Bit Rate Estimation for Cost Function of H.264/AVC

Mohammed Golam Sarwer[1,2], Lai Man Po[1] and Q. M. Jonathan Wu[2]
[1]*City University of Hong Kong, Hong Kong SAR, China*
[2]*University of Windsor, Canada*

1. Introduction

The appearance and development of various new multimedia services have need for higher coding efficiency. The ITU-T/ISO/IEC Joint Video Team established the newest video coding standard known as H.264/AVC (Joint Video Team (JVT), 2002). H.264/AVC offers a significant performance improvement over previous video coding standards such as H.263++ and MPEG-4 part 2 (Erol et al., 1998; Topiwala et al., 2001). New and advanced techniques are introduced in this new standard, such as intra prediction for I-frame encoding, multi-frames inter prediction, small block-size transform coding, context-adaptive arithmetic entropy coding, de-blocking filtering, etc. These advanced techniques make this new standard provides approximately 50% bit rate saving for equivalent perceptual quality relative to the performance of prior standards.

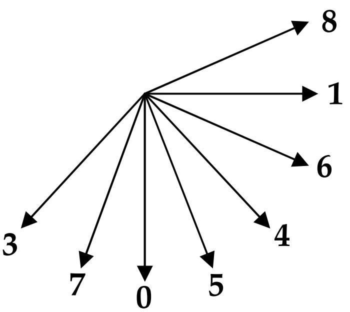
To achieve the best coding efficiency, H.264/AVC employs the rate-distortion (RD) optimization technique to get the best result consider of the visual quality and bit rate. For a macroblock in I-slice, RD optimization exhaustively searches the predefined 13 intra modes (9 modes for 4x4 block and 4 modes for 16x16 block) to produce the best encode mode for this macroblock. However, the latest H.264/AVC standard also defines 9 intra prediction modes for 8x8 block, for simplicity in this chapter only intra 4x4 and 16x16 blocks are considered. When a macroblock is in P-slice, it may be coded in intra mode or inter mode, so RD optimization employs a brute force algorithm to search through all possible inter modes and intra modes to find the best choice. RD optimization ofcourse will get the best encode mode for a macroblock, but it is at the great expanse of higher computational complexity at the encoder. To reduce computational complexity of H.264/AVC, a number of efforts have been made to explore the fast algorithm in motion estimation, intra mode prediction and inter mode prediction for H.264/AVC video coding (Chen et al., 2002; Feng et al., 2005; Han & Lee, 2005; Kim et al., 2006; Lim et al., 2003; Sarwer et al., 2008; Sarwer & Wu, 2009; Wu et al., 2005; Yang et al., 2004). The number of inter modes is reduced by using the amplitude of edge vector (Lim et al., 2003). To reduce the complexity of intra mode decision, H.264/AVC reference software suggested sum of absolute difference (SAD) and sum of absolute transform difference (SATD) based cost functions. These two cost functions reduce computation significantly but performance of RD characteristics is not good enough. In (Chiang & Zhang, 1997) and (Coebera & Lei, 1999) rate models were observed from the quantizer (Q)-domain. But these are considered only for rate control. To improve the RD performance, an enhanced cost function for intra 4x4 mode decisions was proposed in (Tseng et al., 2006). In this cost function, sum of absolute integer transform difference (SAITD) is used in distortion part and a rate prediction algorithm is used in rate part. The major drawback of this cost is that the bit estimation method cannot give the very good estimation.

In this chapter, we propose a shortcut way to get the number of entropy coded bits as soon as the transform coefficients are quantized. A method for estimation of rate for cost function of intra and inter mode decision is proposed. This method is based on the properties of context-based adaptive variable length coding (CAVLC) and observation of VLC tables. The total number of bits need to encode a quantized residual block is predicted by estimating the rate of each symbols of CAVLC separately.

The remainder of this chapter is organized as follows. Section 2 provides the review of intra and inter mode and rate distortion optimized mode decision technique. In section 3, Context-based adaptive variable length coding (CAVLC) is briefly described. In section 4, we present the proposed fast bit rate estimation method. The performance results of the proposed method are presented in section 5. Finally, section 6 concludes the chapter.

M	A	B	C	D	E	F	G	H
I	a	b	c	d				
J	e	f	g	h				
K	i	j	k	l				
L	m	n	o	p				

(a)



(b)

Fig. 1. (a) A 4x4 block with elements (a to p) which are predicted by its neighbouring pixels (b) Eight prediction direction for I4MB prediction

2. Overview of Mode Decision

2.1 Intra mode in H.264/AVC

The H.264/AVC video coding standard supports intra prediction for variable block size. If a macroblock (MB) is encoded in intra mode, a predicted block is formed based on previously encoded and reconstructed upper and left blocks so it exploits the spatial correlation between the adjacent MB. Then the residual data between the current to be coded MB and predicted one is DCT transformed, quantized and entropy coded. For each luma samples, the prediction block may be formed for each 4x4 block (denoted as I4MB) or for an entire MB (denoted as I16MB). When using the I4MB prediction, each 4x4 block of the luma component utilizes one of the nine prediction modes. Besides DC prediction, eight directional modes are specified. When utilizing I16MB, which is well suited for homogeneous area of image, 4 prediction modes are supported. Another 4 prediction modes are used to predict the U and V 8x8 chroma blocks. Nine prediction modes of 4x4 luma block are shown in Figure 1(a) and Figure (b). In addition, the prediction block is calculated based on the samples labeled A-M.

2.2 Inter mode in H.264/AVC

Inter-prediction is to reduce the temporal correlation with help of motion estimation and compensation. In H.264, the current picture can be partitioned into the MBs or the smaller blocks. A MB of 16x16 luma samples can be partitioned into smaller block sizes up to 4x 4. There are altogether conceptually 7 different block sizes (16x16, 16x8, 8x16, 8x8, 8x4, 4x8, 4x4) that are used in a MB that is encoded in inter mode. These block sizes can be classified as 16x16, 16x8, and 8x16 and P8x8. Each 8x8 block of P8x8 MB can be one of the subtypes such as 8x8, 8x4, 4x8 or 4x4. Figure 2 shows the different block sizes in a MB of inter mode. The smaller block size requires larger number of bits to signal the motion vectors and extra data of the type of partition, however the motion compensated residual data can be reduced. Therefore, the choice of partition size depends on the input video characteristics. In general, a large partition size is appropriate for homogeneous areas of the frame and a small partition size may be beneficial for detailed areas.

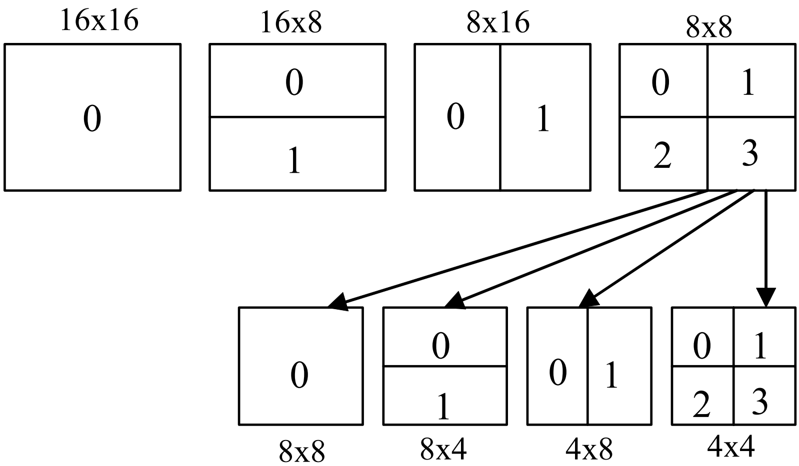


Fig. 2. Variable block sizes of macroblock of INTER mode

2.3 Rate Distortion optimized mode decision

To take the full advantages of all modes, the H.264 encoder can determine the mode that meets the best RD tradeoff using RD optimization mode decision scheme. The optimization approach is based on the assumption that the distortion and rate incurred in coding a MB are independent each other. Let us denote S_t as a block of any rectangular size in a frame at time t ; while $C_{t-\tau}$ is a reconstructed block of the same block size as S_t located in the previously coded frame at time $t-\tau$ ($\tau = 0$ in intra frame coding). Then the best mode for every block that produces the minimum rate-distortion cost is given by

