Secure Data Hiding in H.264/AVC10 Video using Intra Pulse Code Modulation Macro Block

Yogita Raskar
Post Graduate Student
Department of Electronics &
Telecommunication
Engineering,
Sinhgad Academy of
Engineering Pune, India

ABSTRACT

H.264 is well known compression standard for raw video. Data hiding using general techniques will not sustain under this scheme. We propose a data hiding approach in IPCM block. Where these IPCM encoded macro blocks are modified in order to hide the desired data. In general video, the numbers of IPCM blocks generated are very less. A method is proposed to increase the number of IPCM blocks generated in H.264 algorithm, by which more data can be hidden. It is a blind data hiding scheme, i.e. the encoded message can be extracted directly without the need of the original host video.

Keywords

H.264/AVC, Intra prediction, IPCM, Data hiding

1. INTRODUCTION

H.264 is a next-generation video compression format. H.264 is also known as MPEG-4 AVC , developed for use in high definition systems such as HDTV, Blu-ray and HD DVD as well as low resolution portable devices such as Sony's PSP and Apple's iPod. H.264 offers better quality at lower file sizes than both MPEG-2 and MPEG-4 ASP. Apple has officially adopted H.264 as the format for QuickTime. The H.264/AVC standard was designed for high compression efficiency, error resilience and flexibility so that it could support a wide variety of applications and different transport environments such as wired and wireless networks. The video color space used by H.264/AVC separates a color representation into three components called Y, C_b and $C_{r.}$ Component Y is called luma, and represents brightness. The two chroma components Cb and Cr represent the extent to which the color deviates from gray toward blue and red, respectively.

High speed computer networks, the Internet and the World Wide Web have revolutionized the way in which digital data is distributed. The widespread and easy accesses to multimedia contents and possibility to make unlimited copy without loss of considerable fidelity have motivated the need for digital rights management. Digital watermarking is a technology that can serve this purpose. Watermarking is the addition of irremovable data to multimedia content for the purpose of authentication, copy identification and tracking. Watermarking valuable and sensitive images such as medical, military and art work images gives a big challenge to technique used in the most watermarking methods. Data hiding is also equally used like watermarking in order to hide the data & send securely. Data hiding and watermarking techniques are usually studied together because a watermarking technique can serve as a data hiding technique, as well, although the opposite is not always feasible. Early video data hiding approaches [2, 11] were essentially still image watermarking techniques extended to video by hiding the message in each frame independently.

Data hiding deals with the ability of embedding data into a digital cover with a minimal amount of perceivable degradation. Data hiding consist of two set of data: 1. Cover medium and 2. Embedding data. Either of the set used for data hiding can be text, audio, picture or video depending on the size of the message and the capacity of the cover. When data is hidden & send securely, motion vector is calculated at encoder side to remove temporal or spatial redundancy. The H.264/AVC standard uses the spatial correlation property through an Intra prediction. A macroblock of interest can be predicted from block, generally located above and to the left side of the block, since they have already been encoded and reconstructed. The use of IPCM for data hiding is indeed motivating because it can preserve the video quality perfectly.

2. INTRA PULSE CODE MODULATION IN H.264/AVC

Data hiding in H.264 makes use of three different types of predictions like inter prediction, intra prediction & IPCM blocks to hide the data. In inter prediction mode, motion vector can be calculated between the two frames, In case of intra prediction mode, motion vector can be calculated within the frame with left & top side of the frame. Third type of prediction is using IPCM block. In IPCM block, data can be sent without compression. i.e. loss of data due to compression is decreased because in case of inter or intra prediction, data compression at the encoder side & data decompression at the decoder side results in lot of data loss during encoding & decoding process. But in case of IPCM block, data can be sent without compression i.e. at the decoder side same data as that of original one, this means that possibility of loss of data is completely reduced. But one problem faced while using IPCM blocks is rareness of IPCM blocks. Although many researchers have implemented data hiding techniques based on intra-prediction mode of H.264 video [6], [7], [9] but Kapotas et al [1], [8] was to first to propose it using IPCM macroblocks. Data hiding based on motion vectors has been proposed [3], [5] but this suffers from the basic limitation of low embedding capacity, as lot of overhead information is to be embedded to make the watermark data safe.

