DATA HIDING IN VIDEO USING EFFICIENT QUANTIZATION SCALE AND LOW DISTORTION TRANSFORM

Berdin N Prabhu Dhas D.S

M.Vellathurai

Abstract—

This paper proposes two data hiding approaches using compressed MPEG video. The first approach hides message bits by modulating the quantization scale of a constant bit rate video. A payload of one message bit per macro block is achieved. A second order multivariate regression is used to find an association between macro block-level feature variables and the values of a hidden message bit. The regression model is then used by the decoder to predict the values of the hidden message bits with very high prediction accuracy. The second approach uses the flexible macro block ordering feature of H.264/AVC to hide message bits .Macro blocks are assigned to arbitrary slice groups according to the content of the message bits to be hidden. A maximum pay load of three message bits per macro block is achieved. In Recent environment Distortion is a major problem and also multi embedding is a problem. In the proposed system, a distortion less technique with the help of prediction based filter is used. In normal prediction based filter has less security problem and retrieval problem due to no change in prediction context to overcome this embedding is done not only into the current pixel but also into its prediction context. Comparisons with previous work reveal that the proposed solutions are superior in terms of message payload while causing less distortion and compression overhead.

Index terms: **Data hiding, flexible macro block ordering,**

MPEG coding, multivariate regression, steganography, prediction based filter.

I . INTRODUCTION

Data hiding techniques can be used to embed a secret message into a compressed video bit stream for copy right protection, access control, content annotation and transaction tracking. Such data hiding techniques can also be used for other purposes. For instance, used for data hiding techniques to assess the quality of compressed video in the absence of the original reference. The quality is estimated based on computing the degradations of the extracted hidden message. Data hiding can be used for real time scene change detection in compressed video. The information is hidden using the motion compensation block sizes of an H.264/AVC video. Data hiding is also used for error detection and concealment in applications of video transmission. Edge orientation information and number of bits of a block are hidden in the bit stream for that purpose. This paper has two method for hiding secret message into the video file. In the first solution, the message bits are hidden by modifying the quantization scale of MPEG video coded with constant bit rates. Features are extracted from individual macro blocks and a second-order regression model is computed. The decoder uses the regression model to predict the content of the hidden message based on macro block-level feature variables. In the second solution, both constant and variable bit rate coding are supported. The solution utilizes the flexible macro block ordering (FMO) feature of H.264/AVC video for message hiding and extraction. It is shown that both solutions can hide messages at an average payload of around 10 and 30 kb/s, respectively. Therefore, the applications of such solutions are not restricted to copyright protection where few bits are hidden per frame. In Recent environment Distortion is a major problem and also multi embedding is a problem. In the proposed system, a distortion less technique with the help of prediction based filter is used. In normal prediction based filter has less security problem and retrieval problem due to no change in prediction context to overcome this embedding is done not only into the current pixel but also into its prediction context. The proposed solution tells about Transformation: Here the prediction providing change to neighbors with less distortion in Transformation. In Reversible Transformation retrieve data by removing the change in neighbors. It gives more security and easy retrieve. But here it change neighbors' so there may be some chance for distortion. Our technique will eradicate the distortion with proper change in neighbor.

The rest of the paper deals with the following sections: Section II introduces message hiding using quantization scale modulation and multivariate regression. Section III introduces message hiding using FMO. Experimental results and comparisons with existing work are reported in Section IV. Section V tells about proposed work. Lastly, Section VI concludes the paper.

II. MESSAGE HIDING USING QUANTIZATION SCALE MODULATION

To hide a message using quantization scale modulation, the message is first converted into a binary stream of bits.During the MPEG encoding of individual macroblocks, the message bits are read one at a time. For each coded macroblock, the quantization scale is either incremented or decremented based on the corresponding message bit. Clearly, if the original quantization scale was either the lowest or largest allowable values then no modification is applied. This simple process of hiding a message bit in a macroblock is illustrated in Fig. 1.

