An Analysis of Efficient Quad-Tree based on frame complexity of I-frame of video for H 2.64

Amit Kant Pandit
SMVDU, amit.pandit@smvdu.ac.in

Abstract- In this paper, diamond search method, on an optimized pre-processed quad tree, based on I-frame complexities of video is studied for controlled bit rate and PSNR. Here the edge detection technique is used to construct the quad tree of I-frame, which is used in predicting (reconstructing), P-frames. The approach has been applied to standard transform coding algorithm. Motion estimation and compensation techniques are utilized and the results obtained are tested for different combinations of blocks sizes obtained from quad tree for bit rate and PSNR. At the same time proposed approach test the frame skipping approach so as to minimize the total bit rate required for reconstruction of video. The quantization step is calculated as per the ISO/2003 standards. The simplified R-D theorem is used for rate-distortion calculation The Lagrange's parameter used is calculated from the quantization parameter itself thus saving in extra complexities at the transmitter side. Reconstructed frames are shown to provide best PSNR even at five frames per second rate.

Key words: - Edge detection, quad tree, diamond search, Variable length code

I. INTRODUCTION

A typical video compression system consists of motion estimation and compensation, DCT with quantization, lossless variable length coding with a feedback loop to minimize the bit rate and at the same time increase the PSNR. In general the compression system tries to reduce the total bit rate, increase PSNR, and is aimed at less buffer and computation requirement at receiver side.

Motion estimation and DCT exploit the spatial-temporal redundancies that exist in video signals to reduce the total bit requirement for encoding.

Motion estimation and DCT coding process consume maximum processing power of the transmitter; and the coded DCT coefficients consume the maximum bandwidth of the channel. To perform the motion estimation in real time, with limited processing ability is one of the major challenges of video compression. The motion of different objects of a video sequence is content dependent and the objects themselves are not only difficult to extract but also the process is highly compute intensive. So, the first approximation is based on the assumption of uniform motion of all the pixels in defined rectangular block .This simplification allows quicker motion estimation and easier application of the transform techniques. The second assumption is that the motion of these blocks is limited in the spatial domain limiting the search region, where a similar block is to be matched in the reference frame. The test block is matched to a block in the current frame according to some metric like the minimum absolute difference or any other correlation based measure. The consequential difference block is transform coded using DCT, quantized and then coded and transmitted along with the side information consisting of the motion of the block. The compute intensive parts of the operation are determination of the motion vector and DCT coding of the error blocks. This necessitates need for techniques that can yield not only a quick estimate of the motion vector but also steer clear of the application DCT as far as possible.

In real time, video application in the mobile communication, more issue crop up apart from the motion estimation and compensation. The bit rate requirements allow for very low bit rate and the variable bit rate from motion compensated video needs to be smoothed in addition to very low latencies in end to end communication.

So, in designing the system quick adjustment of the bit rate must be provided to cater to the drastic and unpredictable changes in the motion compensated coded video signal.

In mobile domain, algorithms that analyze the video sequence and then apply compression techniques cannot be employed due to time and processing limitations. Alternate ways must be applied to control the bit rate with only permissible loss of perceptual quality. Hence, any preprocessing that can minimize the blocks used for constructing the P frame will drastically diminish the processing requirements at the transmitter. At the same time the effective bit rate will also be reduced, which in turn reduces the bandwidth and buffer requirements.

II. PROBLEM DEFINITION

It is apparent that the problem of most efficient compression is the problem of minimum bit allocation to a frame at some distortion level. Thus, the problem reduces to an optimization problem involving the choice for each image region the most efficient coded representation requiring minimum bits for representing segmentation information (size of block), prediction mode (whether to predict using motion compensation or code the section independently), Motion vector, and coding displaced frame difference [9]. The problem can be stated as:

"What part of a video signal be coded using which method and what is the most appropriate choice of parameters for obtaining the most optimal solution".

Since it is multiple constraint problem. Universal solution is not possible to arrive at. So as a first simplification, the optimization is tried on a particular design. Since, all the existing standards viz. MPEG 1 / 2, H.261, H.263 and H.264 recommends Hybrid Coder, we take the case of Hybrid Coder. The design and operation of a hybrid coder involves:

(i) Segmentation of each picture frame into small region termed blocks
(ii) Selection as to send motion compensated block or new INTRA block.
(iii) Whether to send a full new INTRA picture or predicted (INTER) frame.
(iv) Selecting the best match and its coding.
(v) Coding of displaced frame difference.

Now the most important step is segmenting the picture Frame into logically related small region called blocks .the choice of block size and block decides the final compression and PSNR

The whole process is computationally heavy, in spite of the square blocks, reduced search regions etc. To reduce the computational load further, the DCT computation has to be reduced as it is the most compute intensive operation. The bit rate is a function of the number of quantized DCT, and controlling the bit rate amounts to controlling DCT coefficients.