$$J_{RD}(S_t, C_{t-\tau}, m \mid QP, \lambda_m) = SSD(S_t, C_{t-\tau}, m \mid QP) + \lambda_m R(S_t, C_{t-\tau}, m \mid QP) \tag{1}$$

where QP is the MB quantization parameter, λ_m is the Lagrangian multiplier and m is the candidate mode. In (Sullivan & Wiegand, 1998), a strong connection between the local Lagrangian multiplier and the QP was found experimentally as

$$\lambda_m = 0.85 \times 2^{(QP-12)/3} \tag{2}$$

In (1), SSD is the sum of squared difference between original block and reconstructed block, defined separately in terms of intra and inter frame coding

$$SSD_{\text{intra}}(S_t, C_t, m \mid QP) = \sum_i \sum_j \left| S_t(i,j) - C_t(i,j,m \mid QP) \right|^2 \tag{3}$$

$$SSD_{\text{inter}}(S_t, C_{t-\tau}, m \mid QP) = \sum_i \sum_j \left| S_t(i,j) - C_{t-\tau}(i+mv_i, j+mv_j, m \mid QP) \right|^2 \tag{4}$$

where (i, j) represent the ith and jth element and (mv_i, mv_j) represents the motion vector in the inter-frame case.

The R in (1) reflects the number of bits associated with choosing the mode which includes the bit consumption of quantized transform coefficients, motion vector data and the header. Thus R can be written as

$$R = R_{header} + R_{motion} + R_{res} \tag{5}$$

where R_{header} and R_{motion} means the number of bits need for header information and motion vectors respectively. It should be noted that in case of intra frame coding $R_{motion} = 0$. R_{res} indicates the number of bits need to encode a quantized residual block.

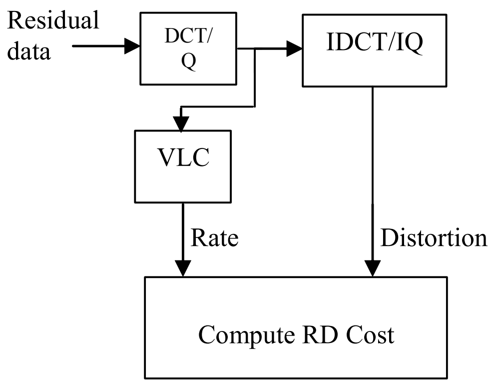


Fig. 3. Computation of RD cost

In intra-frame coding, the final mode decision is selected by the member (either from *I4MB* or *I16MB*) that minimizes the Lagrangian cost in (1). In inter-frame coding, motion estimations with 7 different block-size patterns, as well as the other members in three members (*I4MB*, *I16MB*, and *SKIP*), are calculated. The final decision is determined by the mode that produces the least Lagrangian cost among the available modes. Figure 3 shows the computational process of RD cost of H.264 video coding standard. As indicated in Figure 3, in order to compute RD cost for each mode, same operation of forward and inverse transform/quantization and variable length coding is repetitively performed. All of these processing explains the high complexity of RD cost calculation.
In order to calculate RD cost using (1), 4×4 integer *DCT+Q*, CAVLC, and *(DCT+Q)-1* in units of 4×4 blocks should be performed. If the amount of RD cost computation in P8×8 mode and Inter16×16 mode and I4MB mode in a MB unit are calculated, it is shown that those computations in Inter16×16 mode are performed 16 times, while those computations in P8×8 mode are performed 64 times and those computations in I4MB mode are performed 144 times. To reduce computation, several fast mode-decision approaches proposed in (Feng et al., 2005; Kim et al., 2006; Yang et al., 2004) focused on how to eliminate unnecessary modes. However, it is noted that the remaining modes should be performed RD optimization. So it still introduces a great deal of computation complexity because of the transform, quantization/inverse quantization and entropy coding to get distortion and bit-rate. A bit rate estimator by modeling the coded bits consumption as a function of the number and levels of the nonzero quantized transform coefficients is introduced in (Yu et al., 2006). In (Tseng et al., 2006), a rate predictor for 4x4 quantized residual blocks is proposed as follows

$$R_{e(res)} = \alpha T_c - \beta T_o + 4P \tag{6}$$

where T_c is the total number of non-zero coefficients, T_o is the number of trailing +/-1 values, α and β are the positive constants and the P equal to 0 for the probable mode and 1 for the other modes. From simulation it is shown that this method cannot achieve very good rate estimation. This is because only two parameters are not enough to estimate the bit rate. In this chapter a rate prediction scheme is introduced based on the properties of CAVLC entropy coding method. Before we propose the new estimation method, review of the CAVLC encoding method is described at following section.

3. Review of Context based adaptive variable length coding (CAVLC)

Context based Adaptive Variable Length Coding (CAVLC) is designed to take advantage of several characteristics of quantized 4x4 blocks. The block is encoded by five syntax elements. These elements are described as follows:

1. *coeff_token:* Both the total number of nonzero coefficients and the number of trailing +/- 1s are coded as combined event. If number of trailing +/- 1s is greater than 3, last 3 trailing +/-1s of zig-zig ordered block are consider as trailing +/-1s and remaining trailing +/-1s are consider as normal coefficient. One out of 4 VLC tables is used (ITU-T Rec., 2002). Since the number of non-zero coefficients in neighboring blocks are usually correlated, the selection of VLC table is depends on the number of non-zero coefficients in neighboring blocks (Richardson, 2003).

2. *Sign of trailing +/- 1:* One bit is used to signal sign information of trailing +/- 1s. 0 is used for positive and 1 is used for negative.
3. *Level:* The magnitude (level) of non-zero coefficients gets larger near the DC coefficient and gets smaller around the high frequency coefficients. CAVLC takes advantage of this by making the choice of the VLC look-up table for the level adaptive in a way where the choice depends on the recently coded levels (Richardson, 2003). One out of 7 VLC tables is used to encode the level information (ITU-T Rec., 2002).
4. *Total_zeros:* The codeword Total_zeros is the number of zeros between the last non-zero coefficient of the zig-zag scan and its start. One out of 15 VLC tables is chosen based on the number of non-zero coefficients (ITU-T Rec., 2002).
5. *Run_before:* Run_before means the number of preceding zeros before each non-zero coefficient and Zeros_left called the number of zeros left before each non-zero coefficient. Based on Run_before and Zeros_left, one out of 7 VLC tables is used to encode this element (ITU-T Rec., 2002).

Element	**Value**	**Code**
coeff_token	TotalCoeffs=6, T1s=3 (use VLC0)	00000100
T1 sign (5)	+	0
T1 sign (4)	-	1
T1 sign (3)	+	0
Level (2)	+1 (use Level_VLC0)	1
Level (1)	-2 (use Level_VLC1)	011
Level (0)	+4 (use Level_VLC1)	00010
TotalZeros	2	111
run_before(5)	ZerosLeft=2; run_before=0	1
run_before(4)	ZerosLeft=2; run_before=0	1
run_before(3)	ZerosLeft=2; run_before=1	01
run_before(2)	ZerosLeft=1; run_before=1	0
run_before(1 & 0)	ZerosLeft=0; run_before=0	No code required

Table 1. Example of CAVLC

For example, the coefficients in the zig-zag order of a 4x4 block are [4,-2,0,1,0,1,-1,1,0,0…]. Here total number of non-zero coefficients (T_c) is 6, total number of zero before the last nonzero coefficients (T_z) is 2, number of trailing one's (T_o) is 3. Based on the VLC tables (ITU-T Rec., 2002) as shown in Table 1, the transmitted bit stream for this block is 000001000101011000101111010.

