

# Fast Bit Rate Estimation for Mode Decision of H.264/AVC

Mohammed Golam Sarwer and Lai-Man Po, *Member, IEEE*

**Abstract**—To achieve the highest coding efficiency, H.264/AVC uses rate-distortion optimization technique. This means that the encoder has to code the video by exhaustively trying all the mode combinations including the different intra- and inter-prediction modes. Therefore, the complexity and computation load of video coding in H.264/AVC increase drastically compared to any previous standards. To reduce the complexity of rate-distortion cost computation, we propose a fast bit rate estimation technique to avoid the entropy coding method during intra- and inter-mode decision of H.264/AVC. The estimation method is based on the properties of context-based variable length coding (CAVLC). The proposed rate model predicts the rate of a  $4 \times 4$  quantized residual block using five different tokens of CAVLC. Experimental results demonstrate that the proposed estimation method reduces about 47% of total encoding time on using intra-modes only and saves about 34% of total encoding time on using both inter- and intra-modes with ignorable degradation of coding performance when the fast motion search algorithm is used. When full search motion estimation algorithm is used, the proposed algorithm reduces about 17% of total encoding time.

**Index Terms**—Context-based variable length coding (CAVLC), H.264/AVC, mode decision, rate estimation, rate-distortion (RD) optimization.

## I. INTRODUCTION

THE ITU-T/ISO/IEC joint video team established the newest video coding standard known as H.264/AVC [1]. H.264/AVC offers a significant performance improvement over previous video coding standards such as H.263++ and MPEG-4 part 2 [2], [3]. H.264/AVC employs the rate-distortion (RD) optimization technique to get the best result of the visual quality and bit rate. But the computation complexity of mode decision algorithm is extremely high. To reduce computational complexity of H.264/AVC, a number of efforts have been made to explore the fast algorithm in motion estimation, intra-, and inter-mode prediction [4]–[6], [15]–[18]. In [13] and [14], rate models were observed from the quantizer ( $Q$ )-domain. But these are considered only for rate control. To improve the RD performance, a new cost function for intra  $4 \times 4$  mode decisions was proposed in [9]. The major drawback of this cost is that the bit estimation method cannot give the very good estimation. 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 [12].

In this paper, we propose a shortcut way to get the number of entropy coded bits as soon as the transform coefficients are

quantized. The total number of bits need to encode a quantized residual block is predicted by estimating the rate of each symbols of context-based variable length coding (CAVLC) separately.

The remainder of this paper is organized as follows. Section II provides RD optimized mode decision technique. In Section III, we present the proposed fast bit rate estimation method. The simulation results of the proposed method are presented in Section IV. Finally, Section V concludes the paper.

## II. RD OPTIMIZED MODE DECISION

To take the full advantages of all modes, the H.264/AVC encoder can determine the mode that meets the best RD tradeoff using RD optimization mode decision scheme. The best mode for every block that produces the minimum RD cost is given by

$$J_{RD} = SSD + \lambda_m(R_{\text{header}} + R_{\text{motion}} + R_{\text{res}}) \quad (1)$$

where SSD is the sum of squared difference between original block and reconstructed block,  $\lambda_m$  is the Lagrangian multiplier, and  $m$  is the candidate mode. A strong connection between the local Lagrangian multiplier and the quantization parameter (QP) was found in [8], [19].  $R_{\text{header}}$ ,  $R_{\text{motion}}$ , and  $R_{\text{res}}$  mean the number of bits need to encode the header information, motion vectors, and quantized residual block, respectively.