The basic methodology for frame (picture) is that it is partitioned into fixed-size macroblocks, which covers a rectangular picture area of 16X16 samples of the luma components and 8X8 samples of each of the two chroma components. All luma and chroma samples of a macroblock are either spatially or temporally predicted, and the resulting prediction residual is encoded using transform coding. For transform coding purposes, each color component of the prediction residual signal is subdivided into smaller 4X4 blocks.

3. THE PROPOSED METHOD FOR SECURE DATA HIDING

In order to hide the data using IPCM block in which data is directly processed without compression or without any loss of data. The proposed method of data embedding based on IPCM block is a secure one. Secondly we check the robustness of hidden data & thirdly we perform capacity estimation of the data hiding space available. When IPCM blocks are used to hide the data lower bits of luma and chroma samples are used. Because of change in lower bits of luma and chroma it does not affect the image or video much more. But regarding the IPCM blocks there is one practical problem i.e. rareness of IPCM blocks.

In data hiding process, number of IPCM blocks required to hide the data are less than the number of IPCM blocks required to forcefully generate the IPCM blocks.

Data hiding at the encoder side is shown in Figure 1, where raw video is taken as input to the encoder which is passed at the same time when embedded data is passed to the encoder, so that data can be hidden at the encoder side & converted to the H.264 bit stream. Output of encoder is passed to the decoder where hidden data can be extracted and raw video can be reconstructed back.

Data extraction during decoding is shown in figure 2, where input to the decoder is in the form of H.264 bit stream. Decoder will first find IPCM block. If IPCM block is present then extract hidden data from the IPCM block as well as reconstruct original input frame as it is without loss at the decoder side.

In order to hide the data in the H.264 encoder, we pass input to the mode decision. Mode decision will find out the type of prediction (inter prediction or intra prediction or IPCM) and hide the data in IPCM blocks. If the type of prediction is inter prediction, the motion estimation and compensation have become powerful techniques to eliminate the temporal redundancy due to high correlation between consecutive frames. Because two successive frames of a video sequence often have small differences, it reduces the temporal redundancy.

