EFFICIENT 16X16 BLOCK SIZE MODE DETECTION IN H.264

Youngsub Jo, Wonkyun Kim, Joohyeok Kim, and Jechang Jeong

Department of Electronics and Computer Engineering, Hanyang University
17 Haengdang-dong, Seongdong-gu, Seoul, Korea
longglittz@naver.com, wonkyun@ece.hanyang.ac.kr, kjh76363@hanyang.ac.kr, jjeong@ece.hanyang.ac.kr

ABSTRACT

In this paper, we propose a fast algorithm that can reduce the complexity for inter mode decision of the H.264 encoder by minimizing a large amount of calculation of inter mode decision process adaptively. We focus mainly on extracting the large block size mode, which is 16x16 mode, efficiently. From examination of various sequences, we know that the percentage of large block size mode which selected in best mode is highest of all possible modes. Therefore the efficient filter to extract large block size mode early is proposed in this paper. When the proposed algorithm is adopted in H.264, encoder is shown to have good performance in terms of time saving. The experimental results show that the proposed algorithm can achieve up to 39% speed up ratio with a little PSNR loss. Increase of total bits encoded is also not much noticeable.

1. INTRODUCTION

The improvement in coding efficiency of H.264 is due to multiple reference pictures, quarter-pel accurate motion vector search, variable block size to find out accurate motion vectors, coding for intra mode decision with various prediction directions, inter mode decision using different block size and adaptive deblocking filter, and so on. The most complex and computational burden is given from mode decision process out of many features of H.264.

For inter mode selection, one macroblock (MB) is divided into several block sizes including subblock sizes. These are 16x16, 16x8, 8x16, 8x8, 8x4, 4x8 and 4x4. And for intra mode selection, 4x4 and 16x16 block sizes are used to support 9 and 4 different directional predictions. To predict the best block size for mode decision, all block sizes must be checked. Rate-distortion optimization technique (RDO) is chosen in Joint Model (JM) reference encoder as the most accurate and powerful way. Using this RDO technique, the best mode in mode selection process is determined through the minimization of rate-distortion cost (RDC). RDC is calculated from RDC function [1]. However this technique requires a huge amount of computational complexity.

Up to date, a number of algorithms [6]-[10] have been proposed to reduce the computational complexity of mode decision process using RDO technique. Reference [2] proposed homogeneity detection using edge map and temporal similarity detection using SAD for early termination. Edge map using edge information is created for each frame in [3] with Sobel operator. This technique is effective but time reduction ratio is not considerable, because intra mode selection is executed by homogeneous region detector in every MB. If intra mode selection is not used in every MB, time saving ratio may be larger. Byung-Gyu Kim [4] achieved fast prediction performance using two ways. The one is a simple MB tracking strategy that uses motion vector and correlation ratio between current MB and predicted MB to check temporal similarity. The other is the binary pattern examination to determine homogeneity of 8x8 block size using average pixel intensity. To do early termination, threshold value must be provided in this algorithm. But it is not easy to select this threshold value for better performance. Bin Zhan et al. [5] proposed temporal-spatial correlation algorithm for fast inter mode decision. They made use of best mode information for neighbouring blocks of colocated MB in reference frame. And they also used motion vector of 16x16 block size. Using above two ways, they minimize the number of candidate mode list which must be calculated for RDC of current MB. When this algorithm is applied to H.264 reference software, the result is very considerable in terms of PSNR and bit rate. But the time reduction is not so significant when compared to other algorithms.

In this paper, we have developed an efficient large block size mode detection algorithm adopting three simple and computationally easy steps. In first step, the threshold value to detect 16x16 block size mode, which is SKIP and 16x16 mode, is made from RDC of best mode for neighbouring blocks of current MB. In second step, we examine RDC for all neighbouring blocks of colocated MB from previous frame. Finally, we make use of relation among SKIP, 16x16 and 8x16 block size mode. Experimental results show that total encoding time can be reduced by 39% on average with negligible decrease for performance of encoder.