4. Proposed Fast Bit Rate estimation Method

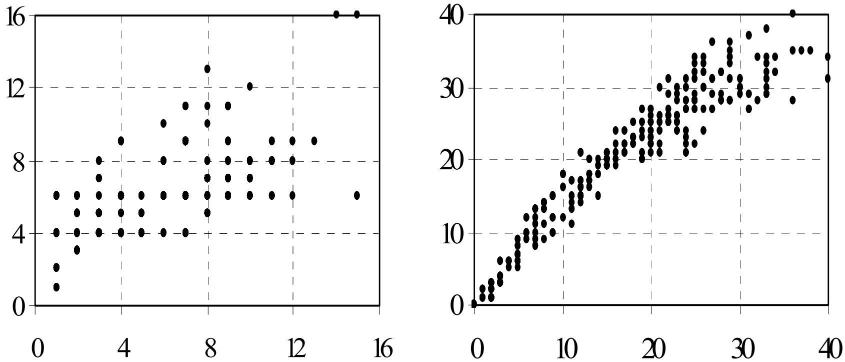
To estimate the bits for quantized transform coefficients, we estimate the number of bits for each of five different types of symbols of CAVLC separately.

1. *The coefficient token (the number of coefficients, the number of trailing ones):* Four VLC tables are used for encoding coefficient token. Selection of VLC table is context adaptive.

Figure 4(a) shows the plot of actual bit rate to encode the coefficient token versus the number of coefficients of foreman video sequence at QP=28. Similar results were found for other video sequences. It is clearly shown that bit consumption to encode the coefficient token is increased with number of coefficients. Based on VLC tables (ITU-T Rec., 2002), it is also shown that bit rate for coefficient token is decreased with number of trailing ones. Based on this criteria, we propose the number of bits require to encode the coefficient token is

$$R_{coff} = w_1 T_c - w_2 T_o + w_3 \tag{7}$$

where T_c and T_o are same as equation (6). These w_1, w_2 and w_3 are weighting constants. In order to set the weighting factors, we have done several experiments for different video sequences (akiyo, foreman, stefan, mobile, table tennis, paris) with QCIF format at different QP values. We have observed rate-distortion performance of these video sequences at different combinations of weighting factor. Better rate-distortion performance was found at $w_1 = w_2 = 1$, $w_3 = 0$.



(a) (b)

Fig. 4. Plot of (a) no. of non-zero coefficients vs true value of coefficient token (X-axis: T_c, Y-axis: True rate of Coeff_token) (b) SAT_l vs actual rate of level (X-axis: SAT_l, Y-axis: True rate of level)

2. *The sign of trailing ones:* For each T_o, a single bit encodes the sign (0=+, 1=-). So bit consumption to encode the trailing ones as follows:

$$R_{trail1} = T_o \tag{8}$$

3. *The level of nonzero coefficients:* From the observation of level-VLC tables (ITU-T Rec., 2002), it is shown that bit requirement is increased with magnitude of non-zero coefficients. Number of bits to encode the level information is proposed as follows:

$$R_{level} = w_4 SAT_l \tag{9}$$

with the SAT_l given by

$$SAT_l = \sum_{k=1}^{Tc} |L_k| \tag{10}$$

where $|L_k|$ is the absolute value of k_{th} non-zero coefficient, SAT_l is the sum of absolute values of all levels of quantized transform residual block. w_4 is a positive constant. Suppose two coefficients are encoded using same level_VLC table. If the magnitude of first coefficient $(|L_1|)$ is larger than that of second coefficient $(|L_2|)$, from level_VLC table it is shown that rate for first coefficient $R(L_1)$ also greater than that of second coefficient $R(L_2)$. So equation (9) is valid. Let us consider two coefficients are encoded using multiple level_VLC tables. If the earlier coefficient is not larger than the other coefficient, by observing the all level_VLC tables (ITU-T Rec., 2002) there is $R(L_1) > R(L_2)$ if $|L_1| > |L_2|$. In some times, $R(L_1) > R(L_2)$ if $|L_1| < |L_2|$ but fortunately this event does not occur high frequently and is slightly influence the estimation result. Figure 4(b) shows the plot of actual bit rate to encode the level information with SAT_l of foreman video sequence at QP=28. Similar results were found for other video sequences. By changing the value of w_4, we have observed RD performance of different sequences. Better results were found with w_4=1.

4. *Encode the total number of zeros before the last coefficient:* From the observation of total zero VLC tables, it is shown than bit consumption to encode the total zero is increased with number of total zero. So we can propose the estimated bits for total zero is as follows

$$R_{zero} = w_5 T_z \tag{11}$$

where w_5 is a positive constant and T_z is the total number of zeros before the last non zero coefficients. Here w_5=1, which is found in similar way of w_4.

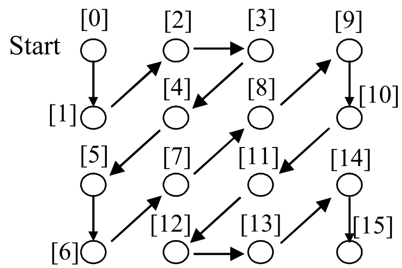


Fig. 5. Zig-zag scan and corresponding frequency of 4x4 luma block

5. *Encode each run of zeros:* After the DCT transform, the high frequency coefficient usually has small energy. By quantization, more zeros are found at the high frequency position of quantized transform block. So the value of *Run* for the high frequency non-zero coefficients is larger. From the observation of run VLC tables, it is shown that more bits

are required for large value of run. So bit consumption is higher to encode the run of high frequency non-zero coefficients. Based on this idea, we propose the rate for run of each non-zero coefficients as follow:

$$R_{run_{(k)}} = w_6 f_k \ , \ 0 \geq f_k \leq 15 \tag{12}$$

where, f_k is the frequency of k_{th} non-zero coefficient of recorded block and w_6 is the positive constant. For example, given a string of coefficients [0, 3, 0, 1,-1,-1, 0, 1, 0, 0…], frequency of first non-zero coefficient (3) is 1 and frequency of last non-zero coefficient (1) is 7. The weighting factor w_6=0.3 is found in similar way of w_4. Figure 5 shows the zig-zag scan of a residual block with corresponding value of frequency.

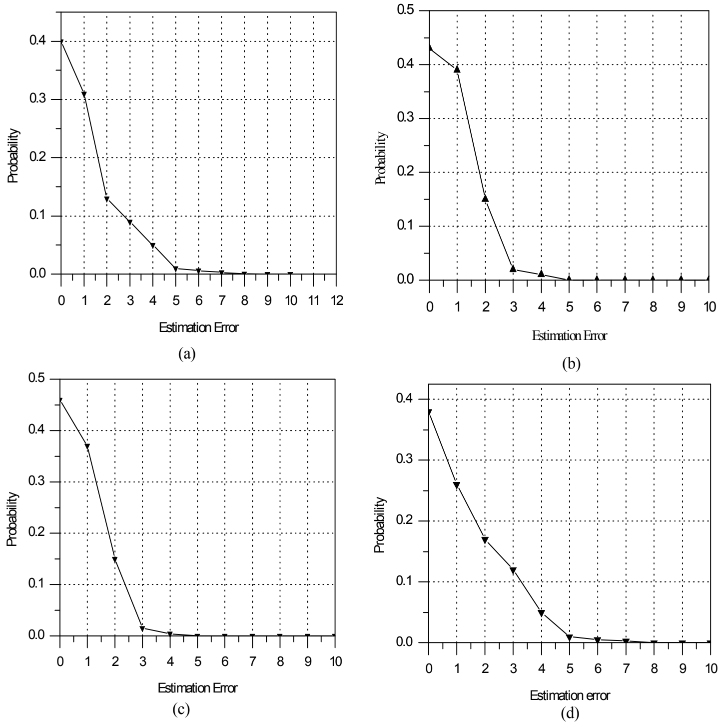


Fig. 6. Probability of estimation error of (a) Coeff_token (b) level (c) total zero (d) run

From above analysis, we have estimated the bits needed to encode a 4x4 residual block ($R_{est_{(res)}}$) is

$$\begin{aligned} R_{est_{(res)}} &= R_{coeff} + R_{trail1} + R_{level} + R_{zero} + \sum_{k=1}^{T_c} R_{run_{(k)}} \\ &= w_1 T_c - w_2 T_o + w_3 + T_o + w_4 SAT_l + w_5 T_z + \sum_{k=1}^{T_c} (w_6 f_k) \end{aligned} \tag{13}$$

By putting the values of different constants, the proposed rate estimator becomes

$$R_{est_{(res)}} = T_c + T_z + SAT_l + 0.3\sum_{k=1}^{T_c} f_k \tag{14}$$

Figure 6 shows the probability of estimation error of four different types of symbols. Estimation error of the symbol is the absolute difference between actual bit rate and estimated bit rate of that symbol. Y-coordinate of Figure 6 is the probability of corresponding estimation error. It is shown that most of the estimation errors of each symbol are between 0 to 4 and the probability of each symbol that the estimated rate perfectly match with CAVLC is about 40%.