The solution to minimize the DCT coefficient lies in the design of variable block size coders, the major difficulties are usually encountered in suitably subdividing a frame into blocks of homogenous characteristics and in coding, in an expensive way. The segmentation choice of block shape and size is highly content dependent and it also depends upon spatial correlation in the frame. This can be seen from decomposing the frame into blocks of different sizes and determining the overall bit rate required for coding the frame [1][2]. It is observed that subject to a distortion criterion, static area can be coded more efficiently using large blocks while changing area be coded with blocks of smaller size so that motion per block is low [3].

In the design of variable block-size coders, the hurdles are usually encountered in suitably sub dividing an image into blocks of consistent characteristics and then coding, in an inexpensive but efficient way, the information about the blocks' positions and sizes. A quad tree structure has been utilized in many applications as an efficient tool to overcome such difficulties. This structure makes it possible to easily and quickly subdivide an image into blocks, with the characteristics required, by simply performing a recursive splitting process.[4]

Hence, the overall requirement for video compression is to find an optimized block size which in turn, will reduces the computation & bit rate. At the same time the proper selection of block size will help in increasing the PSNR of the P- Frame (reconstructed frame).

We need a quad tree which is pre processed and efficient, take into account that the numbers of blocks are to be minimized, to minimize bitrates. The pre processing should be such that the quad tree generated should effectively trace the blocks to a minimum size where actual motion is there and the static blocks are not splitted. The technique used for splitting should reflect on the effect of noise also.

III. PROPOSED TECHNIQUE

Here, modified quad tree structure is studied that allows controlling the minimum bit rate and minimum distortion path. Here, for simulation purpose we have converted the RGB reference and Current frame (I-frame) into gray.

Now this Gray frames obtained act as seed for the optimized quad tree.

We apply the canny edge detection algorithm on the gray (intensity) frame obtained above. Edges are those places in an image that correspond to object boundaries. To find edges, in general an edge detector function looks for places in the image where the intensity changes rapidly, using one of these two criteria:

1] Places where the first derivative of the intensity is larger in magnitude than some threshold

2] Places where the second derivative of the intensity has a zero crossing.

The Edge function provides a number of derivative estimators, each of which implements one of the definitions above. The most powerful edge-detection method is the canny method. The Canny [5] method differs from the other edge-detection methods in that it uses two different thresholds (to detect strong and weak edges), and includes the weak edges in the output only if they are connected to strong edges. This method is therefore less likely than the others to be fooled by noise and more likely to detect true weak edges. This function returns a binary image containing 1's where edges are found and 0's elsewhere. The edge detected frame (binary image) is used, for splitting procedure, the image is processed in the Hilbert scan [6][7] order of block size 16*16.Here we apply the following processing to ,which decide when to break the block of (n*n) size into four blocks of (n/2*n/2)

Algorithm

Calculate;
Size of frame= I
Size of edges in frame = E
No calculate percentage of edges in frame that is complexity parameter P=(E/I)*100

If (P>= X)
[If frame is complex then blocks of bigger size to be broken into smaller ones with following condition
i) Split 16*16 blocks if
Number of edges in block > 10%
ii) Split 8*8 blocks if
Number of edges in block > 10%]

Else if (P>=Y)
[If frame is less complex then blocks of bigger size to be broken into smaller ones with following condition
i) Split 16*16 blocks if
Number of edges in block > 20%
ii) 8*8 blocks if Number of edges in block > 20%

Else if (P<=Z)
[If frame is less complex then blocks of bigger size to be broken into smaller ones with following condition
iii) Split 16*16 blocks if
Number of edges in block > 25%
iv) 8*8 blocks if Number of edges in block > 25%

End
Where X> Y> Z

This algorithm controls the splitting process. Here according to the complexity of the frame the decision to split the blocks of size (n*n) into four is taken. The decision again rest on satisfaction of new test conditions.

This process is recursive where a splitting condition, presence of number of edges, is satisfied according to the algorithm.

This makes it possible to generate large blocks in the homogeneous areas and smaller ones where large complexities are there .The process continues to split till block size of 4*4 is obtained. The block size less than 4*4 is not used as it sharply decreases the compression capability of the quad tree techniques which is base on correlation among pixels. This helps in getting to exact blocks where actual movements are present; it excludes the motion blocks from static information blocks. The effect of noise is minimized by canny itself, as it smoothen any sharp inconsistency in vicinity of a pixel. The quad tree generated will have required blocks which have significant motion only

The resultant quad tree structure is used for block matching for searching the motion of reference frame blocks in the current frame, by using efficient directional diamond search [8] method.

Through this we have, calculated the DFD and DVF of all the blocks formed by the quad tree.. The resultant blocks are uniformly quantized and encoded by VLC in 4*4 blocks.

