Fast Mode Decision Algorithm for H.264/AVC using Edge Characteristics of Residue Images

Amrita Ganguly*
Department of Electronics and Communication Engineering
Indian Institute of Technology Guwahati
Guwahati-781039, Assam, India
a.ganguly@iitg.ernet.in

Anil Mahanta
Department of Electronics and Communication Engineering
Indian Institute of Technology Guwahati
Guwahati-781039, Assam, India
anilm@iitg.ernet.in

ABSTRACT

Video compression algorithms require decision making regarding the selection of different coding modes. Of all the encoding elements in H.264, inter prediction is computationally most intensive and thus adds to the computational burden for the encoder. In this paper, we propose a fast inter prediction algorithm for JVT video coding standard H.264/AVC. The fast prediction algorithm is based on the edge information obtained by calculating the edge histogram parameters of the frame difference residual image. The inclusion of different modes for motion estimation is based on the direction information obtained from the edge histogram parameters. The proposed method reduces the computational complexity and time considerably by upto 63% compared to the JVT benchmark JM12.4 while maintaining similar PSNR and bit rate. Experimental results for various test sequences at different resolutions are presented to show the effectiveness of the proposed method.

Keywords

H.264/AVC, inter mode decision, edge histogram, video coding

1. INTRODUCTION

The H.264/AVC [1] is the video coding standard developed by Joint Video Group (JVT) of ITU-T Video Coding Experts Group (VCEG) and ISO/IEC MPEG Video Group. It has been approved as one of the most efficient coding solution. It builds on the concepts of earlier standards such as MPEG-2 and MPEG-4 Visual and offers better compression efficiency and greater flexibility in compressing, transmitting and storing video. The H.264/AVC video coding standard improves the coding performance significantly by adopting many advanced techniques like entropy encoding, small block size computation and in-built deblocking filter that contribute to the increase in complexity.

H.264 process the video image in blocks [15]. Irrespective of the video resolution, the number of blocks have an effect on the computational requirements. H.264 permits the use of different block sizes, 16×16 pixels called a macroblock (MB) down to 4×4 pixels. Fine details in a MB are defined using smaller blocks whereas coarse and homogenous regions by larger blocks. For each MB, the H.264 forms a prediction based on previously coded data, either from the current frame (intra prediction) or from other frames that have already been coded and transmitted (inter prediction). The rate-distortion optimization (RDO) procedure is utilized in the prediction process for the purpose of selecting the optimal mode from all candidate modes.

Several approaches have been proposed to reduce the complexity and time for the inter mode decision [4, 5, 6, 7, 8, 9, 11, 13, 16, 17]. In [5] adaptive MB selection is used for mode decision. In [7] the fast motion estimation is based on the successive elimination algorithm (SEA) using sum norms to find the best estimate of the motion vectors and to implement efficient calculations for variable blocks. In [3], edge histogram based fast intra mode decision has been proposed. In [16] edge histogram parameters have been used for fast inter mode decision. In this paper, we propose a edge histogram based fast inter mode decision process where we use the edge histogram parameters to first identify whether a MB is homogenous and can be encoded using larger block size. Edge detection is then done for the frame difference residues and based upon the directions of edge, mode decision is taken.

The paper is organized as follows. Section 2 gives an overview of the inter prediction modes and certain observations that have been made based on a detailed study of the inter prediction modes. Section 3 presents the proposed fast encoding algorithm based on the observations made in Section 2. The results of the proposed scheme as compared to JM12.4 are given in Section 4. The comparisons are made in terms of coding efficiency, reduction in computational complexity and time. The paper ends with conclusions.