5. Simulation Results

To verify the proposed technique, JM 8.3 reference software is used in simulation. Six well known video sequences are used as test materials. The test conditions are as follows:

a) Hadamard transform is used
b) RD optimization is enabled
c) CAVLC is enabled.
d) Frame rate is 30
e) MV search range is ± 32 pels for QCIF and CIF
f) Fast motion estimation algorithm is used (Chen et al., 2002).

A group of experiments were carried out on the test sequences with different quantization parameters. Comparison results were produced based on the percentage of difference of coding time (Δ T %), the PSNR difference (ΔP_{snr}) and percentage of the bit rate difference (Δ Bit %). In order to evaluate complexity reduction, Δ T (%) is defined as follows

$$\Delta T = \frac{T_{original} - T_{proposed}}{T_{original}} \times 100\% \tag{15}$$

where, $T_{original}$ denotes the total encoding time of the JM 8.3 encoder with rate distortion optimization and $T_{proposed}$ is the total encoding time with proposed fast rate estimation technique.

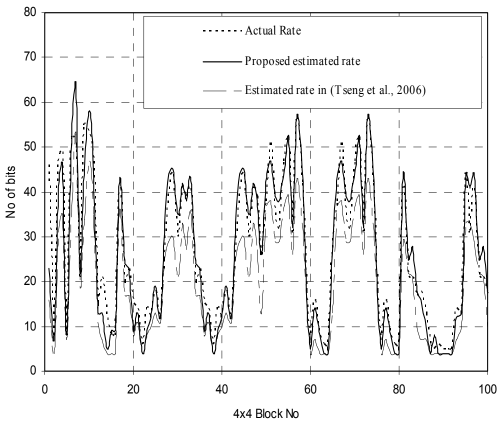


Fig. 7. Comparison of our proposed method with rate estimation method described in (Tseng et al., 2006)

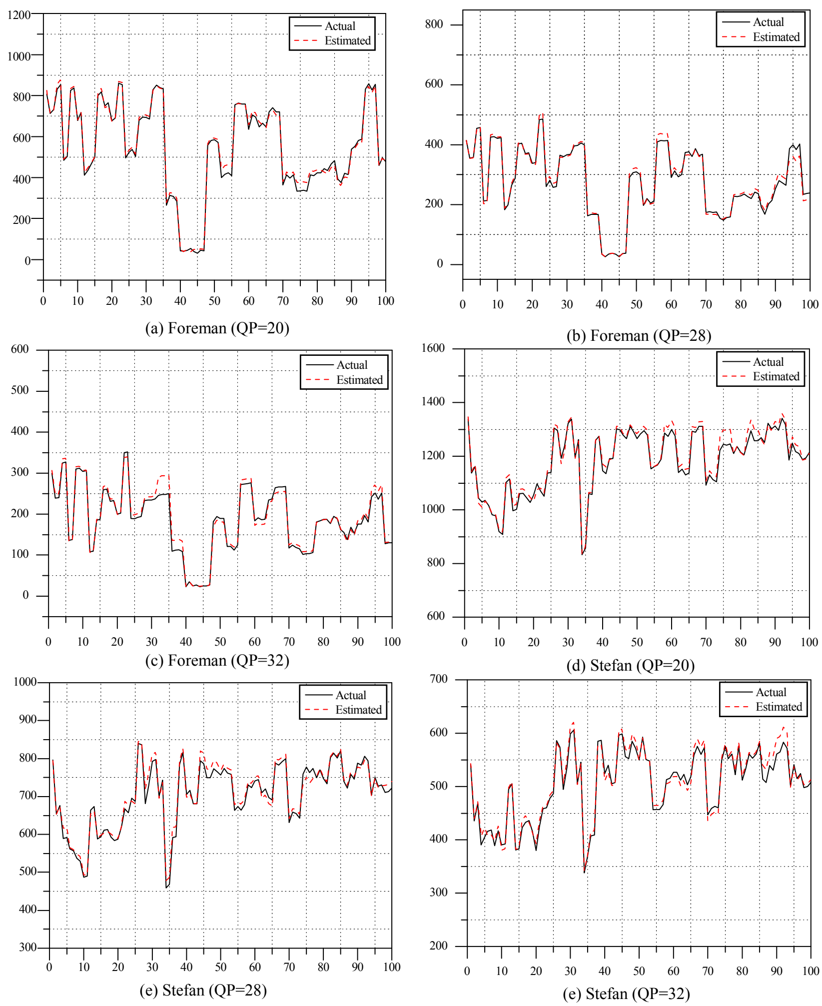


Fig. 8. Curves of the estimated and the actual rates of first 100 macroblocks of I frame of intra coding of foreman and Stefan sequences (X-axis: Macroblock number, Y-axis: number of bits)

5.1 Experiments with all intra frame sequences

In this experiment, a total number of 100 frames are used for each sequence, and period of I-frame is set to 1, i.e., all the frames in the sequence are intracoded. In (Tseng et al., 2006), a rate predictor for 4x4 intra mode decisions is introduced based on the number of non-zero coefficients and number of trailing ones. Figure 7 shows the comparison of our proposed method with actual rate and the rate predictor described in (Tseng et al., 2006) for the Foreman sequence. Data is collected from the first 100 4x4 block of first frame of IIIIII sequence. QP factor is set as 28. It is shown that the proposed method is very closely matched with actual rate as compared to rate predictor in (Tseng et al., 2006). By using the proposed estimation method, Figure 8 shows the predicted rates $R_{est}(= R_{header} + R_{motion} + R_{est\,(res)})$ and actual rates $R(= R_{header} + R_{motion} + R_{res})$ are very closely identical, which are obtained from first 100 MBs of IIIIII sequence of Foreman-QCIF and Stefan_CIF at three different (20, 28, 32) QP values. The proposed estimation achieves a precise prediction in intra frame coding.

Sequence	**QP**	Δ **P**$_{snr}$	Δ **Bit%**
Akiyo (QCIF)	20	-0.06	+0.64
	24	-0.05	+0.50
	28	+0.02	+0.53
	32	+0.17	+0.70
	36	+0.12	+1.49
	40	+0.16	+2.59
Foreman (QCIF)	20	-0.06	+0.49
	24	-0.05	+0.89
	28	-0.02	+1.16
	32	+0.10	+1.75
	36	+0.11	+2.35
	40	+0.15	+3.86
Mobile (QCIF)	20	-0.16	+0.85
	24	-0.09	+0.78
	28	-0.10	+0.54
	32	-0.08	+0.27
	36	-0.06	+0.70
	40	+0.03	+1.83
Paris (CIF)	20	-0.16	+0.46
	24	-0.11	+0.25
	28	-0.10	+0.45
	32	-0.06	+0.88
	36	-0.02	+0.64
	40	+0.04	+1.61
Table Tennis (CIF)	20	-0.14	+0.31
	24	-0.12	+0.36
	28	-0.04	+1.01
	32	+0.01	+0.88
	36	+0.02	+1.48
	40	+0.06	+2.23
Stefan (CIF)	20	-0.19	+0.62
	24	-0.12	+0.44
	28	-0.11	+0.42
	32	-0.02	+0.68
	36	-0.02	+1.69
	40	+0.13	+3.15

Table 2. Performance of PSNR and bit rate of proposed plgorithm while all frames are intra coded

Sequence	Quantization Parameter, QP					
	20	24	28	32	36	40
Akiyo_QCIF	51.21 %	47.29 %	44.11 %	39.34 %	35.35 %	30.42 %
Foreman_QCIF	55.20 %	51.16 %	46.66 %	42.42 %	37.28 %	33.33 %
Mobile_QCIF	62.87 %	61.78 %	59.45 %	57.14 %	51.21 %	47.14 %
Paris_CIF	58.11 %	54.96 %	52.81 %	47.94 %	43.30 %	41.08 %
Table_tennis_CIF	57.04 %	53.56 %	49.54 %	43.41 %	39.11 %	35.21 %
Stefan_CIF	57.41 %	54.12 %	50.55 %	44.96 %	39.20 %	34.52 %
Average = 47.36 %						