## III. 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.

### A. Coefficient Token (Number of Coefficients, Number of Trailing Ones)

Four variable length coding (VLC) tables are used for encoding coefficient token [10]. Fig. 1(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 [11], it is also shown that bit rate for coefficient token is decreased with number of trailing ones. Based on this criteria, we propose that the number of bits require to encode the coefficient token is

$$R_{\text{coeff}} = w_1 T_c - w_2 T_o + w_3 \quad (2)$$

where  $T_c$  is the total number of nonzero coefficients and  $T_o$  is the number of trailing  $\pm 1$  values. 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 RD

Manuscript received September 16, 2006; revised March 4, 2007. This work was supported by the City University of Hong Kong, Hong Kong SAR, under Project 7001853. This paper was recommended by Associate Editor T. Nguyen.

The authors are with the Department of Electronic Engineering, City University of Hong Kong, Kowloon, Hong Kong (e-mail: sarwer@ee.cityu.edu.hk).

Digital Object Identifier 10.1109/TCSVT.2007.903787

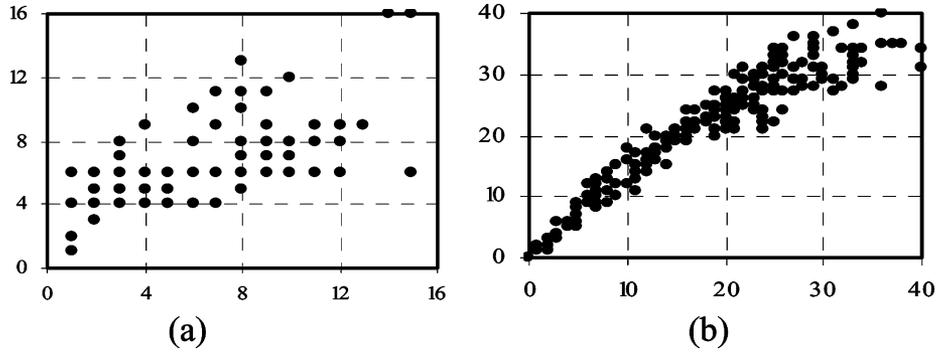


Fig. 1. (a) Number of nonzero coefficients versus true value of coefficient token ( $X$ -axis:  $T_c$ ,  $Y$ -axis: true rate of Coeff\_token). (b)  $SAT_l$  versus actual rate of level ( $X$ -axis:  $SAT_l$ ,  $Y$ -axis: true rate of level).

performance of these video sequences at different combinations of weighting factor. Better RD performance was found at  $w_1 = w_2 = 1$  and  $w_3 = 0$ .

### B. Sign of Trailing Ones

For each  $T_o$ , a single bit encodes the sign ( $0 = +$ ,  $1 = -$ ). Thus, bit consumption to encode the trailing ones is as follows:

$$R_{\text{trailing}} = T_o. \quad (3)$$

### C. Level of Nonzero Coefficients

From the observation of level-VLC tables [11], it is shown that bit requirement is increased with magnitude of nonzero coefficients. Number of bits to encode the level information is proposed as follows:

$$R_{\text{level}} = w_4 SAT_l \quad (4)$$

with the  $SAT_l$  given by

$$SAT_l = \sum_{k=1}^{TC} |L_k| \quad (5)$$

where  $|L_k|$  is the absolute value of  $k$ th nonzero coefficient and  $SAT_l$  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 the 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  $R(L_1)$  is also greater than that of second coefficient  $R(L_2)$ . Therefore, (4) is valid. Let us consider two coefficients being encoded using multiple level\_VLC tables. If the earlier coefficient is not larger than the other coefficient, by observing the all level\_VLC tables [11], there is  $R(L_1) > R(L_2)$  if  $|L_1| > |L_2|$ . Sometimes,  $R(L_1) > R(L_2)$  if  $|L_1| < |L_2|$  but fortunately this event does not occur frequently and slightly influences the estimation result. Fig. 1(b) shows the plot of actual bit rate to encode the level information with sum of absolute values of levels ( $SAT_l$ ) 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  $w_4 = 1$ .