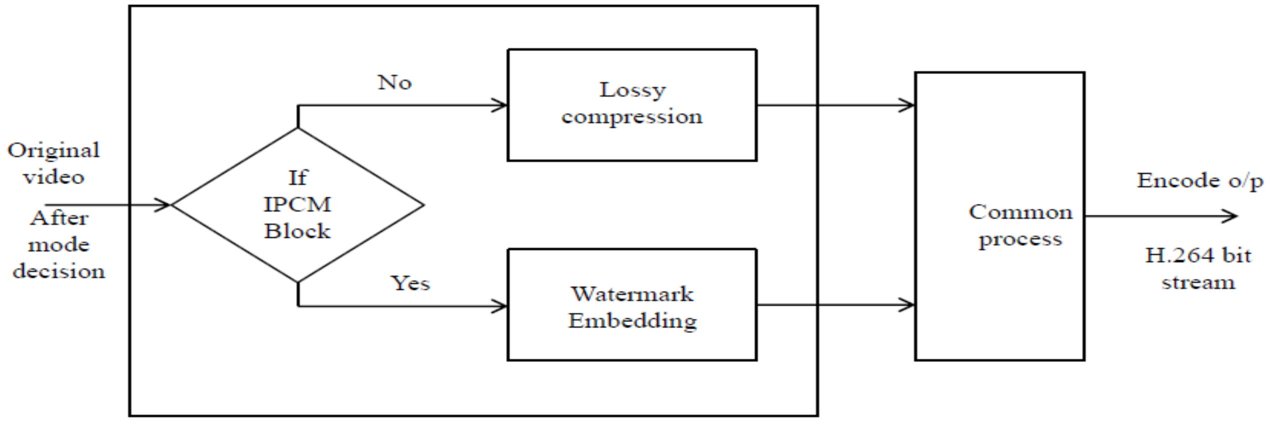

Fig 1: Data hiding during the encoding

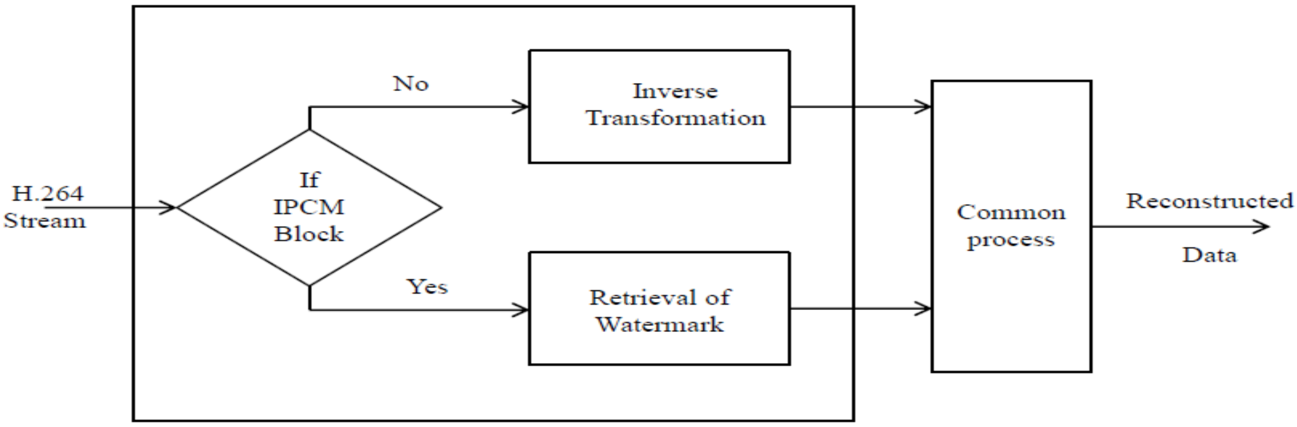

Fig 2: Data extraction during decoding.

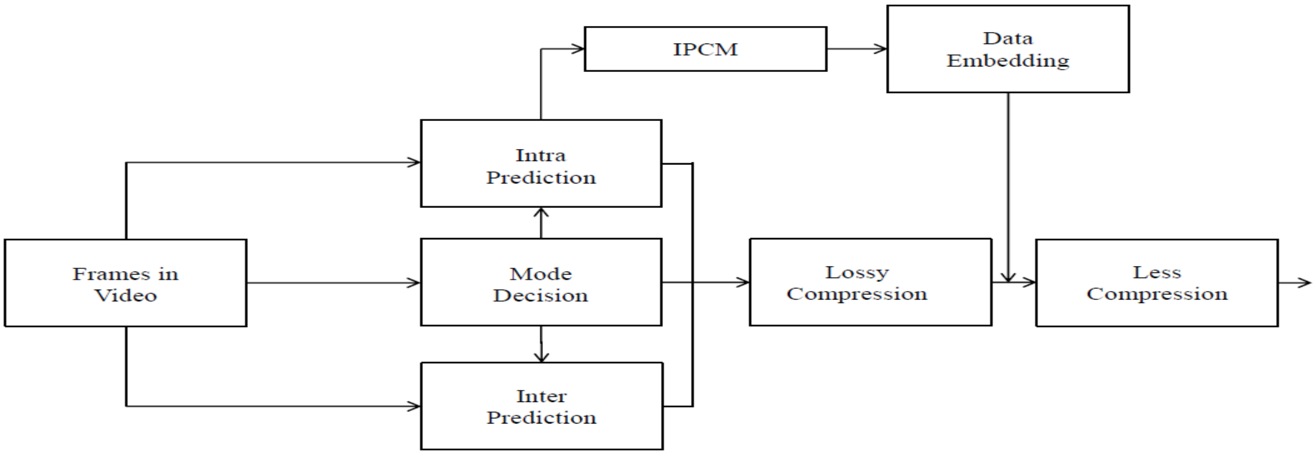

Fig 3: Block diagram of the data hiding within the H.264 encoder [1].