Table 3. Computational complexity reduction of proposed algorithm while all frames are intra coded

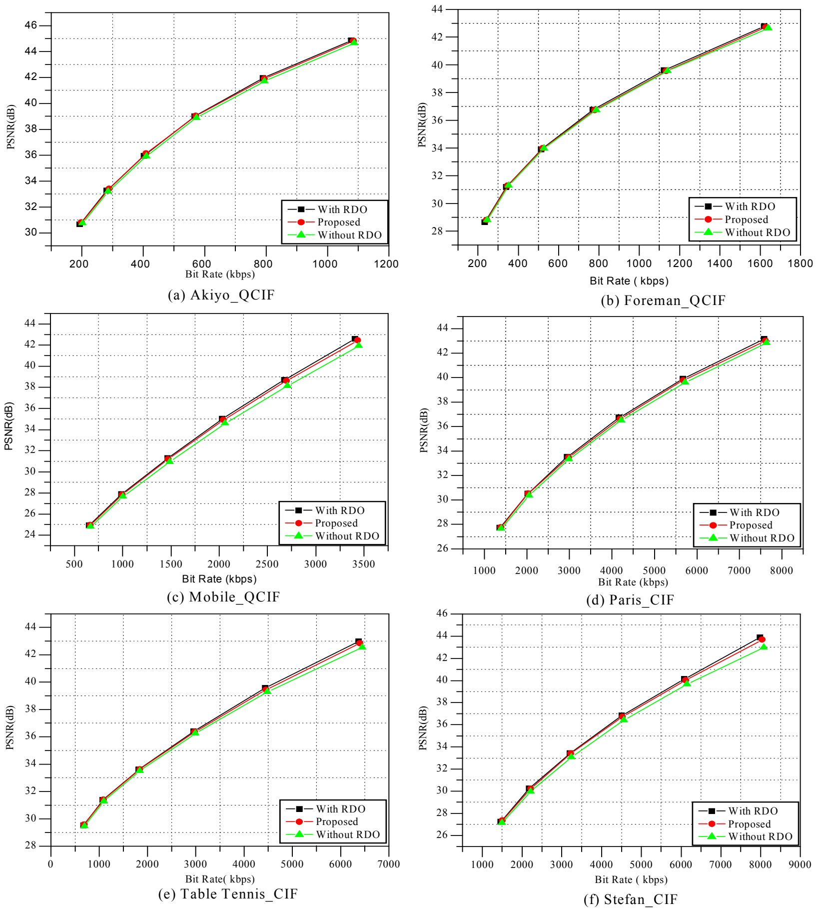


Fig. 9. Rate-distortion performance of proposed rate estimation method of different video sequences while all frames are intra coded.

To evaluate the rate distortion performance six different sequences (Akiyo, Foreman, Mobile, Paris, Table tennis and Stefan) are used in simulation. As shown in Table 2, it is clear that PSNR loss and bit rate increment is negligible. Figure 9 shows the rate-distortion curves of different sequences. The proposed method is very close with RD optimized curve. Comparing with the original H.264/AVC encoder with RD optimization, the proposed algorithm achieves about 47% time reduction of total encoding time on average. As shown in Figure 10, for the sequence "Mobile" and "Paris", the coding speed is high because both the sequences contains high detail such as different books in bookshelf of "Paris". On the other hand sequence "Akiyo" show strong spatial homogeneity and low detail. Therefore, the time saving of this sequence is not much as compared to other sequences. Figure 10 also indicates that percentage of complexity reduction is decreased with increasing the QP values. This is because there is large number of non-zero coefficients at small QP values. If true encoding process is used large entropy coding time is spent at small QP values whereas in our proposed method no entropy coding is required during mode-decision process.

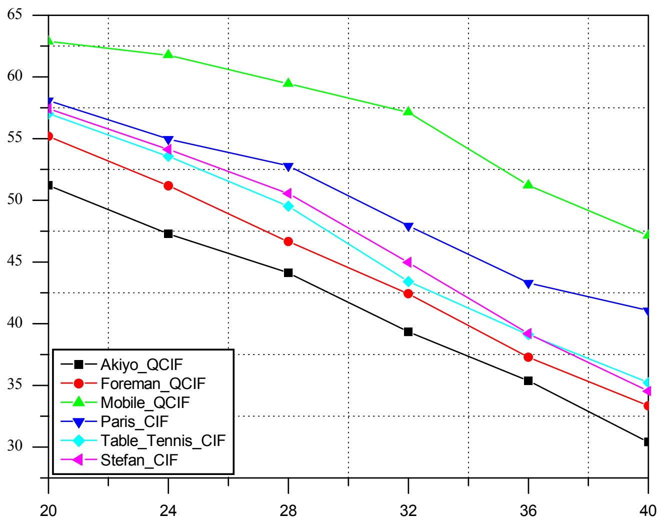


Fig. 10. Complexity reduction of proposed algorithm during intra frame coding (X-axis: QP; Y-axis: % of complexity reduction)

5.2 Experiments with IPP sequences

To evaluate the performance of proposed method during inter frame coding, several test video sequences are used. In this experiment total number of frames is 100 for each sequence, and the period of I-frame is 3. Figure 11 shows the actual and predicted rates of foreman and Stefan at different QP values. Data is generated as similar ways of intra frame coding.

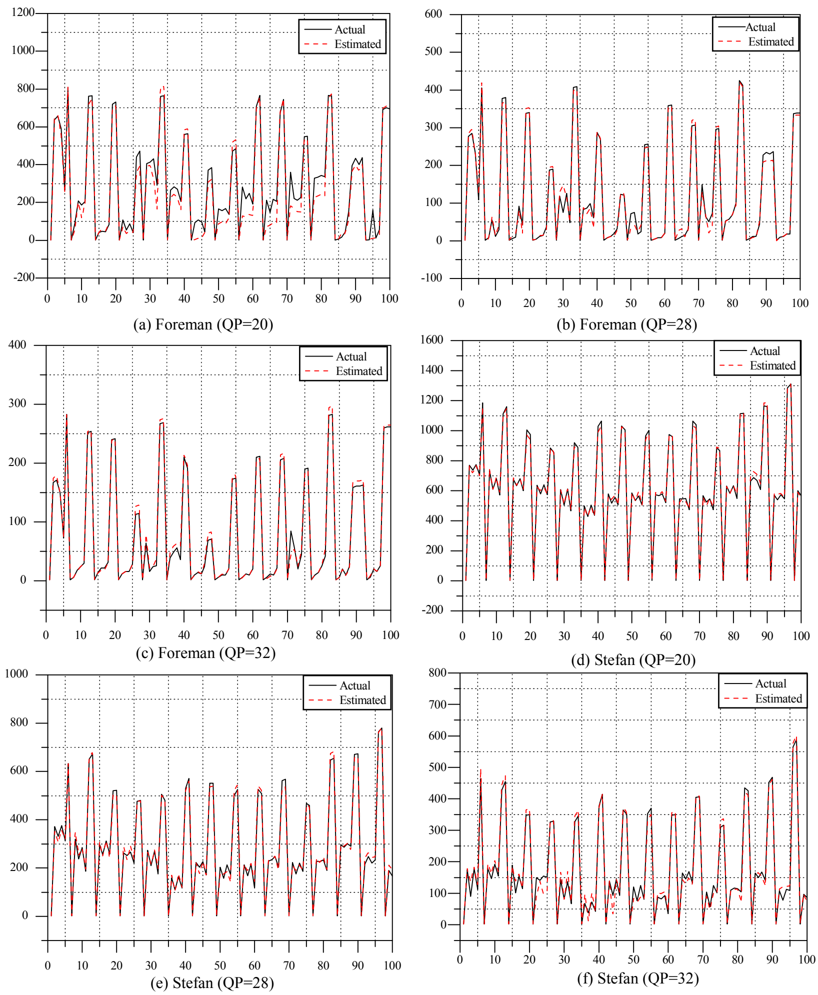


Fig. 11. Curves of the estimated and the actual rates of first 100 macroblocks of P frame of inter coding of foreman and Stefan sequences (X-axis: Macroblock number, Y-axis: number of bits)

Experimental results are tabulated as Table 4 which means that PSNR reduction and bit rate increments are negligible. The positive values mean increments whereas negative values mean decrements. From the experimental results in Table 5, it is observed that the proposed approach has reduced the encoding time by 34% on average. RD performances are given in Figure 12 which shows that the proposed method is closely matched with actual RD curves. Figure 13 presents the plot of computational complexity with quantization parameter of different sequences. Figure 13 show the general tendency that time saving increases as QP decreases. This is understandable that at small QP, coding quality is high and many detailed are retained. From this figure it is also shown that computation saving is large for those sequences which have high motion and large detail.