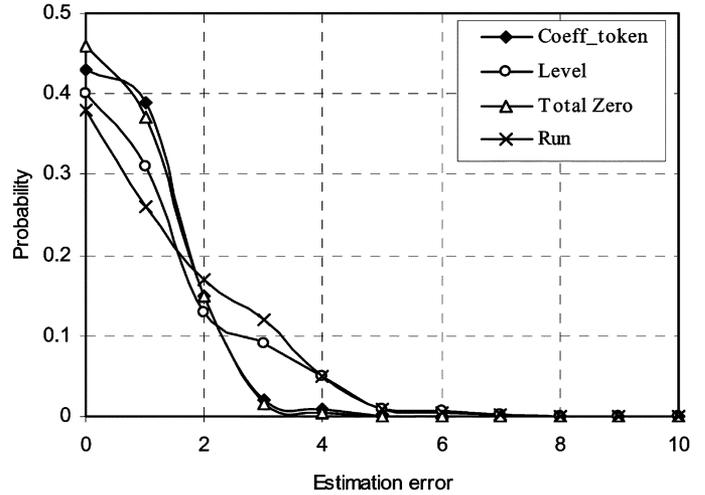


Fig. 2. Probability of estimation error of four different token.

### D. Encoding the Total Number of Zeros Before the Last Coefficient

From the observation of total zero VLC tables, it is shown that bit consumption to encode the total zero is increased with number of total zero. So we can propose the estimated bits for total zero as follows:

$$R_{\text{zero}} = w_5 T_z \quad (6)$$

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

### E. Encode Each Run of Zeros

After the discrete-cosine transform (DCT), 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 nonzero 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 nonzero coefficients. Based on this idea, we propose the rate for run of each nonzero coefficients as follow:

$$R_{\text{run}(k)} = w_6 f_k, \quad 0 \leq k \leq 15 \quad (7)$$

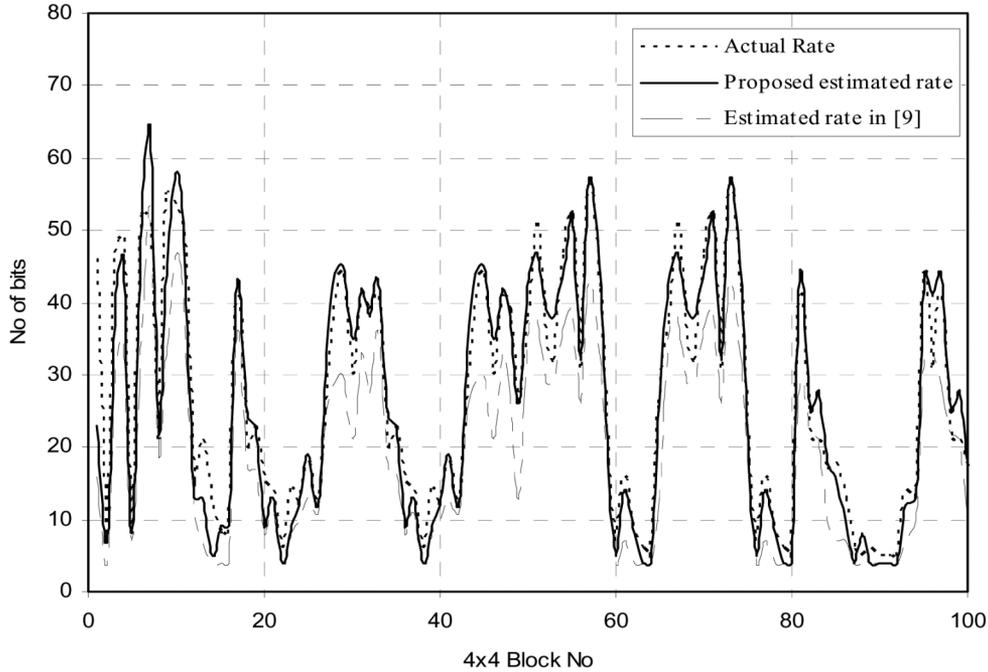


Fig. 3. Comparison of our proposed method with rate estimation method described in [9].

where  $f_k$  is the frequency of  $k$ th nonzero coefficient of recorded block and  $w_6$  is the positive constant. For example, given a string of coefficients  $[0, 3, 0, 1, -1, -1, 0, 1, 0, 0, \dots]$ , frequency of first nonzero coefficient (3) is 1 and frequency of last nonzero coefficient (1) is 7. The weighting factor  $w_6 = 0.3$  is found in similar way as  $w_4$ .