The process of block matching is the most time consuming one during encoding. In order to find a matching block, each block of the current frame is compared with a past frame within a search area.

If the type of prediction intra prediction has been conducted in the transform domain intra prediction in H.264/AVC is always conducted in the spatial domain, by referring to neighboring samples of previously-coded blocks which are to the left and/or above the block to be predicted.

If the encoder succeeds to find a matching block on a reference frame, it will obtain a motion pointing to the matched block and a prediction error. Using both the elements, the decoder will be able to recover the raw pixels of the block.

Some data may be lost due to compression; such type of data can be seen as it is without compression. For sending such type of data IPCM blocks are used. When IPCM blocks are used there is no need of transformation and quantization.

In order to minimize the number of expensive motion estimation calculations, they are only calculated if the diffference between two blocks at the same position is higher than a threshold, otherwise the whole block is transmitted. In H.264/AVC, two methods of entropy coding are supported.

The simpler entropy coding method uses a single infinite-extent code word table for all syntax elements except the quantized transform coefficients. For transmitting the quantized transform coefficients, a more efficient method called Context-Adaptive Variable Length Coding (CAVLC) is employed. In the CAVLC entropy coding method, the number of nonzero quantized coefficients (N) and the actual size and position of the coefficients are coded separately.

The proposed algorithm for mode decision is given below:

3.1 Algorithm for mode decision

1. Calculate Total number of IPCM blocks already present, ipcm_cnt

2. Calculate Total number of IPCM blocks required to hide the data, Hide_no.

3. If ipcm_cnt<hide_no

a. Then find the extra ipcm blocks required, req_blks

b. For every req_blks number of block, forcefully create a IPCM block

4. Repeat this for each block of the frame

Decision making algorithm will help to find out total number of IPCM blocks available & number of blocks required for hiding the data. Depending upon the number of required blocks, forceful generation of IPCM blocks is done. Using IPCM blocks, data can be send without compression as well as without loss.

For this purpose, data embedding algorithm is given below.

3.2 Algorithm for Data Embedding

1. Partition the entire image into pairs of pixels (for instance, on rows, on columns).

2. Get the image data and txt data.

3. Convert txt data into a linear binary vector.

4. Convert the image data array into a single column vector.

5. If pixel value is odd

Make it even

6. Else

Leave it as such

7. Modifies the low bits of the values of the macroblocks / pixel image data in such a way that the modified bits form the hidden data.

8. Reshape the pixel image data into a block array

3.3 Data Embedding Capacity

After data embedding process, embedding capacity can be find out. The data capacity of a video sequence (YUV 4:2:0) is

Data capacity per IPCM block (Dc)

$$Dc = \text{Luma capacity} + (2 \times \text{Chroma capacity}) \quad (1)$$

Where,

Luma capacity = $256 \times NIPCM \times Lbits$,

Chroma capacity = $2 * 64 \times NIPCM \times Cbits$, NIPCM = the number of the IPCM macroblocks used for data hiding,

Lbits=the number of the low bits per IPCM luma sample used for data hiding,

Cbits = the number of the low bits per IPCM chroma sample used for data hiding.

The luma and chroma samples are 256 and 64 respectively.

After finding data capacity per IPCM block, find the number of IPCM blocks required to hide the data. If more IPCM

blocks are required to generate the IPCM block forcefully to hide the data.

Embedding capacity per frame (EC)

EC = Data Embedding capacity per macro block × Number of macroblocks. (2)

When IPCM blocks are forcefully generated, find the embedding capacity per frame using above formula. After finding embedding capacity per frame, total video capacity can be easily obtained by multiplying number of frame.

Total video capacity (TVC)

TVC= Embedding per frame × number of frame (3)

3.4 PSNR of hidden data

The data hiding scheme should resist destruction from image processing and malicious attacks. When the image quality does not get seriously harmed or addition of the watermark must not degrade the content in a perceptible way. From the PSNR value robustness of given video after data hiding can be calculated.