Sequence	QP	ΔP_{snr}	Δ Bit%
Akiyo (QCIF)	20	+0.01	+1.47
	24	+0.01	+1.38
	28	+0.09	+1.97
	32	+0.14	+2.79
	36	+0.11	+3.00
	40	+0.16	+2.89
Foreman (QCIF)	20	+0.02	+2.12
	24	+0.03	+2.23
	28	+0.05	+2.57
	32	+0.07	+3.01
	36	+0.13	+3.05
	40	+0.15	+4.44
Mobile (QCIF)	20	-0.05	+0.95
	24	-0.04	+0.99
	28	-0.03	+1.17
	32	-0.01	+2.39
	36	+0.02	+2.89
	40	+0.07	+2.94
Paris (CIF)	20	-0.05	+1.17
	24	-0.05	+1.37
	28	-0.02	+1.43
	32	+0.00	+2.38
	36	+0.02	+2.33
	40	+0.08	+3.21
Table Tennis (CIF)	20	-0.03	+2.20
	24	+0.01	+2.33
	28	+0.04	+2.52
	32	+0.06	+2.88
	36	+0.06	+2.95
	40	+0.06	+3.11
Stefan (CIF)	20	-0.10	+1.17
	24	-0.09	+0.08
	28	-0.03	+1.97
	32	-0.02	+3.01
	36	+0.07	+3.65
	40	+0.14	+4.74

Table 4. Performance of PSNR and Bit Rate of Proposed Algorithm of Inter frame (IPP sequences) coding

Sequence	Quantization Parameter, QP					
	20	24	28	32	36	40
Akiyo_QCIF	36.63 %	31.91 %	28.73 %	23.45 %	22.07 %	21.33 %
Foreman_QCIF	41.66 %	37.85 %	33.33 %	29.47 %	27.58 %	22.22 %
Mobile_QCIF	53.08 %	50.99 %	46.71 %	40.49 %	34.57 %	29.89 %
Paris_CIF	45.15 %	41.82 %	38.39 %	33.98 %	28.64 %	24.29 %
Table_tennis_CIF	41.61 %	36.59 %	32.45 %	28.16 %	25.94 %	23.87 %
Stefan_CIF	46.19 %	42.65 %	38.61 %	35.13 %	30.56 %	25.14 %
Average = 34.20 %						

Table 5. Computational complexity reduction of proposed algorithm during Inter frame (IPP sequences) coding

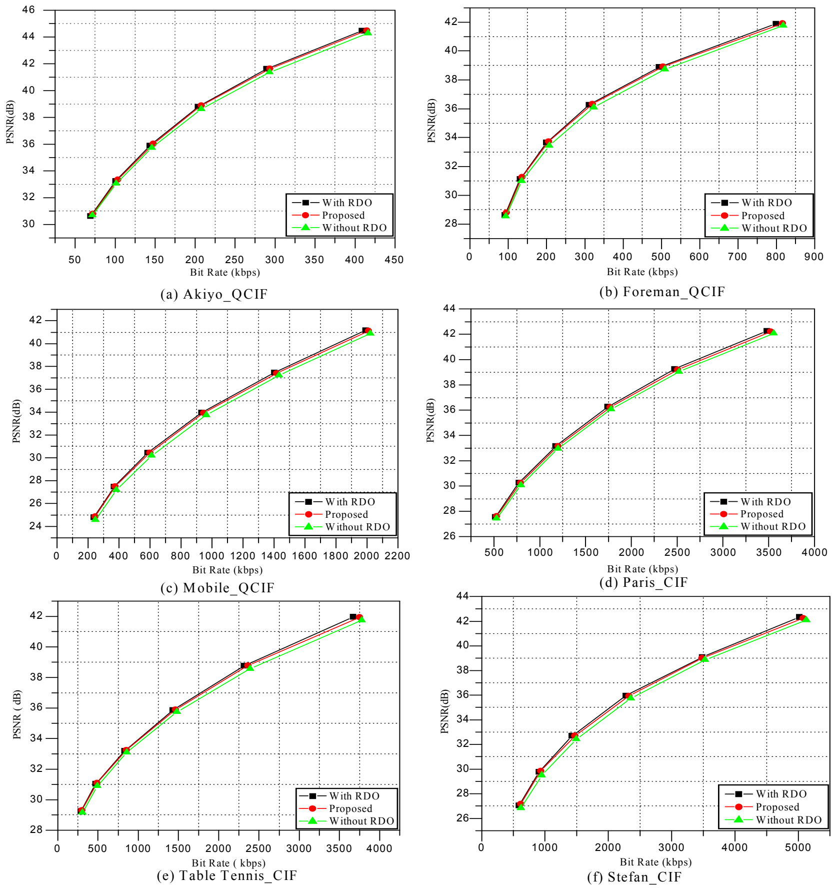


Fig. 12. Rate-distortion performance of proposed rate estimation method of different video sequences during Inter frame (IPP sequences) coding

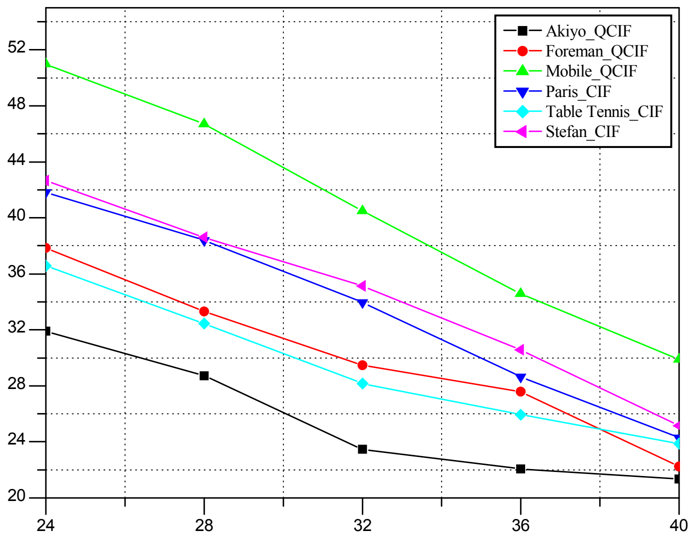


Fig. 13. Complexity reduction of proposed algorithm during IPP sequences (X-axis: QP; Y-axis: % of complexity reduction)

Sequence	Frame Skip	QP	Δ P_{snr}	Δ Bit%	Δ T%
Akiyo (QCIF)	1	20	+0.09	+3.84	30.41
		28	+0.14	+3.00	20.84
		32	+0.06	+3.02	17.76
	2	20	+0.11	+2.85	33.20
		28	+0.14	+3.55	25.99
		32	+0.10	+3.67	20.46
Foreman (QCIF)	1	20	+0.09	+3.24	36.86
		28	+0.11	+3.96	29.60
		32	+0.10	+2.98	23.88
	2	20	+0.06	+2.03	34.51
		28	+0.10	+4.13	26.25
		32	+0.13	+4.41	20.27
Mobile (QCIF)	1	20	+0.01	+3.04	46.26
		28	+0.11	+2.82	39.67
		32	+0.08	+3.33	35.35
	2	20	+0.04	+1.60	47.46
		28	+0.11	+3.10	40.06
		32	+0.08	+2.87	36.51