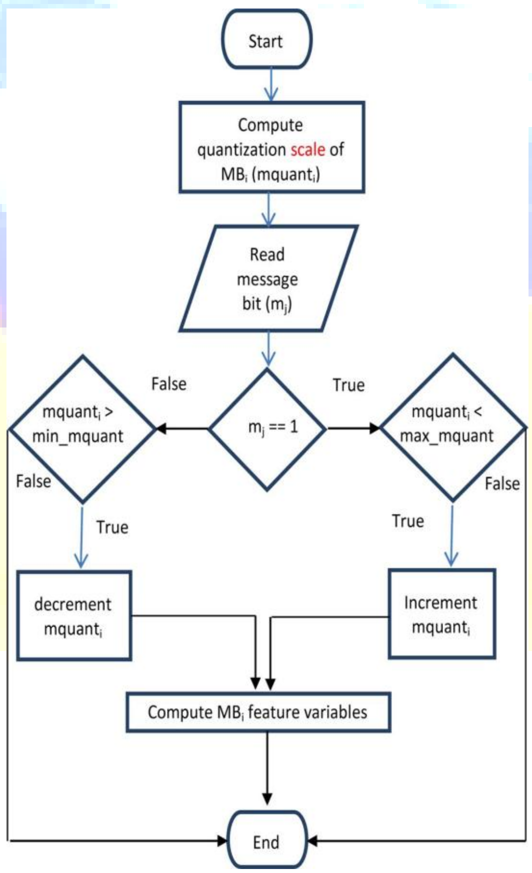

Fig1 : Message insertion flowchart for one macroblock.

Although the message hiding procedure is straightforward, nonetheless, the question that remains is how to extract the message from the bitstream. This problem can be solved by extracting macroblock-level feature variables during the encoding process. Once thewholemessage is hiddenwe end up with a feature matrix and a message vector. We will then treat the feature matrix as predictors and the message bits as a response variable and use multivariate regression to compute a prediction model. Once computed, the prediction model can be used to predict the message bit hidden in a given macroblock based on its feature variables.

A. *Macroblock Level Features Variables*

The following feature variables are extracted or computed from a MPEG-2 video stream for each coded macroblock.

1) The first feature is the virtual buffer discrepancy from uiform distruibution model.

$$d_j^t = d_0^t + B_{j-1} - (T_{t * j-1} / \#MBs)$$

This discrepancy is computed where the subscript j, #MBs indicates a macroblock index, indicates the total number of macroblocks in a video frame and t indicates the frame type; I, P, or B. d_0^t is the initial buffer fullness at the beginning of coding a frame. It is calculated as the accumulated differences between the actual number of coded frame bits minus the target number of frame bits . d_0^t is updated after the encoding of each video frame. Additionally, B_{j-1} indicates the number of bits spent on coding the previous macroblocks in the current frame. Lastly, T_t indicates the target number of bits in the current Group of Pictures (GoP). The computation depends on the overall bitrate and frame rate, it also depends on number of bits used for coding the previous frames in the same GoP, the remaining number of P and B frames in the current GoP and the average quantization scale of the previous frames in

the same GoP. It can be concluded that the virtual buffer discrepancy from uniform distribution model can be recalculated at the decoder for each macroblock. Note that the video bitrate, the frame rate, horizontal and vertical image size are all part of the video sequence header. Hence, provided that the GoP structure is known, the decoder can use this information and keep track of the number of bits spent on previous frames and previous macroblocks to compute the value of the virtual buffer discrepancy. Assuming that the GoP structure is unknown, which is unlikely, the bit stream can be scanned ahead of computing the virtual buffer discrepancy to figure out the total number of P and B frame in a GoP.

2) The second feature is the spatial activity of the underlying macroblock. This activity is computed from the four original (i.e., noncoded) luminance blocks of the current macroblock. It is computed using

$$act_j = 1 + \min(vb1, vb2, vb3, vb4)$$

where the subscript j indicates a macroblock index. The variables vb1,vb2,vb3,vb4 indicate the spatial variance of each luminance block in a frame-based coding.

3) The third feature is the actual quantization scale of the current macroblock. This scale is available from the macroblock header in the video bit stream.

B. Message Prediction

The message prediction problem is formulated using a second-order multivariate regression. The response variable in this case is the message binary bits denoted by the vector. As mentioned previously, each macroblock has three feature variables, consequently, the predictors or the feature vectors of macroblocks are arranged into one matrix which is referred to as the feature matrix.

C. Message Extraction

To extract the hiddenmessage from a coded video, the feature variables of each macroblock are computed and/or extracted from the bitstream.