PSNR= 10 log10 (max_value / sqrt (mean square error)) (4)

PSNR can be calculated between original frame and decoded frame after data retrieval. Calculate PSNR per frame and take average of the entire frames to get PSNR for video.

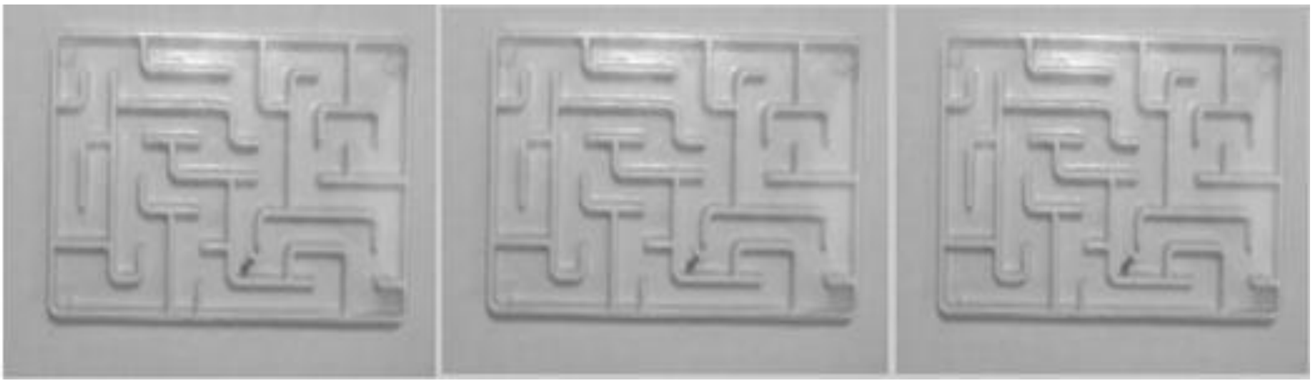

Figure 5: Sample frame of Ant maze in that (a) original image (b) Decoded image with hidden data (c) Decoded image after data retrieval

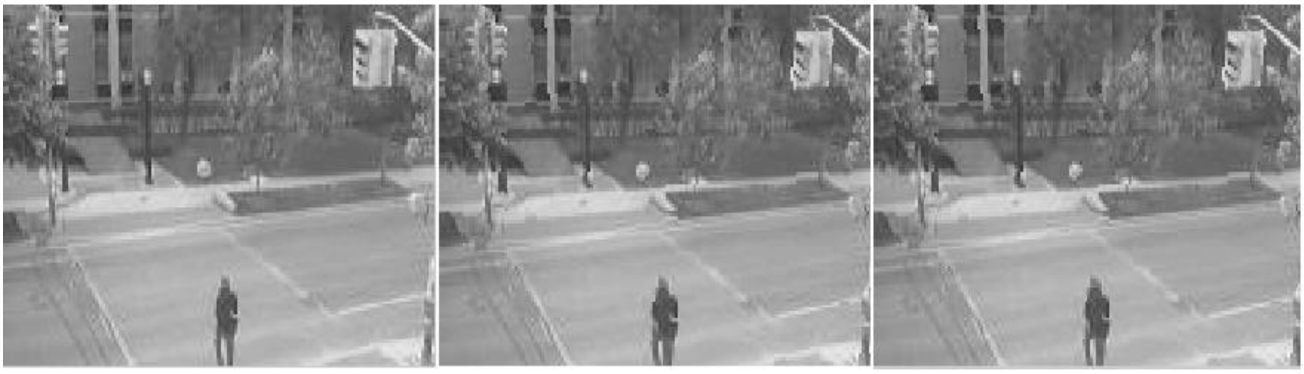

Figure 6: Sample frame of Man crossing street in that (a) original image (b) Decoded image with hidden data (c) Decoded image after data retrieval

Table .1 PSNR Values for Data Hiding in Different LSB

No. Of IPCM Block required for one Video	**Data hidden in**	**Video name**		
		Ant maze	**Man crossing street**	**Vehicle on street**
6	Last 1 LSB PSNR Value	51.5863	51.1078	51.214
	File Size	6720	7680	8640

3	Last 1 LSB PSNR Value	43.0596	42.872	43.6994
	File Size	6720	7680	8640
2	Last 1 LSB PSNR Value	36.4731	36.9606	32.2645
	File Size	6720	7680	8640

4. RESULTS AND DISCUSSION

Data hiding using proposed method was done and the PSNR of a sample video (ant maze) was calculated. Data hiding was done in different LSBs is mentioned in table 1. Data hiding capacity means how much data can be hidden in that video. Using encoder, data was be hidden in the series of frames (video) and hidden data was extracted without degradation at the decoder side.