Table 6. Experimental results of IBPBP sequences

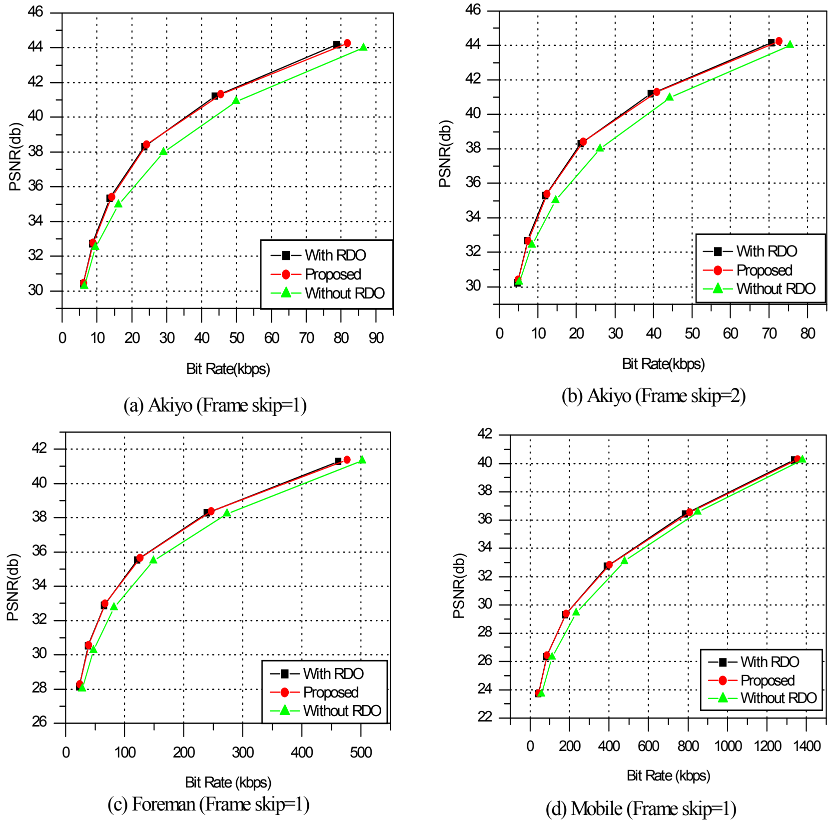


Fig. 14. Rate distortion performance of proposed method with IBPBP sequences

5.3 Experiments with IBPBP sequences

In this experiment, there is one B-frame between any two I- or P-frames with different frame skip values. Period of I-frames is set to 100. Number of frames is 100. Experimental results were tabulated at table 6. As shown in table 6, the proposed algorithm achieves time saving of about 32% (on average) with slight increment in bit rate. Rate-distortion performances of different sequences are shown in Figure 14 with different frame skip value.

5.4 Experiments with Full Search Motion Estimation

It is well known that motion estimation requires major portion of the processing power. In all of the previous experiments, fast motion estimation technique described in (Chen et al., 2002) is utilized. In order to show the complexity of proposed method with full search motion estimation technique, several video sequences are encoded. In this experiment, IPP sequence is used and number of frames is set to 50. Experimental results are tabulated at table 7. It is shown that bit rate increment is up to about 4%. Since the PSNR also increases, the resulting RD performance is very much closed with original one. The proposed algorithm reduces about 17% (on average) of computation time when full search motion estimation method is used.

Sequence	QP	Δ P_{snr}	Δ Bit%	Δ T%
Akiyo (QCIF)	20	+0.09	+3.96	12.53
	28	+0.12	+2.94	6.66
	32	+0.13	+3.65	5.72
Foreman (QCIF)	20	+0.07	+2.86	20.66
	28	+0.04	+3.59	9.09
	32	+0.08	+3.50	6.62
Mobile (QCIF)	20	-0.01	+0.64	32.19
	28	+0.11	+2.52	24.42
	32	+0.10	+4.21	21.95
Stefan (QCIF)	20	+0.02	+1.48	24.80
	28	+0.07	+2.46	20.11
	32	+0.08	+2.57	15.78

Table 7. Experimental results with full search motion estimation

Sequence	QP	Δ P_{snr}	Δ Bit%	Δ T%
Akiyo (QCIF)	20	+0.02	-0.91	-4.11
	28	+0.01	-1.38	-4.54
	32	+0.05	-1.47	-4.02
Foreman (QCIF)	20	+0.05	-1.14	-4.47
	28	+0.07	-0.77	-4.61
	32	+0.11	-0.67	-4.68
Mobile (QCIF)	20	+0.22	-0.29	-5.86
	28	+0.13	-0.78	-5.17
	32	+0.06	-1.02	-5.02
Stefan (CIF)	20	+0.33	-0.52	-5.59
	28	+0.10	-0.88	-5.12
	32	+0.11	-1.23	-4.92

Table 8. Comparison of proposed method with rate estimation method stated in (Tseng et al., 2006)

5.5 Comparison with other methods

In this experiment, the proposed method is compared with two different methods in terms of RD performance and complexity. Fast motion estimation technique described in (Chen et al., 2002) is used. Table 8 shows the comparison of our proposed method with rate estimation method defined in (Tseng et al., 2006). In this simulation, only intra 4x4 modes are used. RD performance of proposed method is better from method (Tseng et al., 2006). Positive value of Δ T% means complexity reduction and negative value means increment of complexity. It is shown that as compared with (Tseng et al., 2006), this algorithm reduces about 1% of bit rate and increases about 0.10 db PSNR with slight increment (about 4%) of computational time. Table 9 shows the RD performance as well as complexity reduction of this algorithm as compared with fast inter mode decision algorithm described in (Lim et al., 2003). IPP sequences are used in this comparison. Complexity reduction is about 14% (on average) for medium and high motion i.e. foreman, mobile and stefan sequences. It is also shown that the proposed method increases about 7% (on average) of computation of low motion sequence such as akiyo as compared with mode decision method stated in (Lim et al., 2003). RD performance is better for all types of sequences.

Sequence	QP	Δ P_{snr}	Δ Bit%	Δ T%
Akiyo (QCIF)	20	+0.05	-0.24	-10.71
	28	+0.21	-0.11	-6.56
	32	+0.15	-0.67	-4.16
Foreman (QCIF)	20	+0.10	-0.49	9.09
	28	+0.11	-0.95	7.93
	32	+0.12	-1.48	6.12
Mobile (QCIF)	20	+0.03	-0.39	31.53
	28	+0.05	-0.42	21.50
	32	+0.04	-0.84	14.28
Stefan (CIF)	20	+0.08	-0.35	19.60
	28	+0.04	-0.60	10.94
	32	+0.09	-1.03	8.34

Table 9. Comparison of proposed method with fast inter mode decision method stated in (Lim et al., 2003)

QP	Akiyo (QCIF)		Foreman (QCIF)		Stefan(QCIF)	
	$\Delta Rate$ %	$\Delta PSNR$	$\Delta Rate$	$\Delta PSNR$	$\Delta Rate$	$\Delta PSNR$
20	1.00	-0.06	1.30	0	0.14	-0.14
24	2.16	-0.02	1.68	+0.04	0.61	-0.15
28	2.57	+0.04	2.68	+0.08	1.34	-0.08
32	2.23	-0.01	3.68	+0.09	1.85	-0.06
36	3.39	0	3.88	+0.12	2.92	-0.02
40	3.22	-0.03	4.41	+0.06	4.09	+0.01

Table 10. Experimental results with CABAC entropy coding method

QP	Akiyo (QCIF)	Foreman (QCIF)	Stefan(QCIF)
20	32.20	40.58	46.25
24	27.78	39.13	43.42
28	23.53	25.45	37.68
32	22.45	25.00	35.48
36	15.22	20.41	29.82
40	11.36	17.02	24.53