2. ANALYSIS OF FULL MOTION ESTIMATION IN H.264

2.1 Inter Mode Decision in H.264/AVC

The H.264/AVC standard supports both intra and inter

*Corresponding author

ICVGIP '10, December 12-15, 2010, Chennai, India

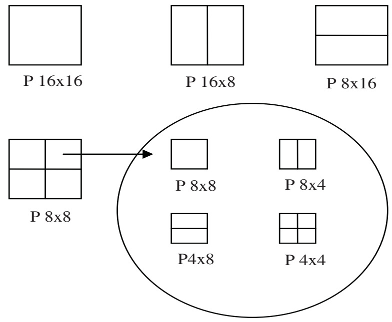


Figure 1: Inter prediction block sizes for a MB

prediction processes. In the inter prediction process, there are seven block sizes P_{16x16}, P_{16x8}, P_{8x16}, P_{8x8}, P_{8x4}, P_{4x8} and P_{4x4} that are used by H.264 besides the SKIP and the INTRA modes [10], [15]. For each MB, all the modes are tried and one which gives the least RD cost is selected for encoding. However, in real video sequences the distribution of modes is not uniform among the MBs [10]. It depends upon the characteristics of the video sequences. Regions with homogenous motion, smooth motion of moving background and static background use larger block size motion compensation. For regions with high detail and complex motion, smaller block sizes to represent motion gives better coding efficiency.

The temporal redundancies between frames are exploited in inter mode decision. Fig.1 shows the partitioning of a MB in seven modes. The SKIP mode directly copies the contents of the MB of the reference frame at the same position in the reconstructed current frame without performing motion compensation. INTRA mode uses the spatial redundancies in the pixels of the MB without performing motion estimation (ME). In the full motion estimation (FME) the encoder tries all the modes and chooses the best mode in terms of the least RD cost (J_{MODE}) by employing the Lagrangian RDO process [14]. The Lagrange mode decision for a macroblock S_k proceeds by minimizing the J_{MODE} defined as

$$J_{MODE}(S_k, MODE|QP, \lambda_{MODE}) \\ = SSD(S_k, MODE|QP) + \lambda_{MODE}R(S_k, MODE|QP), \quad (1)$$

where QP indicates the MB quantization parameter and λ_{MODE} is the Lagrangian multiplier. $R(\cdot)$ represents the number of bits associated with the chosen mode, residual data, MVs, MB header and QP. Macroblock mode, MODE, is varied over the sets of possible MB modes. $SSD(\cdot)$ is the sum of the squared differences between the original and the reconstructed pixels. The computation of the RD cost requires the availability of the reconstructed image and the actual bit count. This necessitates that the encoding and decoding processes are completed for every mode. Thus the computational requirements for the mode selection process is very high.

2.2 Mode Distribution in H.264/AVC

Neighboring video frames have large similarities between them. ME and compensation attempts to reduce the temporal redundancy by exploiting these similarities. It is observed that natural videos have a lot of homogenous regions with similar spatial property which usually get encoded with larger block sizes. Regions having little or no motion that is region which are considered stationary get encoded in the SKIP mode. Regions having similar motion also get encoded with larger block sizes. Areas with large and complex motion and having boundaries of moving objects get encoded with smaller block sizes. In order to study the distribution of modes, FME was performed on sequences having different motion complexity. Table 1 shows video sequences classified into three classes: Class A having low and simple motion, Class B having medium to high motion and Class C having high motion complexity.

Table 1: Different Classes of Sequences

Type	CIF Sequence	QCIF Sequence
Class A	News, Mother and daughter(MaD) Container, Hallmonitor	Suzie, Claire Missamerica
ClassB	Foreman, Coastguard Harbour, Ice	Foreman, Silent Crew
Class C	Mobile, Flower Tempete, Stefan	Mobile, Football Soccer