TABLE I

EXAMPLE MACROBLOCK-LEVEL FEATURE VARIABLES

Feature	MB_i	MB_{i+1}
Buffer occupancy	9.903725	0.580925
MB spatial activity	0.741272	9.870142
Quantization scale	9	5

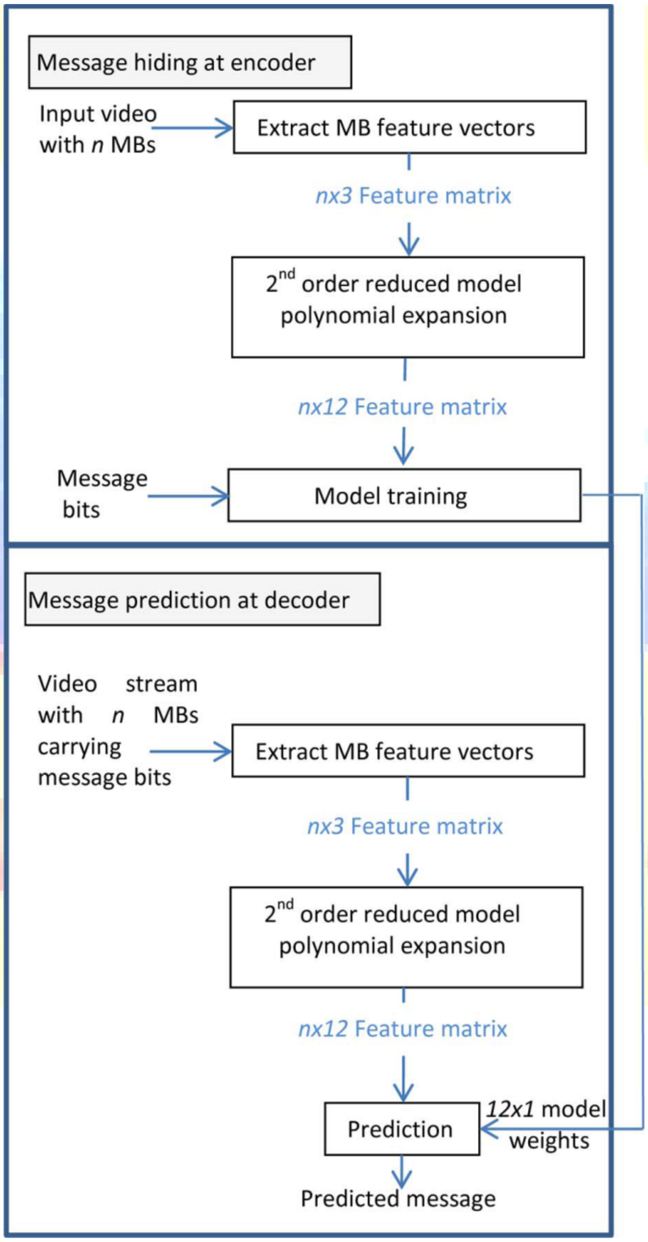

Fig. 2. Block diagram of message hiding and prediction.

III. MESSAGE HIDING USING FLEXIBLE MACROBLOCK ORDERING (FMO)

One of the limitations of the quantization scale modulation solution of the previous section is related to themessage payload where only one message bit can be hidden per macroblock. This section introduces a second solution that benefits from a higher message bitrate through the use of FMO of the H.264/AVC video coding standard.In general, a coded picture is divided into one or more slices. Slices are self-contained and can be decoded and displayed independently of other slices. Hence, intraprediction of DCT coefficients and coding parameters of a macroblock is restricted to previous macroblocks within the same slice. This feature is important to suppress error propagation within a picture due to the nature of variable length coding. In regular encoding, when FMO is not used, slices contain a sequence of macroblocks in raster scan order. However, FMO allows the encoder to create what is known as slice groups. Each slice group contains one or more slices and macroblocks can be assigned in any order to these slices. The assignment of macroblocks to different groups is signaled by a syntax structure called the “slice group id”. This syntax structure is available in the picture parameter set header and therefore can be altered on picture basis. Notice that the H.264/AVC standard allows for a maximum of eight slice groups per picture.The idea behind the use of FMO in H.264/AVC is to spread the errors caused by burst packet losses to a larger portion of the picture. As such, error concealment becomes easier and more effective. There are a number of predefined slice group types in H.264/AVC that are designed for that purpose. Examples include interleaved slice groups, dispersed slice groups, foreground/background slice groups, box-out and wipe slice group. The H.264/AVC standard also allows for a sixth type for the explicit assignment of macroblocks to slice groups.