Table 11. Percentage of complexity reduction with CABAC

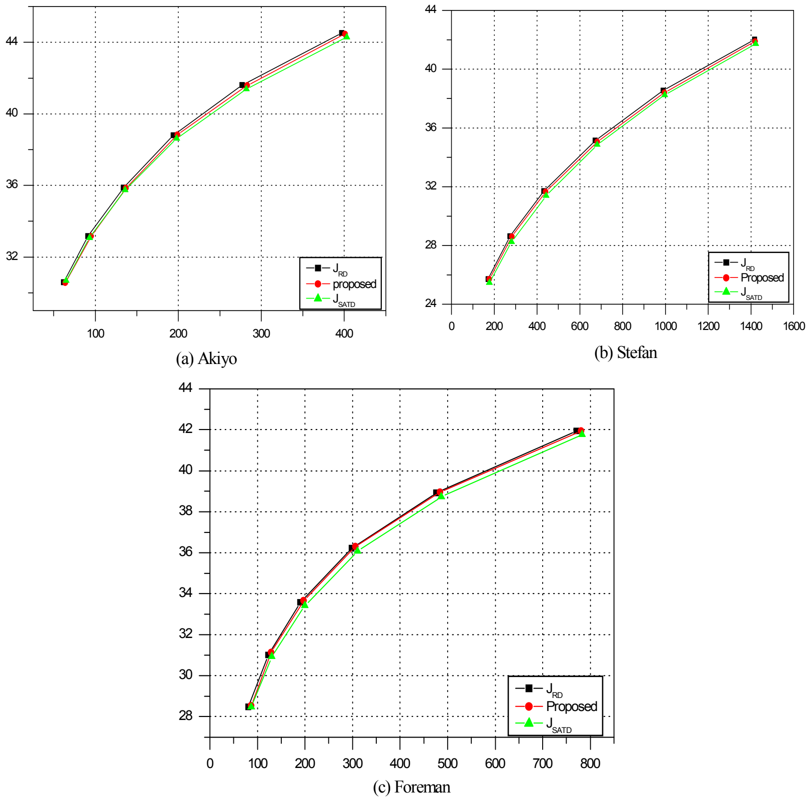


Fig. 15. RD performance with CABAC entropy coding method

5.6 Experiments with CABAC entropy coding method

In all of the above experiments CAVLC entropy coding method was used. Although the proposed algorithm is developed based on CAVLC, it is also suitable during mode decision of CABAC. This is because the proposed algorithm is used only for RD optimization. After selection of best mode, true entropy coding is used. To justify the performance of proposed rate estimation with CABAC entropy coding method, some video sequences are tested. IPP sequence is used in simulations and number of frames was set to 50. Table 10 shows the experimental results in terms of PSNR and bit rate. Table 11 shows the complexity reduction is also significant. The rate-distortion performance is given in Figure 15, which shows that the proposed algorithm is also suitable while CABAC entropy coding method is used. The proposed method is very close to RD optimized curves.

6. Conclusion

In this chapter, simple and fast bit rate estimation method for mode decision of H.264/AVC is proposed. This method is based on the VLC tables used in CAVLC entropy coding method. The experimental results verified that the proposed technique is suitable both for inter and intra mode decision of H.264/AVC. With the proposed scheme, entropy coding can be skipped during the mode decision process. The proposed technique reduces encoding time by 47 %, 34% and 32 % on average during intra frame, IPP sequences and IBPBPBP sequences respectively. The RD performance of this algorithm is better than both methods stated in reference (Tseng et al., 2006) and (Lim et al., 2003). The proposed method is also suitable for mode decision while CABAC entropy coding method is utilized.

7. Acknowledgement

The work described in this chapter was substantially supported by a GRF grant from Hong Kong SAR Government with project number of 9041251 (CityU 119207).

8. References

Chiang T. & Zhang Y. Q. (1997) A new rate control scheme using quadratic rate distortion model, *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 7, No. 1, February 1997, pp. 246-250, ISSN 1051-8215

Coebera J. R. & Lei S. (1999). Rate control of DCT video coding for low-delay communication, *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 9, No. 1, February 1999, pp. 172-185, ISSN 1051-8215

Chen Z.; Zhou P. & He Y. (2002) Fast Integer Pel and Fractional Pel Motion Estimation for JVT, *Joint Video Team (JVT) Docs*, JVT-F017, Dec. 2002.

Erol B.; Gallant M.; Cote G. & Kossentini F. (1998). The H.263+ video coding standard: complexity and performance, *Proceedings of IEEE Data Compression Conference*, pp. 259-268, Snowbird, Utah, March 1998.

Feng Pan F.; Lin X.; Rahardja S.; Lim K. P.; Li Z.G.; Wu D. & Wu S. (2005). Fast Mode Decision Algorithm for Intra-prediction in H.264/AVC Video Coding, *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 15, no. 7, July 2005, pp. 813-822, ISSN 1051-8215

Han K. H. & Lee Y. L. (2005). Fast macroblock mode determination to reduce H.264 complexity, *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. E88-A, No. 3, March 2005, pp. 800-804, ISSN 1745-1337

ITU-T Rec. (2002). Advanced Video Coding, ITU-T Rec. H.264/ISO/IEC 11496-10 Final Committee Draft, Documents JVT-E022, September 2002.

Iain E. G. Richardson (2003) *H.264 and MPEG-4 Video Compression – video coding for next generation multimedia*, John Wily & Sons Ltd, ISBN: 0-470-84837-5, West Sussex, England

Joint Video Team (JVT) (2002). Joint Video Specification (ITU-T Rec. H.264/ISO/ISE 14496-10)- Joint Committee Draft, Doc. JVT-G050r1.doc, Dec. 2002.

Kim C.; Shih H.H. & Kuo C. C (2006). Fast H.264 Intra-prediction mode selection using joint spatial and transform domain features, *Journal of Visual Communication and Image Representation*, vol. 17, No. 2, April 2006, pp. 291-310, ISSN 1047-3203.

Lim K. P; Wu S.; Wu D. J; Rahardja S.; Lin X.; Pan F. & Li Z. G. (2003). Fast inter mode selection, *Joint Video Team (JVT) Docs*, JVT-I020, Sep. 2003.

Sullivan G. J. & Wiegand T. (1998). Rate-distortion optimization for video compression, *IEEE Signal Processing Magazine*, vol. 15, No. 6, November 1998, pp. 74-90, ISSN: 1053-5888

Sarwer M. G.; Po L. M. & Wu J. (2008) Fast Sum of Absolute Transformed Difference based 4x4 Intra Mode Decision of H.264/AVC Video Coding Standard, *Elsevier Journal of Signal Processing: Image Communications*, Vol. 23, Issue 8, September 2008, pp. 571-580, ISSN: 0923-5965.

Sarwer M. G. & Wu J. Q. M (2009) Region Based Searching for Early Terminated Motion Estimation Algorithm of H.264/AVC Video Coding Standard, *Proceedings of IEEE Canadian Conference on Electrical and Computer Engineering*, pp. 468-471. St John's, NL, May 2009

Topiwala. P.; Sullivan G.; Joch A. & Kossentini F. (2001). Performance evaluation of H.26L TML 8 vs. H.263++ and MPEG4, *Document VCEG-042, 15th Meeting, ITU-T Q.6/SG16*, Pattaya, Thailand, December 2001.

Tseng C. H.; Wang H.M. & Yang J. F. (2006), Enhanced Intra 4x4 Mode Decision for H.264/AVC Coders, *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 16, no. 8, August 2006, pp. 1027-1032, ISSN 1051-8215

Weigand T.; Schwarz H.; Joch A.; Kossentini F. & Sullivan G. (2003). Rate-constrained coder control and comparison of video coding standards, *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 13, No. 7, July 2003, pp. 688-703, ISSN 1051-8215

Wu D.; Pan F.; Lim K. P.; Wu S.; Li Z. G.;Lin X.; Rahardja S. & Ko C. C. (2005). Fast inter mode Decision in H.264/AVC Video Coding, *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 15, No. 6, July 2005, pp. 953-958, ISSN 1051-8215

Yang C. L.; Po L. M. & Lam W. H. (2004). A fast H.264 Intra Prediction algorithm using Macroblock Properties, *Proceedings of IEEE International Conference on Image Processing*, pp. 461-464, Singapore, October 2004.

Yu K.; Yang J. F. &Sun M. T. (2006) Efficient rate-distortion estimation for H.264/AVC coders, *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 16, no. 5, May 2006, pp. 600-611, ISSN 1051-8215