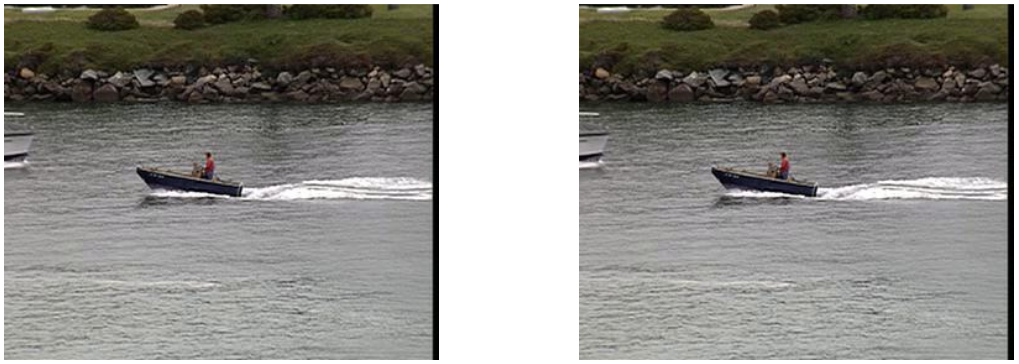

Figure 5 P frame before and after embedding

Table 2 Capacity and PSNR values of stego I frames

Stego I frames	PSNR	Capacity(bytes)
I_1	**65.1**	**6243**
I_{23}	**62.5**	**5234**
I_{60}	**61.3**	**5140**

Table 3 Capacity and PSNR values of B frames

Stego B frames	PSNR	Capacity(bytes)
B_3	**61.1**	**1234**
B_6	**60.4**	**1356**
B_9	**62.4**	**1543**

Table 4 Capacity and PSNR values of P frames

Stego P frames	PSNR	Capacity(bytes)
P_5	**62.8**	**2543**
P_8	**61.4**	**3423**
P_{14}	**63.6**	**3245**

5 Histogram Analysis

An important digital image tool is the histogram. A histogram is a statistical representation of the data within an image that shows how many pixels there are with each of the possible values. An image and its histogram are shown below figure 6,7and8. The histogram is a bar graph where each entry on the horizontal axis is one of the possible values that a pixel can have. In an 8-bit image, those values range from 0 to 255. Each vertical bar in the graph indicates the number of pixels of that value. The sum of all vertical bars is equal to the total number of pixels in the image. Usually, the absolute value of each vertical bar, or number of pixels at a specific value, is not important. The histogram analysis shows that there is no deviation in histogram before and after embedding.