TABLE II

NUMBER OF SLICE GROUPS VERSUS NUMBER OF HIDDEN MESSAGE BITS PER MACROBLOCK

Number of slice groups	Potential message bits / MB	Message bits / MB
2	0,1	1
4	00,01,10,11	2
8	000,001,010,011, 100,101,110,111	3

Although FMO was devised for enhancing error resiliency and concealment, nonetheless it has been used for other purposes as well. For instance, the use of FMO to aid video scrambling for privacy protection. FMO has also been used to enhance the efficiency of video transcoding . In this paper, we make use of the explicit assignment of macroblocks to slice groups to hide messages in the video stream. Since macroblocks can be arbitrary assigned to slice groups, we propose to use the slice group ID of individual macroblocks as an indication of message bits. Assume for instance that two slice groups are used, the allocation of a macroblock to slice group 0 indicates a message bit of 0 and the allocation of macroblock to slice group 1 indicates a message bit of 1. Hence, one message bit per macroblock can be carried. Furthermore, since the H.264/AVC standard allows for a maximum of eight slice groups per picture then two or three message bits can be carried per macroblock as elaborated in Table II.

Clearly, one can think of other arrangements for assigning
message bits to macroblocks. The arrangement given in Table II is one straightforward example. Other examples might use eight slice groups yet use a subset of them for data hiding. For instance, one can use slice groups 2 and 5 to indicate a message bit of 0 and slice group 3 and 7 to indicate a message bit of 1. In general, what can be varied is the size of the subset of slice groups that are used to hide information, the message bit values hidden in these slices groups and the order in which message bits are assigned to slice groups. All of these permutations can even be altered per frame. Such scenarios indicate that the permutations for message hiding using this approach are very large. In this paper, however, we only consider the straightforward scenarios given in Table II.

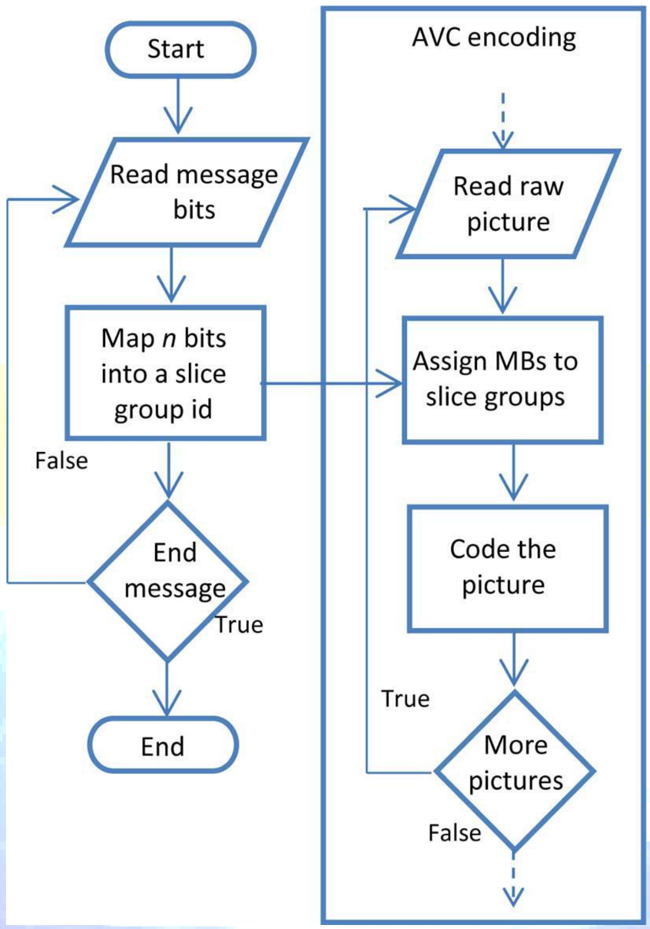

Fig. 3. Message hiding using FMO.