So as to have optimum path for constructing the current frame, restraint to bit rate and picture quality is used, The Sum of DVF and DFD are minimized as per the RD theorem. The cost is calculated as follows

Cost=SAD of blocks +? (bits used in coding the blocks including motion and residual)

Where '**?**' is Lagrange multiplier which controls which path will be chosen, among the possible allocation in the R-D plane here

? =0.85*2$^{(QP-12/6)}$ is used

Where "QP" stands for quantization parameter used.here for analysis, we are using QP =32 fixed quantization parameter.

The cost is used to determine whether the father or its four Childs are to be taken into account. Here the minimum cost combination is chosen between the blocks of different sizes.

This approach helps in overall minimization of blocks to be processed and total number of bits required for transmission

IV. RESULTS AND CONCLUSION

For testing following were assumed X=20; Y=10; Z=10. The results are tested for frames *of* - **Foreman frame size 176*144 QCIF Format (frame-1 ,6) , and Mother Daughter frame of size 352*288CIF Format (frame-2-7)**

To evaluate the performance, we have implemented our approach on MATLAB 7.5.

The results are compared for different combinations of blocks. The permissible block size is 16*16/16*16-8*8/16*16-8*8-4*4. The results show that there is decrease in total bit rates and increased PSNR values. The frames are reconstructed only by using inter mode of prediction at a frame rate of 5frame/sec.

Even though, the above mentioned technique involves an increase in number of blocks used for reconstruction and bits used for encoding the motion vectors. at the same time the bits used to encode residual energy is decreased .the overall effect is that there is a net decrease in the total bit rate required and better PSNR values are obtained .table 1,2 shows the results obtained The results show that by using the proposed approach the total bit rate is decreased and better PSNR is obtained this shows that by calculating the complexity parameter we can control the splitting process ,which results in saving the total bits and increase or same PSNR values.

The Approach can be extended for better results. By using better coding for motion vectors we can minimize the motion bit, and total bits. Further the complexity parameter of individual frames is used in deciding the splitting hence the decision of block size is computed fast.

Table1

Foreman frame size 176*144 QCIF Format

(results of proposed approach)

Sequence	Combination of blocks	No. Of Blocks Used	Bits used to encode residual energy	Bits used in encoding motion vectors	Bit Rate.	PSNR (db)
Foreman	16x16	99	22361	1099	23460	29.7914
	16x16 8x8	62 148	20475	2229	22704	30.1326
	16x16 8x8 4x4	59 135 100	19653	2915	22568	30.2391

Table2

Mother Daughter size 352*288 CIF Format

Sequence	Combination of blocks	No. Of Blocks Used	Bits used to encode residual energy	Bits used in encoding motion vectors	Bit Rate.	PSNR (db)
Mother daughter	16x16	396	30631	2346	32977	34.1980
	16x16 8x8	308 352	27910	3739	31649	34.3521
	16x16 8x8 4x4	279 361 428	25157	5862	31019	34.1972

REFERENCES

[1] G.J.Sullivan and B.L.Baker, "Efficient quadtree coding of images and video," *IEEE Trans. on Image Processing*, vol.3, pp.327- 331, May 1994

[2] J.Ribas-Corbera and D.L.Neuhoff, "Optimal bit allocation for lossless video coders: Motion vector vs difference frame," *Proc. of ICIP*, vol.3, pp. 180-183, 1995

[3] A.Ortega, K.Ramchandran, and M.Vetterli, "Optimal trellis-based buffered compression and fast approximation," *IEEE Trans. on image Processing*, vol.3, no.1, pp. 26-40, Jan.1994.

[4] Francesco G.B. De Natale, Giuseppe S. Desoli, Stefan0 Fioravanti, and Daniele D. Giusto "AN EDGE- BASED SPLITTING CRITERION FOR ADAPTIVE TRANSFORM CODING" , 1993 IEEE

[5] [Canny861] J. F. Canny, "A Computational Approach to Edge Detection", IEEE Trans. Patt. **Anal.** Mach. Intell., Vol. PAMI-8, pp. 679-698, NOV. 1986.

[6] W.Enkelmann, "Investigation of multigrid algorithm for the estimation of optical flow fields in image sequences," *Computer graphics and Image Processing*, vol.43, pp. 150 – 177, Aug.1988.

[7] M.Elad, P.Teo and Y.Hel-Or, "Optical filters for gradient based motion estimation," *IEEE Int'l conf. on Computer Vision*, 1999.

[8] Hongjun Jia and Li Zhang ," Directional diamond search pattern for fast block motion estimation *"ELECTRONICS LETTERS 30th October 2003 Vol. 39 No. 22*

[9] B.Girod, "Rate-constrained motion estimation," *Proc. of SPIE conf. on Visual communication and Image Processing*, Chicago, USA, pp. 1026 – 1034, Sept.1994.