The organization of this paper is as follows: In section 2, three steps to detect large block size mode are described sequentially based on exhaustive examination. Section 3 presents experimental results. Finally, conclusion is shown in section 4.

2. PROPOSED FAST ALGORITHM

2.1 Motivation

In general, the blocks having less motion information are encoded as *SKIP mode* (MD0) or *16x16 block size mode* (MD1) and blocks having more motion information are encoded in smaller size modes like *16x8 block size mode*

(MD2), *8x16 block size mode* (MD3) and *8x8 block size mode* (MD8). As you can see from Table 1, the motionless sequences such as Akiyo or Container have a large percentage for MD0 and MD1 as a best mode. And also the sequences including more motion information like Foreman, Coastguard, Stefan and Tempete have average percentage over 50%. Consequently, if we can detect large block size mode like MD0 and MD1 well, computational complexity of mode decision process can be reduced efficiently with proposed algorithm. To detect *large block size mode* (MD0 and MD1) well, we propose three level mode detector. The details are shown in section 2.2-2.5.

2.2 MD0 and MD1 detection with spatial information (First step)

In first step, we examine the RDC information for best mode of neighbouring blocks of current frame. We only use the spatially nearest four blocks (left, left-up, up, and up-right) from current block. These four neighbouring blocks are already encoded blocks and we know their each best mode information and RDC value of best mode. Using these four blocks we predict threshold RDC value to detect MD0 and MD1 in first step.

The position within current frame of four neighbouring MBs which is utilized in first step is shown in Fig. 1 with a block diagram. As you can see from Fig. 1, the MBs placed on boundary of current frame are not target MBs for the early detection of MD0 and MD1. These MBs don't have enough information to detect MD0 and MD1 of the current MB. If the boundary MBs are used in first step, performance degradation with negligible speed-up may be shown in some sequences. Therefore we don't consider early detection in boundary MBs to avoid performance decrease.

The condition for detecting MD0 and MD1 in first step is shown in (1). The RDC value of neighbouring block is used in (1). Neighbouring blocks having best mode of MD0 or MD1 are only available in the calculation of (1). At first, we find RDC_L and RDC_H. RDC_L means the average RDC value of the blocks which best mode is MD0 or MD1. RDC_H means the average RDC value of the blocks which best mode is not MD0 and MD1. Second, we compare RDC_L with RDC_H.

After comparison between RDC_L and RDC_H, if RDC_L is smaller than RDC_H we compare RDC_L with both RDC_0 and RDC_1 respectively again. RDC_0 means the RDC value of MD0 for current MB, and RDC_1 means the RDC value of MD1 for current MB. In this comparison step, our aim is to find RDC_0 or RDC_1 which is smaller than RDC_L.

If above condition is satisfied, we ultimately can finish the mode decision process through early termination. In this early termination case, if RDC_0 is smaller than RDC_1, MD0 is final best mode, or vice versa.

$$\begin{aligned}&\textit{If } \{(RDC_L < RDC_H) \,\&\, (RDC_0 < RDC_L) \,\&\, (RDC_1 < RDC_L)\}\\ &\qquad \textit{Then} \text{ best mode is MD0 or MD1 and early terminated}\\ &\textit{Otherwise} \text{ jump to the next step}\end{aligned} \quad \textbf{(1)}$$

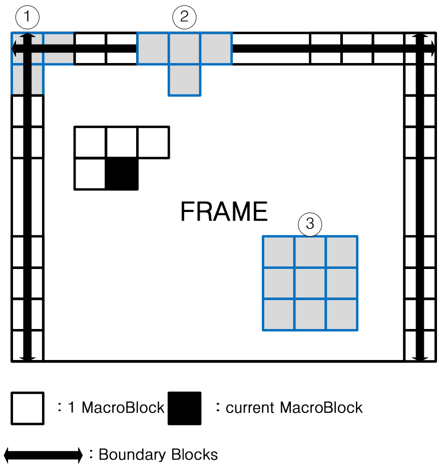

Figure 1. The boundary regions of current frame

Sequences	SKIP + 16x16(%)	16x8 + 8x16(%)	8x8(%)
Foreman	62.8	22.7	12.7
Akiyo	91.2	6.6	2.2
Coastguard	52.8	28.2	18.5
Stefan	55.4	19.7	22.4
Mobile	42.3	23.1	34.3
Container	89.9	6.6	3.0
Tempete	50.8	21.0	25.2