In general, to hide a message into the H.264/AVC bit stream, the message is first read into chunks of bits, where is either 1, 2, or 3 according to the values in Table II. If macroblocks are coded per picture, then message bits can be used to allocate the macroblocks to slice groups. The process of message hiding is illustrated in Fig. 3. To extract the message bits, each time a picture is decoded, the macroblock to slice group mapping syntax structure is used

to read message bits and append them to the extracted message. The process of message extracting is illustrated in

Fig. 4.

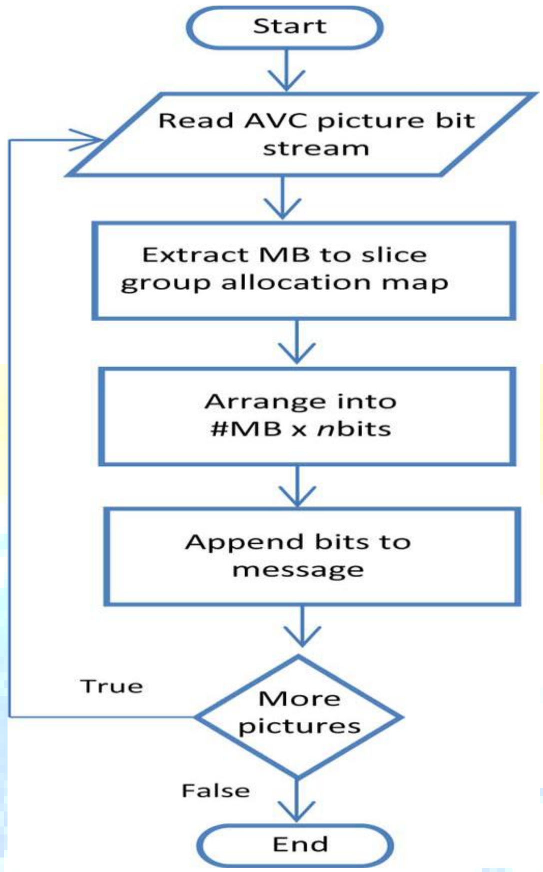

Fig. 4. Message extracting using FMO.

IV. RESULTS AND DISCUSSION

This section reports the experimental results of the proposed message hiding solutions and compares them to existing work reported. All the messages in Section V are generated randomly using a uniform distribution of ones and zeros.

A. Quantization Scale Modulation Experimental Results

We evaluate the quantization scale message hiding solution using the following criteria.

- Message prediction accuracy.
- Message hiding payload which can be measured in Kilobits per second (Kb/s).
- The excessive bitrate as a result of message hiding in Kb/s. This is computed as the difference between the bit rates of the video carrying the message and the original video.
- Lastly, the drop in PSNR measured in decibels.

B. FMO Experimental Results

Unlike the proposed quantization scale modulation solution, there is no message prediction in the FMO solution, hence we evaluate the FMO message hiding solution using the following criteria:

- message hiding payload which can bemeasured in Kilobits per second (Kb/s);
- excessive bitrate as a result of message hiding in Kb/s;
- lastly, the drop in PSNR measured in decibels. In the first set of experiments we use CBR coding at 1.5Mb/s and observe the drop in PSNR as a result of message hiding using FMO.

The results are reported for three cases. The first case uses two slice groups per frame hence the maximum bits to hide per macroblock is 1. The second case uses four slice groups per frame hence the maximum bits to hide per macro block is 2. Lastly, the third case uses eight slice groups per frame and therefore the maximum bits to hide per macro block is 3.