Table 2: Distribution of MBs (%) in different Modes at QP=32

Sequence		SKIP	P_{16x16}	P_{16x8}	P_{8x16}	P_{8x8}	INTRA
CIF Class A	News	82.57	6.55	2.25	1.85	6.68	0.00
	Container	83.46	15.42	0.21	0.23	0.67	0.01
	Hallmon	64.76	23.30	2.92	3.25	5.57	0.20
Class B	Coastguard	20.58	44.76	6.70	8.35	19.38	0.23
	Foreman	29.02	34.41	8.90	7.97	18.50	1.20
	Ice	64.76	11.20	4.82	3.92	14.85	0.45
Class C	Flower	24.92	51.82	3.06	2.70	17.47	0.03
	Mobile	12.04	40.54	8.62	9.82	28.98	0.42
	Stefan	22.69	39.67	4.71	5.86	26.04	1.03
QCIF Class A	Claire	91.17	5.95	0.49	0.66	1.73	0.00
	Suzie	60.63	23.12	7.41	5.03	3.54	0.27
	MissAmerica	64.89	28.46	2.09	2.01	2.49	0.06
Class B	Foreman	34.53	37.88	6.54	7.34	12.76	0.95
	Silent	75.70	8.10	1.93	1.32	12.88	0.07
	Crew	16.92	25.37	12.63	5.97	38.01	1.10
Class C	Football	15.20	20.48	6.26	11.45	28.35	18.26
	Mobile	11.57	42.22	10.85	12.14	22.87	0.35
	Soccer	18.90	29.00	12.63	15.97	22.40	1.10

Table 2 shows the distribution of MBs (in %) encoded in different modes for a QP value of 32. The P_{8x8} mode includes the P_{8x8}, P_{8x4}, P_{4x8} and P_{4x4}. The results clearly show that Class A sequences have most of the MBs encoded with larger block sizes, Class B sequences have MBs encoded with both large and small block sizes whereas for Class C sequences, smaller block sizes are more prevalent. From the observations made above it may be concluded that for all types of sequences there are many MBs encoded in the SKIP mode and P_{16x16} mode. If such MBs can be detected at an early stage, it will result in large savings in computation. Thus, if the stationary and homogenous regions in a frame can be determined prior to ME early decision on the encoding mode can be taken.

2.3 Directional Partitioning in H.264/AVC

From Fig.1 it is observed that the the partitioning of the MB is either horizontal or vertical. P_{16x8} and P_{8x4} are vertical partitions whereas P_{8x16} and P_{4x8} are horizontal partitions. Hence, if the direction of the moving boundaries in a video can be determined early, then only directional modes can be chosen for encoding.

3. PROPOSED FAST MODE DECISION ALGORITHM

The proposed fast mode decision algorithm is based on the observations made in the previous section. Here, each MB is first examined for stationarity. If the MB is stationary, it is encoded in the SKIP mode. Otherwise, the MB is further examined for homogeneity. If it is homogeneous, the MB is encoded using larger block sizes. If both these conditions are not satisfied, then directional prediction is done for the MB. Each of the decision process is described here.

3.1 Determination of Stationarity

Stationarity refers to the stillness between consecutive frames in the temporal direction. Regions having similar motion in consecutive frames are also considered stationary. The MBs which are temporally stationary will usually get encoded in the SKIP or in the P_{16x16} mode. The simplest method of temporal prediction is to use the previous frame as the prediction for the current frame. Hence to form a prediction, the frame difference residue (R_{DF}) is obtained by subtracting the previous frame (MB_P) from the current frame (MB_C) and the sum of difference absolute values S_{DIFF} is obtained as

$$R_{DF}(i,j) = MB_C(i,j) - MB_P(i,j), \quad i,j = 1,\ldots,16 \quad (2)$$

$$S_{DIFF} = \sum_{i=1,j=1}^{16,16} abs(R_{DF}) \quad (3)$$

It is observed that if the MB is stationary, then the frame difference residues in R_{DF} have very low values resulting in a small value of S_{DIFF}. Thus if S_{DIFF} is below a threshold T_S, the MB is classified as stationary and will be encoded in the SKIP mode or in P_{16x16} mode. Here the threshold T_S is taken as 200 as per [16]. If the MB satisfies this condition then further it is determined whether the MB qualifies for the SKIP mode. If all the residues in R_{DF} are very low, then the MB can be considered temporally stationary. Hence the residues in R_{DF} are examined and if all the residues in R_{DF} have absolute values less than or equal to 1, then the MB qualifies for the SKIP mode and further ME is terminated.