For a fixed amount of data(6720 bits in case of Ant maze video), we choose last one LSB and for this we had to force 6 IPCM macroblock, the same data was hided using 3 IPCM macroblocks where its last two LSBs were used. And only 2 IPCM MBs were invoked using its last three LAB planes. But it was observed that we get a poor frame quality (reduced PSNR-36.47dB) when we did data hiding in last three bits. On the other hand we get a good PSNR value when only one LSB was used, but this resulted in a encoded video of larger size (as we invoked 6 IPCM macroblocks). The same is true for other two videos also (Man Crossing street and Vehicle on street).

5. CONCLUSION

Data hiding for H.264 video format has been done. IPCM blocks are used for this purpose. To generate more IPCM blocks, the Mode Decision Block of H.264 has been tweaked with the generated IPCM blocks, data is hidden and transmitted. When more data is to be hidden, more IPCM blocks are generated which makes the compression to be ineffective. An Ideal amount of data has been taken and the new algorithm has been tested on four different videos. The result so obtained gives choice to user to take advantage of either PSNR or Video Size.

6. REFERENCES

[1] Spyridon K. Kapotas, Athanassios N. Skodras, “Real time data hiding by exploiting the IPCM macroblocks in H.264/AVC streams” , Springer Journal of Real Time Image Processing, Volume 1, Issue 4,March 2009.

[2] Chae, J.J., Manjunath, B.S.: Data hiding in video. In: IEEE Proceedings of International Conference on Image Processing (ICIP), pp. 243–246 (1999)

[3] Sarkar, A., Madhow, U., Chandrasekaran, S., Manjunath, B S.: Adaptive MPEG-2 video data hiding scheme. In: Proceedings of SPIE Security, Steganography, and Watermarking of Multimedia Contents IX, January 2007

[4] Gary J. Sullivan, Pankaj N. Topiwala, and Ajay Luthra, “The H.264/AVC advanced video coding standard: overview and introduction to the fidelity range extensions,” Proc. SPIE, vol. 5558, 454-476, 2004.

[5] Fang, D.-Y., Chang, L.-W.: Data hiding for digital video with phase of motion vector. In: IEEE Proceedings of International Symposium on Circuits and Systems (ISCAS), May 2006

[6] Cao, H., Zhou, J., Yu, S.: An implement of fast hiding data into H.264 bitstream based on intra-prediction coding. Proc. SPIE 6043, 123–130 (2005)

[7] Hu, Y., Zhang, C., Su, Y.: Information hiding based on intra prediction modes for H.264/AVC. In: IEEE International Conference on Multimedia and Expo (ICME), Beijing, China, July 2–5, 2007

[8] Kapotas, S.K., Varsaki, E.E., Skodras, A.N.: Data hiding in H.264 encoded video sequences. In: IEEE International Workshop on Multimedia Signal Processing (MMSP), Chania, Greece, October 1–3, 2007

[9] Kim, S.M., Kim, S.B., Hong, Y., Won, C.S.: Data hiding on H.264/AVC compressed video. In: Proceedings of ICIAR 2007. LNCS, vol. 4633, pp. 698–707 (2007)

[10] Wiegand, T., Sullivan, G.J., Luthra, A.: Draft ITU-T Recommendation H.264 and Final Draft International Standard 14496- 10 AVC, JVT of ISO/IEC JTC1/SC29/WG11 and ITU-T SG16/ Q.6, Doc. JVT-G050r1, Geneva, Switzerland, May 2003.