5.1 Histogram Analysis of I frames

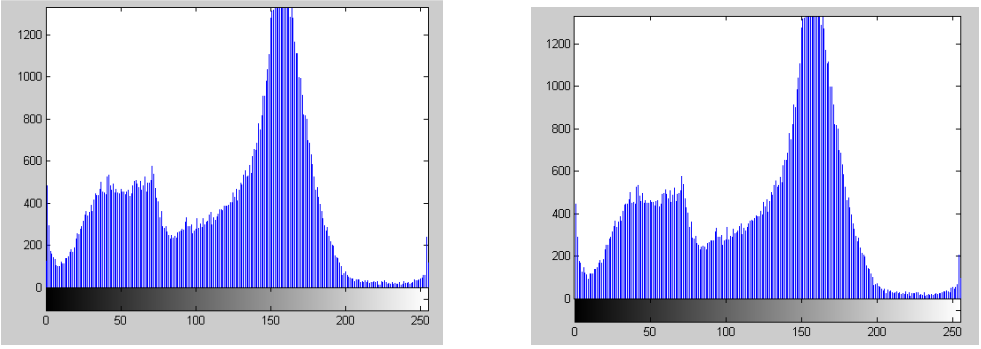

Figure 6 Histogram of I frame Before and After embedding

5.2 Histogram Analysis of B frames

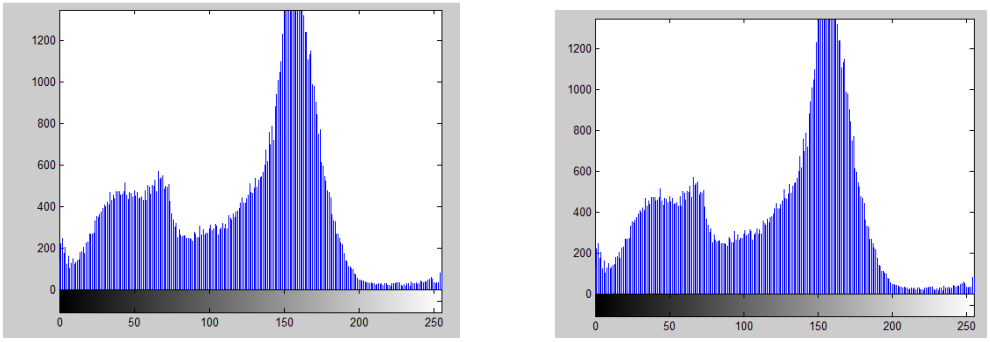

Figure 7 Histogram of B frame before and after embedding

5.3 Histogram Analysis of P frame

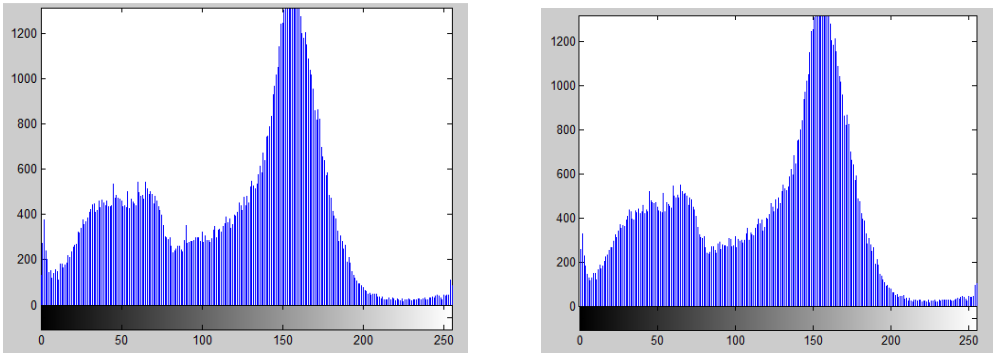

Figure 8 Histogram of P frame before and after embedding

6 Conclusion

A new Video Steganographic Scheme was proposed in this paper, operating directly in compressed domain. For data hiding tri-way pixel-value differencing (TPVD) algorithm has been used. This algorithm provides high capacity and imperceptible stego-image for human vision of the hidden secret information. Here I frame with maximum scene change blocks were used for embedding. The performance of the steganographic algorithm is studied and experimental results shows that this scheme can be applied on compressed videos with no noticeable degradation in visual quality.

7 References

[1] F Hartung., B. Girod.: Watermarking of uncompressed and compressed video, Signal Processing, Special Issue on Copyright Protection and Access Control for Multimedia Services, 1998, 66 (3): 283-301.

[2] Bin Liu., Fenlin Liu., Chunfang Yang and Yifeng Sun.: Secure Steganography in Compressed Video Bitstreams,The Third International Conference on Availability,Reliability and Security,2008

[3] Ko-Chin Chang., Chien-Ping Chang., Ping S. Huang., and Te-Ming Tu,: A Novel Image Steganographic Method Using Tri-way Pixel-Value Differencing, Journal of Multimedia , VOL. 3, NO. 2, JUNE 2008

[4] Y. K. Lee., L. H. Chen.: High capacity image steganographic model, IEE Proceedings on Vision, Image and Signal Processing, Vol. 147, No.3, pp. 288-294, 2000.

[5] D.-C. Wu., and W.-H. Tsai.: A steganographic method for images by pixel-value differencing, Pattern Recognition Letters, Vol. 24, pp. 1613–1626, 2003

[6] Y. J. Dai., L. H. Zhang and Y. X. Yang.: A New Method of MPEG VideoWatermarking Technology. International Conference on Communication Technology Proceedings (ICCT), 2003.

[7] G. C. Langelaar and R. L. Lagendijk.: Optimal Differential Energy Watermarking of DCT Encoded Images and Video. IEEE Trans. on Image Processing, 2001, 10(1):148-158.

[8] Bin Liu, Fenlin Liu, Chunfang Yang and Yifeng Sun, "*Secure Steganography in Compressed Video Bitstreams"* " Proc of the Int. Conf. IEEE ARS ,pp 520-525,2008

[9] A.Hanafy,Gouda I.salama and Yahya Z.Mohasseb, "*A Secure Covert Communication model Based On Video Steganography"* Proc of the Int. Conf. IEEE Military Communication,2008