From previous analysis, we have estimated the bits needed to encode a  $4 \times 4$  residual block ( $R_{\text{est}(\text{res})}$ ) are

$$R_{\text{est}(\text{res})} = R_{\text{coeff}} + R_{\text{trail1}} + R_{\text{level}} + R_{\text{zero}} + \sum_{k=1}^{T_c} R_{\text{run}(k)}.$$

By putting (2)–(7) and the values of different constants in the previous equation, the proposed rate estimator becomes

$$R_{\text{est}(\text{res})} = T_c + T_z + \text{SAT}_l + 0.3 \sum_{k=1}^{T_c} f_k. \quad (8)$$

Fig. 2 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. It is shown that most of the estimation errors of each symbol are between 0 and 4 and the probability of each symbol having estimated rate perfectly matching with CAVLC is about 40%.

#### IV. EXPERIMENTAL RESULTS

To verify the proposed technique, JM 8.3 [7] reference software is used in simulation. The test conditions are as follows: 1) CAVLC is enabled, 2) frame rate is 30, 3) motion vector (MV) search range is  $\pm 32$ , and 4) number of frame is 100. The results are performed on a Pentium IV 2.8-GHz personal computer with 1-GB random access memory (RAM) and Microsoft Windows XP as the operating system. Comparison results were produced

based on the percentage of difference of coding time ( $\Delta T\%$ ), the peak-signal-to-noise ratio (PSNR) difference ( $\Delta P_{\text{snr}}$ ), and percentage of the bit rate difference ( $\Delta \text{Bit}\%$ ). In order to evaluate complexity reduction,  $\Delta T(\%)$  is defined as follows:

$$\Delta T = \frac{T_{\text{original}} - T_{\text{proposed}}}{T_{\text{original}}} \times 100\% \quad (9)$$

where  $T_{\text{original}}$  denotes the total encoding time of the JM 8.3 encoder with RD optimization and  $T_{\text{proposed}}$  being the total encoding time with proposed fast rate estimation technique.

##### A. Experiments on All Intra-Frame Sequence

In this experiment, all frames are  $I$  frame. Fig. 3 shows the comparison of our proposed method with actual rate and the rate predictor described in [9] for the Foreman sequence. It is shown that the proposed method is very closely matched with actual rate as compared to rate predictor in [9]. As shown in Table I, it is clear that PSNR loss and bit rate increment is negligible when all frames are  $I$  frames. In Table I, positive values mean increments whereas negative values mean decrements. 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. It is also shown that, 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.” Table I also indicates that percentage of complexity reduction is decreased with increasing QP values. Because, 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. 4 shows the RD curves of different sequences. The proposed method is very close with RD optimized curve.