The MBs that do not satisfy this condition but still have S_{DIFF} below T_S is encoded in the P_{16x16} mode. MBs that do not qualify for the stationarity condition are further tested for homogeneity.

3.2 Determination of Homogeneous Region

Homogeneous regions have similar spatial properties. There exist many techniques for detecting homogenous regions. In this paper the homogenous region detection is based on the edge information as video object have strong edges. The edge based homogeneous region detection has been proposed in [16] and is described here.

The edge map is created for each frame using the Sobel operator. Each pixel in the block will be associated with an edge vector containing edge direction and amplitude. We define edge vector $D_{i,j} = \{dx_{i,j}, dy_{i,j}\}$ where

$$dx_{i,j} = p_{i-1,j+1} + 2\times p_{i,j+1} + p_{i+1,j+1} - p_{i-1,j-1} - 2\times p_{i,j-1} - p_{i+1,j-1} \quad (4)$$

$$dy_{i,j} = p_{i+1,j-1} + 2\times p_{i+1,j} + p_{i+1,j+1} - p_{i-1,j-1} - 2\times p_{i-1,j} - p_{i-1,j+1} \quad (5)$$

Here $dx_{i,j}$ and $dy_{i,j}$ are the degrees of differences in vertical and horizontal directions. The amplitude of the edge vector is computed by

$$Amp(D_{i,j}) = |dx_{i,j}| + |dy_{i,j}| \quad (6)$$

$$S_{Amp}(N) = \sum_{i=1,j=1}^{N,N} Amp(i,j) \quad (7)$$

Homogeneity of a block of size $N \times N$ (where N is 16 or 8) is determined by using the amplitudes as per (7). If the sum of the amplitude S_{Amp} is below a certain threshold (T_{th}) it is designated as a homogenous block otherwise it is nonhomogeneous. The decision is taken as follows:

$$Decision = \begin{cases} 1, & if \quad S_{Amp} < T_{th} \\ 0, & otherwise \end{cases}$$

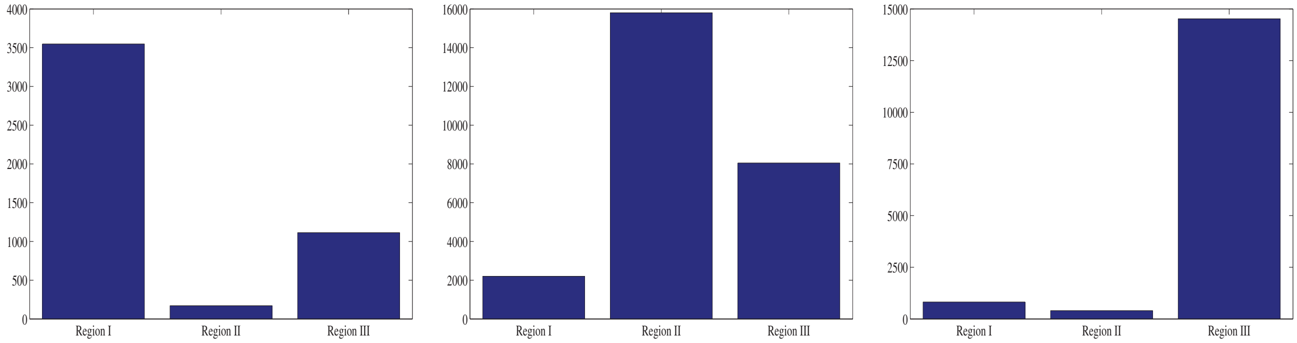
The homogeneity is first tested for N=16 that is for 16×16 block which is a MB. The T_{th16} for N=16 is taken as 20000 as given in [16]. If $S_{Amp}(16)$ is less than T_{th16} then the MB is homogenous and ME will be performed with larger block sizes namely P_{16x16}, P_{16x8} and P_{8x16}. Otherwise, each MB is divided into four 8×8 subblocks (N=8) and each subblock is tested for homogeneity. The T_{th8} for each subblock is taken as 5000. If $S_{Amp}(8)$ is less than the T_{th8} then the subblock is homogenous and ME is performed with P_{8x8} mode. Otherwise for each subblock, direction based ME is performed and is described next.