V. PROPOSED WORK

In the existing system Distortion is a major problem and also multi embedding is a problem. In order to overcome these problem we use a distortion less technique with the help of prediction based filter. In normal prediction based filter has less security problem and retrieval problem due to no change in prediction context . In order to overcome this we have embedding not only into the current pixel but also into its prediction context. In the previously discussed techniques the prediction based gives less distorted o/p and it support multi-embedding But it fail in retrieval Our technique will eradicate it by embedding not only into the current pixel but also into its prediction context. Transformation: Here prediction providing change to neighbors with less distortion in Transformation. In Reversible Transformation we retrieve data by removing the change in neighbors. It gives more security and easy retrieve But here we change neighbors so there may be some chance for distortion. Our technique will eradicate the distortion with proper change in neighbor. The function modules are, 2*2 odd and even Block Preparation, Prediction error for odd and even, Bit Adding,. Optimization, Re change Original.IT tells that the pixel and its context cover a 2x2 image block. If the image is partitioned in disjoint 2x2 blocks and the transform is applied on each block, the upper bound of the provided embedding capacity is 0.25bpp. In order to increase the embedding capacity, there are two common solutions such as multiple embedding or transforming over a denser partition. For each block a prediction error is calculated. The prediction error mean that it is a difference between the current pixel intensity value and the previous pixel intensity value. Since the predicted error value is high In order to overcome it an optimized prediction error value be determined The optimized value is d ,in order to provide high embedding capacity with low distortion and also for retrieval efficiency the optimized value for n,nw,w pixel also be calculated d=f(Pb) Pb=P+b (b-bit to be embedded). Now Replace the pixels with corresponding optimized value .

VI. CONCLUSION

This paper proposed two novel approaches to message hiding. In the first approach, the quantization scale of a CBR video is either incremented or decremented according to the underlying message bit. A second-order multivariate regression is used to associate macro block-level features with the hidden message bit. The decoder makes use of this regression model to predict the message bits. It was shown that high prediction accuracy can be achieved. However, the message payload is restricted to one bit per macro block. The second approach proposed in the paper works for both CBR and VBR coding and achieves a message payload of 3 bits per macro block. The FMO was used to allocate macro blocks to slice groups according to the content of the message. The low distortion transform based error prediction gives better data embedding in Video frames. This is done by using prediction based filter.It provide robustness against channel bit errors.

REFERENCES

[1] M. Carli, M. Farais, E. D. Gelasca, R. Tedesco, and A. Neri, "Quality assessment using data hiding on perceptually important areas," in Proc. IEEE Int. Conf. Image Processing, ICIP, Sep. 2005, pp. III-1200-3–III-1200-3.

[2] S. Kapotas and A. Skodras, "A new data hiding scheme for scene change detection in H.264 encoded video sequences," in Proc. IEEE Int. Conf. Multimedia Expo ICME, Jun. 2008, pp. 277–280.

[3] A. Yilmaz and A. Aydin, "Error detection and concealment for video transmission using information hiding," Signal Processing: Image Communication, vol. 23, no. 4, pp. 298–312, Apr. 2008.

[4] Y. Li, H.-X. Chen, and Y. Zhao, "A new method of data hiding based on H.264 encoded video sequences," in Proc. IEEE Int. Conf. Signal Processing, ICSP, Oct. 2010, pp. 1833–1836.

[5] K. Nakajima, K. Tanaka, T. Matsuoka, and Y. Nakajima, "Rewritable data embedding on MPEG coded data domain," in Proc. IEEE Int. Conf. Multimedia and Expo, ICME, Jul. 2005, pp. 682–685.

[6] D.-Y. Fang and L.-W. Chang, "Data hiding for digital video with phase of motion vector," in Proc. IEEE Int. Symp. Circuits Systems, ISCAS, Sep. 2006.

[7] C. Xu, X. Ping, and T. Zhang, "Steganography in compressed video stream," in Proc. Int. Conf. Innovative Computing, Information and Control, ICICIC'06, 2006, vol. II, pp. 803–806.

[8] H. A. Aly, "Data hiding in motion vectors of compressed video based on their associated prediction error," IEEE Trans. Inform. Forensics Security, vol. 6, no. 1, pp. 14–18, Mar. 2011.

[9] K. Wong, K. Tanaka, K. Takagi, and Y. Nakajima, "Complete video quality-preserving data hiding," IEEE Trans. Circuits Syst. Video Technol., vol. 19, no. 10, Oct. 2009.

[10] Y. Hu, C. Zhang, and Y. Su, "Information hiding based on intra prediction modes H.264/AVC," in Proc. IEEE Int. Conf. Multimedia and Expo, ICME, Jul. 2007, pp. 1231–1234.

Author's Detail:

Berdin N Prabhudhas D.S received the B.Tech degree in Information Technology from, C.S.I Institute of Technology, Thovalai in 2011. Presently pursuing M.E degree in Computer Science and Engineering in PSN College of Engineering and Technology, Tirunelveli.