TABLE I  
EXPERIMENTAL RESULTS OF PROPOSED METHOD

Sequence	QP	All I frame			IPP sequence			IBPBP sequence		
		$\Delta P_{snr}$	$\Delta Bit\%$	$\Delta T\%$	$\Delta P_{snr}$	$\Delta Bit\%$	$\Delta T\%$	$\Delta P_{snr}$	$\Delta Bit\%$	$\Delta T\%$
Akiyo (QCIF)	20	-0.06	+0.64	51.21	+0.01	+1.47	36.63	+0.09	+3.84	30.41
	28	+0.02	+0.53	44.11	+0.09	+1.97	28.73	+0.14	+3.00	20.84
	36	+0.12	+1.49	35.35	+0.11	+3.00	22.07	+0.05	+2.79	16.33
Foreman (QCIF)	20	-0.06	+0.49	55.20	+0.02	+2.12	41.66	+0.09	+3.24	36.86
	28	-0.02	+1.16	46.66	+0.05	+2.57	33.33	+0.11	+3.96	29.60
	36	+0.11	+2.35	37.28	+0.13	+3.05	27.58	+0.04	+2.56	20.05
Mobile (QCIF)	20	-0.16	+0.85	62.87	-0.05	+0.95	53.08	+0.01	+3.04	46.26
	28	-0.10	+0.54	59.45	-0.03	+1.17	46.71	+0.11	+2.82	39.67
	36	-0.06	+0.70	51.21	+0.02	+2.89	34.57	+0.11	+2.80	28.45
Paris (CIF)	20	-0.16	+0.46	58.11	-0.05	+1.17	45.15	+0.05	+1.78	39.46
	28	-0.10	+0.45	52.81	-0.02	+1.43	38.39	+0.07	+2.79	31.44
	36	-0.02	+0.64	43.30	+0.02	+2.33	28.64	+0.11	+4.96	20.64
Table Tennis (CIF)	20	+0.31	+0.31	57.04	-0.03	+2.20	41.61	+0.07	+2.81	36.86
	28	-0.04	+1.01	49.54	+0.04	+2.52	32.45	+0.11	+4.38	25.10
	36	+0.02	+1.48	39.11	+0.06	+2.95	25.94	+0.09	+3.96	25.82
Stefan (CIF)	20	-0.19	+0.62	57.41	-0.10	+1.17	46.19	+0.03	+1.03	38.55
	28	-0.11	+0.42	50.55	-0.03	+1.97	38.61	+0.14	+4.38	33.82
	36	-0.02	+1.69	39.20	+0.07	+3.65	30.56	+0.14	+5.03	25.95

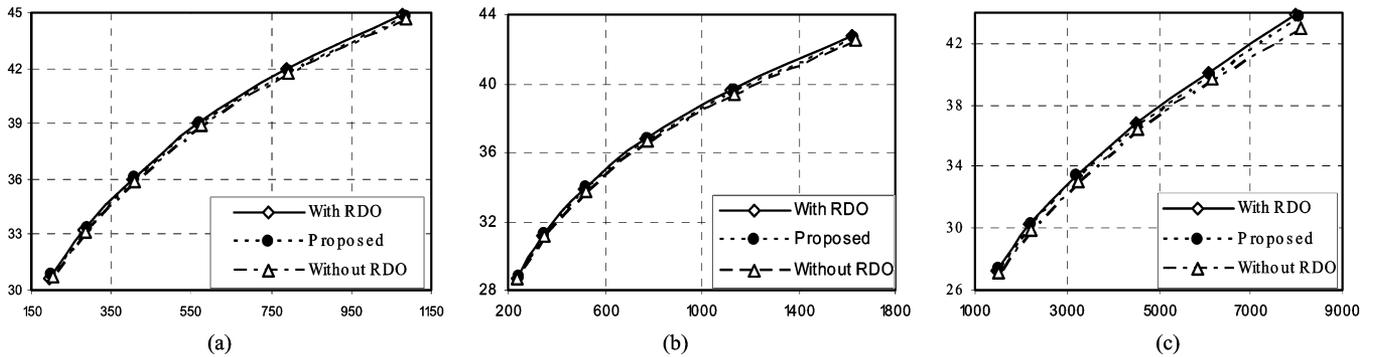


Fig. 4. RD performance of proposed rate estimation method of different video sequences while all frames are intra-coded ( $X$ -axis: bit rate in kilobits per second,  $Y$ -axis: PSNR in decibels). (a) Akiyo (QCIF). (b) Foreman (QCIF). (c) Stefan (CIF).