3.3 Directional Motion Estimation

From Fig.1 it is seen that the partitioning of the MBs into different sizes are either horizontal or vertical. Hence if the direction of the motion can be determined early, only those modes pertaining to the motion direction can be utilized for ME. Since the computational complexity for smaller block size ME is more, hence the directional approach is applied to those subblocks which are nonhomogeneous. The frame difference residue R_{DF} of the collocated MB in the difference frame will contain the information of the changes between two consecutive frames. In the proposed algorithm, R_{DF} is divided into four 8x8 subblocks and edge detection is done on each subblock. The direction of the edge is determined for each subblock. The direction of the edge is determined by the hyper-function

$$Ang(D_{i,j}) = \frac{180^\circ}{\pi} \times \arctan(\frac{dy_{i,j}}{dx_{i,j}}) \quad (8)$$
$$|Ang(D_{i,j})| < 90^\circ$$

The direction which gives the maximum histogram is taken as the direction for the motion estimation. To determine the direction, we divide the search range into three regions namely horizontal, vertical and plane region as shown in Fig.3. The region selected will depend upon the maximum histogram. Region I pertain to horizontal edge, Region II to vertical edge and Region III pertain to diagonal edges respectively. The modes selected for different regions are as



(a) Edge pertaining to Region I (b) Edge pertaining to Region II (c) Edge pertaining to Region III

Figure 2: Typical edge histograms for the frame difference image

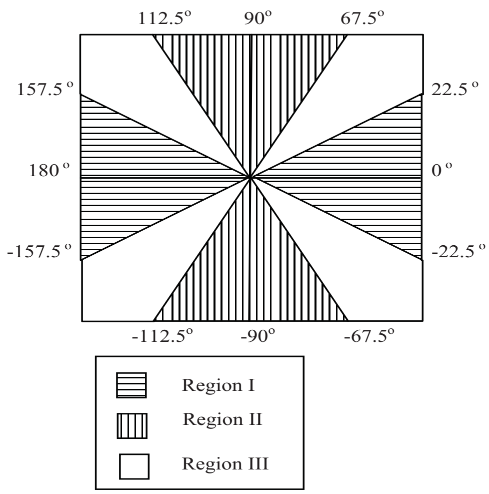


Figure 3: Partitioning into different regions

Table 3: Mode Selection for Different Regions

Region	Modes Selected
I	P_{8x8}, P_{4x8}
II	P_{8x8}, P_{8x4}
III	P_{8x8}, P_{8x4}
	P_{4x8}, P_{4x4}

given in Table 3. A typical histogram generated for different regions for the difference frame is given in Fig. 2.

3.4 Overall Algorithm

In the previous subsections, the principle idea of determination of homogeneous region and selecting a subset of all the modes for ME is illustrated. For clarity, the overall procedure for the proposed method for fast inter mode decision is summarized below:

1. For each MB in the current frame, subtract the collocated MB in the previous frame to get the residual difference R_{DF} and calculate S_{DIFF}.
2. If $S_{DIFF} < T_S$ and all residues in R_{DF} is ≤ 1, encode MB in the SKIP mode. Go to step 10 . Else
3. If $S_{DIFF} < T_S$ but have some residues in $R_{DF} > 1$, encode the MB in P_{16x16} mode. Go to step 10. Else
4. If the MB is non stationary, perform edge detection and generate the edge histogram.
5. If $S_{Amp}(16) < T_{th16}$ then MB is homogenous. Perform ME with P_{16x16}, P_{16x8} and P_{8x16} modes. Select the mode that minimize the RDO cost. Go to step 10. Else
6. Divide the MB into four 8×8 subblocks.
7. For each 8×8 subblock, if $S_{Amp}(8) < T_{th8}$ then subblock is homogenous. Perform ME with P_{8x8} mode.
8. Otherwise, perform edge detection on the collocated subblock in R_{DF} and find the maximum histogram M1.
   (a) If M1 corresponds to Region I, perform ME with P_{8x8} and P_{4x8} modes. Select the mode that minimize the RDO cost.
   (b) If M1 corresponds to Region II, perform ME with P_{8x8} and P_{8x4} modes. Select the mode that minimize the RDO cost.
   (c) If M1 corresponds to Region III, perform ME with P_{8x8}, P_{8x4}, P_{4x8} and P_{4x4}modes. Select the mode that minimize the RDO cost.
9. Repeat steps 7 and 8 for all subblocks.
10. Select the best mode and proceed with the next MB.

4. EXPERIMENTAL RESULTS

This section compares the results of the proposed algorithm with the previously reported Wu. *et al.*'s algorithm [16]. Results are presented as improvements over the standard H.264/AVC benchmark JM12.4. The fast intra prediction from [3] is included in the intra prediction process. The experiments were carried out using some common video sequences of different classes at the CIF and QCIF resolution as listed in Table 1. The configuration used is the baseline profile, motion search range of ± 16, sequence type IPPP and one reference frame. First 100 frames of each sequence (90 frames for Stefan) are processed and only the first frame is intra coded. QP used are 24, 28, 32 and 36 as per the recommended simulation conditions in [12]. Comparisons are

Table 4: Performance Comparison For Different Sequences

Class	Sequence	Performance Comparison					
		Proposed			Wu *et al.*'s [16]		
		ΔPSNR	ΔRate (%)	ΔT (%)	ΔPSNR	ΔRate (%)	ΔT (%)
CIF Class A	News	-0.10	-0.08	76.00	0.02	2.01	39.23
	MaD	0.02	2.46	73.98	0.01	0.85	43.21
	Container	0.04	1.06	63.32	0.05	1.55	46.18
	Hall	0.05	2.98	63.90	0.06	0.90	34.67
Class B	Foreman	0.05	2.92	64.96	0.07	1.22	34.90
	Coastguard	-0.01	1.78	43.48	0.05	0.54	26.30
	Ice	0.03	2.67	77.11	0.07	1.29	45.68
	Harbour	0.01	0.81	30.36	0.05	1.10	21.56
Class C	Flower	0.02	1.62	52.41	0.05	2.98	36.85
	Stefan	0.09	2.26	40.72	0.02	1.46	32.25
	Tempete	-0.01	0.92	42.68	0.09	1.02	27.21
	Mobile	-0.04	-1.30	38.62	0.09	1.69	12.35
	Average	**0.01**	**1.50**	**55.62**	**0.05**	**1.38**	**33.36**
QCIF Class A	Claire	-0.08	-1.01	83.82	-0.01	-0.95	47.35
	MissAmerica	-0.01	0.56	79.67	0.02	0.91	48.23
	Suzie	0.05	0.50	75.05	0.05	0.49	42.91
Class B	Foreman	0.04	1.91	52.24	0.04	1.29	30.25
	Silent	0.09	0.92	73.12	0.09	0.79	42.62
	Crew	0.09	1.04	62.90	0.05	1.65	19.64
Class C	Football	0.04	1.05	53.18	0.05	1.86	32.42
	Mobile	0.07	1.04	28.08	0.07	1.50	15.32
	Soccer	-0.04	-0.90	61.15	0.03	3.03	20.19
	Average	**0.02**	**0.57**	**63.24**	**0.04**	**1.17**	**33.21**

made in terms of distortion and percentage differences in rate and time taken for encoding. To evaluate the average encoding performance over a range of QPs, the differences in PSNR (ΔPSNR) in dB and bitrate (ΔRate (%)) are calculated according to numerical averages between RD curves as given by Bjontegaard [2]. The parameters ΔT, ΔPSNR and ΔRate are defined as,