Table 1. Best mode ratio (CIF, QP=28, and 50 frames)

2.3 MD0 and MD1 detection with temporal information (Second step)

In second step, the RDC value of neighbouring blocks around the colocated MB of previous frame is examined to early detect MD0 and MD1 of current MB. Because the RDC value of current MB is highly correlated to that of the previous colocated MB [5].

The best mode of neighbouring blocks for colocated MB is categorized into large block size mode and smaller block size mode. The maximum number of neighbouring blocks is 9 which are placed around the colocated MB such as the gray part of ③ in Fig. 1. In this case, colocated MB must not be in boundary regions. If colocated MB is placed in the boundary region, the number of available neighbouring MB is limited to 4 such as the gray part of ② in Fig. 1. Although the useful information in the boundary region is constricted, it is possible to detect large block size mode with negligible performance degradation. The least number of neighbouring MB is fixed to 3. The 4 corner blocks of previous frame have only 3 neighbouring MBs such as the gray part of ① in Fig. 1. Why we classify MBs in previous frame into different classes is to increase performance of our fast algorithm. In many sequences, the boundary regions are usually motionless, therefore we may need less information for the detection in the boundary region. Using these neighbouring MBs, we calculate (1) to enter early detection process again. If (1) is not satisfied, we can consider third step.

2.4 MD0 and MD1 detection using relation among the modes (Third step)

Although a number of MD0 and MD1 are extracted through first and second step, the significant number of them is still remained. In third step, we propose efficient comparison method to detect MD0 and MD1 using relation among the modes. The target mode about MD0 and MD1 in comparison method is decided by exhaustive examination. To avoid computational complexity, we need to add only one target mode except MD0 and MD1. The candidates of target mode are the three modes(MD2, MD3 and MD8). The specific statistical data are shown in Table 2. Average values of *hit ratio* (HR) and *accuracy* (AR) are used as basis of target mode decision. The Best target mode is MD8 from Table 2. But the computational complexity of MD8 is much higher than MD0 and MD1. Therefore if we use MD8 as the target mode, we can't nearly obtain the encoding gain about time saving. Also the differences of average values about HR and AR of three modes are not big. We can mathematically express above discussion using following equation.

$$m_k = \underset{m=\{m_2, m_3, m_8\}}{\operatorname{argmax}} \operatorname{MP}(HR, AR, T) \quad \mathrm{MP} = \alpha(HR + AR) + (1-\alpha)T \tag{2}$$

In (2), MP means maximum performance. T means encoding time reduction rate and α means the experiential coefficient. We know from our experiment about HR, AR, PSNR and bit rate that the high value of HR and AR means better performance in viewpoint of PSNR and bit rate. Time reduction rate as well as performance for the PSNR and bit rate is very important parameter. Ultimately, we need target mode having much time reduction and better performance. Using this property about three parameters HR, AR and T, we can determine the best target mode among the three candidate modes.

As a result, MD3 was selected as best target mode because the result of (2) for MD3 is better than that of MD2 and MD8. The equation for calculation of HR and AR is shown in (3) and (4).

$$HR = \frac{Num.\ of\ Hit\ MB}{Total\ num.\ of\ MB} \times 100(\%) \tag{3}$$

$$AR = \frac{Num.\ of\ Hit\ MB}{Num.\ of\ Hit\ MB\ +\ Miss\ MB} \times 100(\%) \tag{4}$$