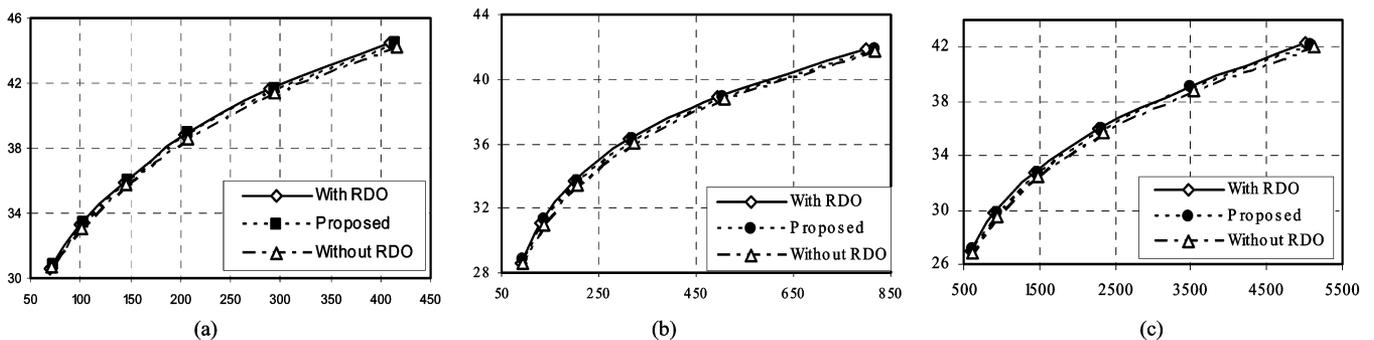


Fig. 5. RD performance of proposed rate estimation method of different video sequences during inter frame (IPP sequence) coding ( $X$ -axis: Bit rate in kilobits per second,  $Y$ -axis: PSNR in decibels). (a) Akiyo (QCIF). (b) Foreman (QCIF). (c) Stefan (CIF).

### B. Experiments on IPP Sequence

In this experiment, the period of  $I$  frame is 3. Fast motion estimation algorithm [15] is used. Experimental results are tabulated in Table I which means that PSNR reduction and bit rate increment are negligible. From the experimental results, it is observed that the proposed approach has reduced the encoding time by 34% on average. As shown in Fig. 5, RD performance of a proposed method is very close to the actual RD curves.

### C. Experiments on IBPBP Sequence

In this experiment, there is one  $B$  frame between any two  $I$  or  $P$  frames. Fast motion estimation algorithm [15] is used. Experimental results were tabulated in Table I. As shown in Table I, the proposed algorithm achieves the time saving of about 32% (on average) with slight increment in bit rate. RD performances of different sequences are shown in Fig. 6.

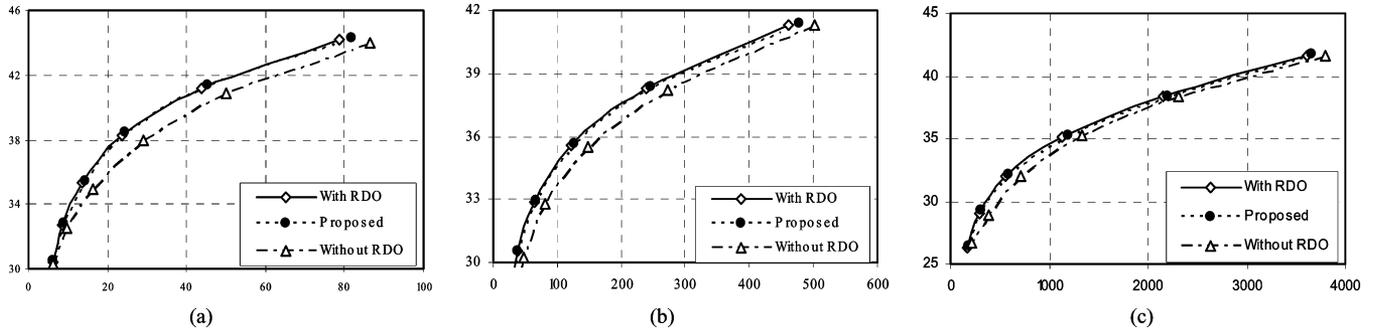


Fig. 6. RD performance of proposed method with IBPBP sequences ( $X$ -axis: Bit rate in kilobits per second,  $Y$ -axis: PSNR in decibels). (a) Akiyo (QCIF). (b) Foreman (QCIF). (c) Stefan (CIF).