$$\Delta\mathrm{T} = \left(1 - \frac{\mathrm{T_{fast}}}{\mathrm{T_{JM}}}\right) \times 100\%$$
$$\Delta\mathrm{PSNR} = (\mathrm{PSNR_{JM}} - \mathrm{PSNR_{fast}})$$
$$\Delta\mathrm{Rate} = \left(\frac{\mathrm{Rate_{fast}} - \mathrm{Rate_{JM}}}{\mathrm{Rate_{JM}}}\right) \times 100\% \quad (9)$$

Positive/negative value of ΔRate indicate increase/decrease in bitrate as compared to the standard JM software. Positive/negative value of ΔPSNR indicate decrease/increase in the quality as compared to the standard JM software. Positive values of ΔT indicate the time saving or speedup.

4.1 *Distortion and Compression Ratio comparisons*

Table 4 lists the performance of the proposed algorithm in comparison to JM12.4 implementation and Wu *et al.*'s algorithm [16]. The results are arranged for different classes of sequences. The trend in the results shows that for some of the sequences there has been marginal improvement in the PSNR performance. There is an average 0.01 dB loss in PSNR in the proposed method. It is interesting to note that the bitrate for some of the sequences has actually decreased marginally giving a better bitrate performance. However the average bitrate increase is 1.5%. Fig.4 shows the RD curve for QCIF sequence Foreman and CIF sequence Mobile.

4.2 *Computational Speedup*

Table 4 shows the percentage reduction in encoding time ΔT(%) for sequences of different classes. The time saving obtained depends upon the type of sequence. An increased

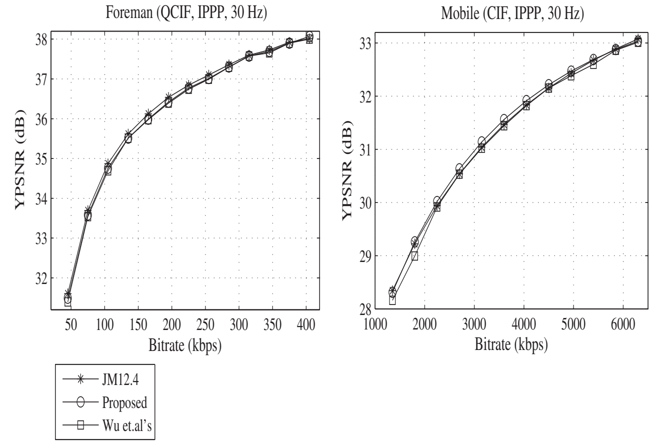


Figure 4: RD Curve for Foreman(QCIF) and Mobile (CIF) sequence

saving (83%) is noted for Class A sequences whereas time saving obtained for Class C sequences is comparatively low. This is due to the fact that Class A sequences have low motion complexity and hence a large number of MBs get encoded with larger block sizes. However, for all sequences, the proposed algorithm exhibits a good computational saving regardless of the QP setting. Class A and B sequences benefit more from the SKIP mode detection by detecting the stationary regions whereas Class C sequences benefit more from further examination of the directional features for smaller subblocks.

5. CONCLUSIONS

In this paper, an improved mode decision algorithm based on the stationarity of the MB and the edge histogram statistics of the residual difference frames for H.264/AVC video coding standard has been proposed. The temporal redundancies between frames of the video is exploited in the proposed scheme. The proposed fast inter prediction is based on observations made from extensive experiments performed on different test sequences. Results of simulations carried out on different sequences demonstrate that there is very little degradation of the PSNR and the bitrate performance of the proposed algorithm despite a large saving in encoding time and computation. The average encoding time saving is around 74% for Class A, 58% for Class B and 45% for Class C sequences. The proposed method achieves almost the same coding performance in terms of picture quality and compression ratio as that of the H.264/AVC standard and improves on Wu *et al.*'s [16] algorithm. Hence, for a variety of sequences with varying motion activities, the proposed algorithm gives a consistent performance on encoding time reduction, computational saving and coding efficiency.