Total number of MB means total number of MD0 and MD1 in sequence. Hit MB means MB which we predict correctly using proposed algorithm. Miss MB means MB which we can't predict correctly. In Miss MB case, original best mode is not MD0 or MD1 but the best mode is predicted to MD0 or MD1 in our prediction. We define relation of MD0, MD1 and MD3 using (5).

$$\textit{If } \{(RDC_0 < RDC_3) \& (RDC_1 < RDC_3)\} \tag{5}$$

Then best mode is MD0 or MD1 and early terminated

Otherwise calculate all remained modes

Sequences	Mode 2		Mode 3		Mode 8	
	HR	AR	HR	AR	HR	AR
Foreman	59.4	84.1	58.4	86.9	60.7	85.9
Akiyo	90.9	97.6	90.8	97.9	93.4	97.5
Coastguard	46.6	81.3	47.1	82.3	48.5	83.8
Stefan	54.8	86.3	54.5	86.7	54.1	87.9
Mobile	36.7	70.5	36.9	71.9	33.5	75.4
Container	92.9	97.5	92.6	97.1	94.1	98.1
News	93.6	96.7	93.7	96.8	93.6	97.4
Average	67.8	87.7	67.8	88.5	68.2	89.4

Table 2. Relation among the modes (QCIF, QP=28, and 50 frames)

2.5 Detailed flow of proposed mode decision algorithm

We used three techniques to detect MD0 and MD1 for early termination of mode decision process. Though we considered only MD0 and MD1 detection, we could obtain much amount of time saving in all encoding process. The whole algorithm is shown in Fig. 2.

In first step, we investigate RDC for neighbouring blocks of current MB. This reduces encoding time using spatial correlation of inter mode decision for large block. In second step, we examine neighbouring MBs' RDC for colocated MB of previous frame. This reduces encoding time using temporal correlation of large block inter mode decision. In third step, we used relation of the RDC value for modes. This

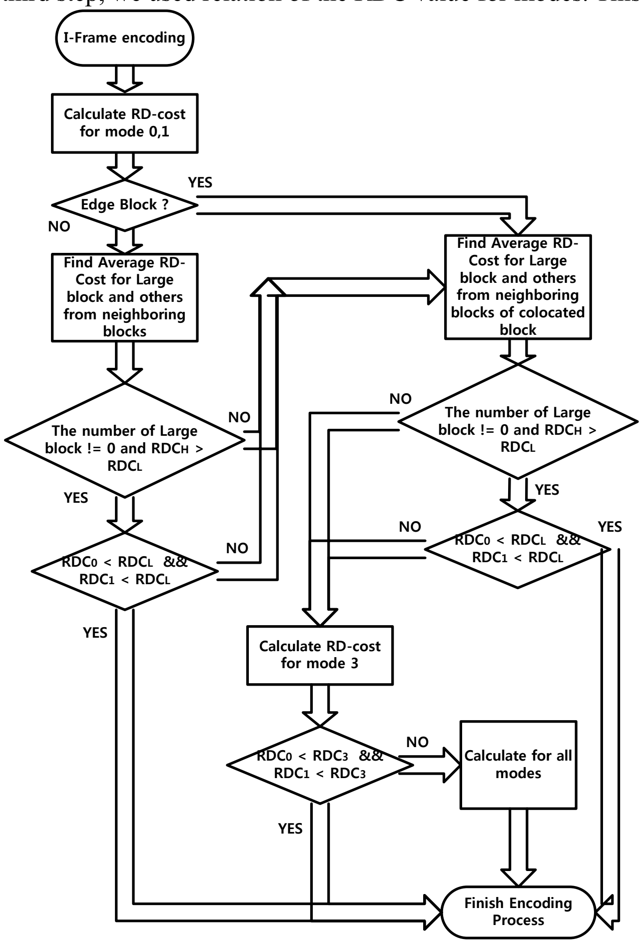

Figure 2. The flow diagram of proposed algorithm

reduces encoding time using statistical correlation of large block modes and MD3.