TABLE II  
EXPERIMENTAL RESULTS WITH FULL SEARCH MOTION ESTIMATION

Sequence	QP	$\Delta P_{snr}$	$\Delta \text{Bit}\%$	$\Delta T\%$
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 III  
COMPARISON OF PROPOSED METHOD WITH RATE ESTIMATION METHOD IN [9]

Sequence	QP	$\Delta P_{snr}$	$\Delta \text{Bit}\%$	$\Delta T\%$
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

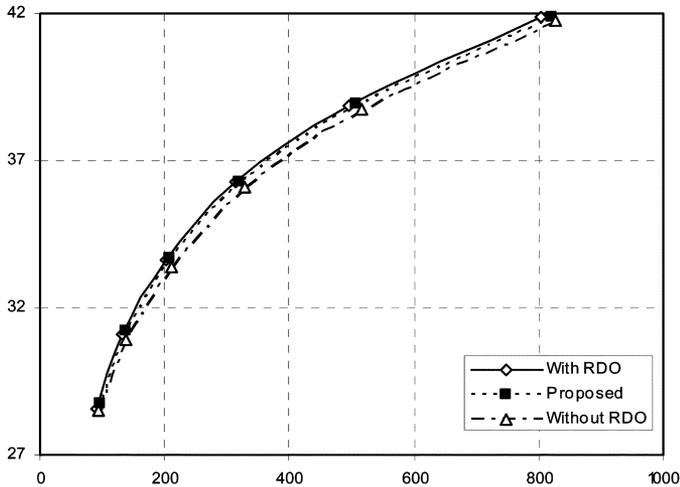


Fig. 7. RD performance of foreman with full search motion estimation ( $X$ -axis: Bit rate in kilobits per second,  $Y$ -axis: PSNR in decibels).

#### D. Experiments With Full Search Motion Estimation

It is well known that motion estimation requires major portion of the processing power. In order to show the complexity of the 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 in Table II. It is shown that bit rate increment is up to about 4%. Since the PSNR also increases, the resulting RD performance is very much close to the original one. The proposed algorithm reduces about 17% (on average) of computation time when full search motion estimation method is used. RD performance of foreman sequence is given in Fig. 7.

#### E. Comparison With Other Method

In this experiment, the proposed method is compared with rate estimation method defined in [9] in terms of RD performance and complexity. Only intra  $4 \times 4$  modes are used. Table III shows the comparison results. Negative value of  $\Delta T\%$  means increment of complexity of our method. As compared with [9], this algorithm reduces about 1% of bit rate and increases about 0.10-dB PSNR with slight increment (about 4%) of computational time.

#### V. CONCLUSION

In this paper, 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 result 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. When fast search motion estimation is used, the proposed technique reduces encoding time by 47%, 34%, and 32% on average during intra-frame, IPP sequences, and IBPBP sequences, respectively. When full search motion estimation is utilized, about 17% of encoding time can be reduced.

#### REFERENCES

- [1] "Joint video specification (ITU-T Rec. H.264/ISO/IEC 14496-10)—Joint committee draft," Joint Video Team(JVT) of ISO/IEC MPEG and ITU-T VCEG, 2002, Doc. JVT-G050r1.doc.
- [2] P. Topiwala, G. Sullivan, A. Joch, and F. Kossentini, "Performance evaluation of H.26L TML 8 vs. H.263++ and MPEG4," in *Proc. 15th Meeting, ITU-T Q.6/SG16*, Pattaya, Thailand, Dec. 2001.