6. REFERENCES

[1] Joint Video Team of ITU-T and ISO/IEC JTC 1, Draft ITU-T Recommendation and Final Draft International Standard of Joint Video Specification (ITU-T Rec. H.264 | ISO/IEC 14496-10 AVC), document JVT-G050r1, 2003.

[2] G. Bjontegaard. Calculation of average PSNR difference between RD-curves. *ITU-T VCEG, Doc. VCEG-M33*, pages 1–2, March 2001.

[3] A. Ganguly and A. Mahanta. Fast Mode Decision Algorithm for H.264/AVC Intra Prediction. In *TENCON*, 2009.

[4] X. Jing and L. P. Chau. Fast approach for H.264 inter mode decision. *Electronic Letters*, 40(17):1050–1052, 2004.

[5] D. Kim, J. Kim, and J. Jeong. Adaptive Macroblock Mode Selection for Reducing the Encoder Complexity in H.264. In *ACVIS*, number 4179 in LNCS, pages 396 – 405, 2006.

[6] G. Y. Kim, Y. H. Moon, and J. H. Kim. An Early Detection of All-Zero DCT Block in H.264. In *Int. Conf. Image Process (ICIP)*, volume 1, pages 453–456, October 2004.

[7] Y. L. Lee, Y. K. Lee, and H. Park. A fast motion vector search algorithm for variable blocks. In *ACVIS*, number 4179 in LNCS, page 311 Ű 322, 2006.

[8] Z. Liu, L.Shen, and Z. Zhang. An Efficient Intermode Decision Algorithm Based on Motion Homogeneity for H.264/AVC. *IEEE Trans. on Circuits and System for Video Technology*, 19(1):128–132, January 2009.

[9] I. Park and D. W. Capson. Improved Inter Mode Decision Based on Residue in H.264/AVC. In *International Conference on Multimedia & Expo*, pages 709–712, 2008.

[10] I. E. G. Richardson. H.264 and MPEG-4 Video Compression: Video Coding for Next-generation Multimedia. *John Wiley and Sons, 2003*.

[11] L. Shen, Z. Liu, Z. Zhang, and X. Shi. Fast Inter Mode Decisions Using Spatial Property of Motion Field. *IEEE Trans. on Multimedia*, 10(6):1208–1214, October 2008.

[12] G. Sullivan and G. Bjontegaard. Recommended Simulation Common Conditions for H.26L Coding Efficiency Experiments on Low-Resolution Progressive-Scan Source Material. *ITU-T VCEG, Doc. VCEG-N81*, pages 1–3, 2001.

[13] H. Wang, S. Kwong, and C.-W. Kok. An Efficient Mode Decision Algorithm for H.264/AVC Encoding Optimization. *IEEE Trans. on Multimedia*, 9(4):882–888, June 2007.

[14] T. Wiegand, H. Schwarz, A. Joch, F. Kossentini, and G. J. Sullivan. Rate-Constrained Coder Control and Comparison of Video Coding Standards. *IEEE Trans. on Circuits and System for Video Technology*, 13(7):688–703, July 2003.

[15] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra. Overview of the H.264/AVC Video Coding Standard. *IEEE Trans. on Circuits and System for Video Technology*, 13(7):560–576, July 2003.

[16] D. Wu, F. Pan, K. P. Lim, S. Wu, Z. G. Li, X. Lin, S. Rahardja, and C. C. Ko. Fast Intermode Decision in H.264/AVC Video Coding. *IEEE Trans. on Circuits and System for Video Technology*, 15(6):953–958, July 2005.

[17] H. Zeng, C. Cai, and K.-K. Ma. Fast Mode Decision for H.264/AVC Based on Macroblock Motion Activity. *IEEE Trans. on Circuits and System for Video Technology*, 19(1):1–11, April 2009.