- [3] B. Erol, M. Gallant, G. Cote, and F. Kossentini, "The H.263+ video coding standard: Complexity and performance," in *Proc. IEEE Data Compression Conf.*, Snowbird, UT, Mar. 1998, pp. 259–268.
- [4] C. Kim, H.-H. Shih, and C.-C. Kuo, "Fast H.264 intra-prediction mode selection using joint spatial and transform domain features," *J. Vis. Commun. Image Represent.*, vol. 17, pp. 291–310, 2006.
- [5] C.-L. Yang, L. M. Po, and W.-H. Lam, "A fast H.264 intra prediction algorithm using macroblock properties," in *Proc. Int. Conf. Image Process (ICIP)*, 2004, pp. 461–464.
- [6] F. Pan, X. Lin, S. Rahardja, K. P. Lim, Z. G. Li, D. Wu, and S. Wu, "Fast mode decision algorithm for intraprediction in H.264/AVC video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 15, no. 7, pp. 813–822, Jul. 2005.
- [7] "Joint video team (JVT) reference software version 8.3," [Online]. Available: [http://iphome.hhi.de/suehring/tml/download/old\\_jm/](http://iphome.hhi.de/suehring/tml/download/old_jm/)
- [8] G. J. Sullivan and T. Wiegand, "Rate-distortion optimization for video compression," *IEEE Signal Process. Mag.*, vol. 15, no. 6, pp. 74–90, Nov. 1998.
- [9] C. H. Tseng, H. M. Wang, and J. F. Yang, "Enhanced intra  $4 \times 4$  mode decision for H.264/AVC coders," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 16, no. 8, pp. 1027–1032, Aug. 2006.
- [10] I. E. G. Richardson, *H.264 and MPEG-4 Video Compression—Video Coding for Next Generation Multimedia*. New York: Wiley, 2003, pp. 198–207.
- [11] "Advanced video coding," 2002, ITU-T Rec. H.264/ISO/IEC 11496-10, Final Committee Draft, Documents JVT-E022.
- [12] Yu-Kuang, J.-F. Yang, and M.-T. Sun, "Efficient rate-distortion estimation for H.264/AVC coders," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 16, no. 5, pp. 600–611, May 2006.
- [13] T. Chiang and Y. Q. Zhang, "A new rate control scheme using quadratic rate distortion model," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, no. 1, pp. 246–250, Feb. 1997.
- [14] J. Ribas-Coebera and S. Lei, "Rate control of DCT video coding for low-delay communication," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, no. 1, pp. 172–185, Feb. 1999.
- [15] Z. Chen, P. Zhou, and Y. He, "Fast integer pel and fractional pel motion estimation for JVT," *Joint Video Team (JVT) Docs, JVT-F017*, Dec. 2002.
- [16] K. P. Lim, S. Wu, D. J. Wu, S. Rahardja, X. Lin, F. Pan, and Z. G. Li, "Fast inter mode selection," *Joint Video Team (JVT) Docs, JVT-I020*, Sep. 2003.
- [17] D. Wu, F. Pan, K. P. Lim, S. Wu, Z. G. Li, X. Lin, S. Rahardja, and C. C. Ko, "Fast inter mode decision in H.264/AVC video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 15, no. 6, pp. 953–958, Jul. 2005.
- [18] K. H. Han and Y. L. Lee, "Fast macroblock mode determination to reduce H.264 complexity," *IEICE Trans. Fundam.*, vol. E88-A, no. 3, pp. 800–804, Mar. 2005.
- [19] T. Wiegand, H. Schwarz, A. Joch, F. Kossentini, and G. Sullivan, "Rate-constrained coder control and comparison of video coding standards," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 688–703, Jul. 2003.