3. EXPERIMENTAL RESULTS

The proposed fast mode decision algorithm is implemented into JM12.4 [11] encoder with full search motion estimation. We compared our proposed technique with [5]. The results of two algorithms' comparison are presented in the Table 3. And the test conditions are as follows.

1) MV search range is ±16 pels for QCIF.
2) RD optimization is enabled.
3) The number of reference frame is 1.
4) Baseline profile is used.
5) MV resolution is 1/4pel.
6) GOP structure is IPPP…
7) The number of frames in a sequence is 50.
8) Motion estimation scheme is full search.
9) QP value 24, 28, 32, 36 and 40 are used.
10) PC environment for test is based on Windows XP professional, 2GB memory and Intel Pentium D3.2 GHz.

The results of Table 3 are calculated based on three parameters. The first parameter is difference of PSNR (ΔPSNR), the second thing is percentage of bit rate difference (ΔBR) and last thing is percentage of difference for time saving (Δ Time). The comparison target of all three parameters is the result of JM reference software tested on same experimental conditions.

The average gain of proposed algorithm for time saving is about 39%. This result shows that proposed algorithm is more effective than Zhan's algorithm by 13% in view point of time reduction. Nevertheless, proposed algorithm obtained only small time saving in sequence with fast motion such as Mobile. [5] has very outstanding performance in the bit rate decrease for the sequence Akiyo and Container. But bit rate increase in the sequence Susie is considerably high. Averagely, the performance of [5] is very remarkable, but the results of performance of proposed algorithm are better than that of [5]. The PSNR reduction level of proposed algorithm is so small that it is ignorable. The average bit rate change of proposed algorithm even indicates about 0.5% decrease. Although we obtained a little loss in terms of PSNR, 0.5% bit rate decrease of proposed algorithm is significant value for the view point of performance. The RD curve for performance is shown in Fig. 3-5.

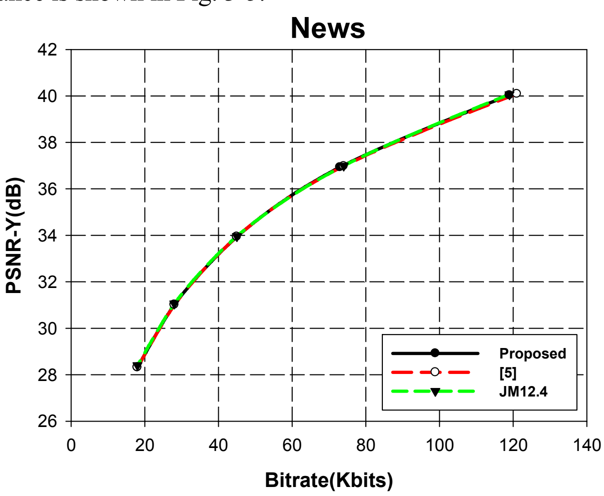

Figure 3. RD curve for News(QP: 24, 28, 32, 36, and 40)

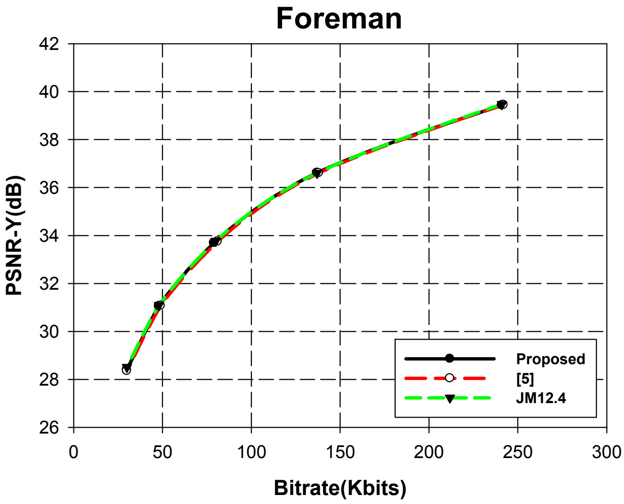

Figure 4. RD curve for Foreman(QP: 24, 28, 32, 36, and 40)

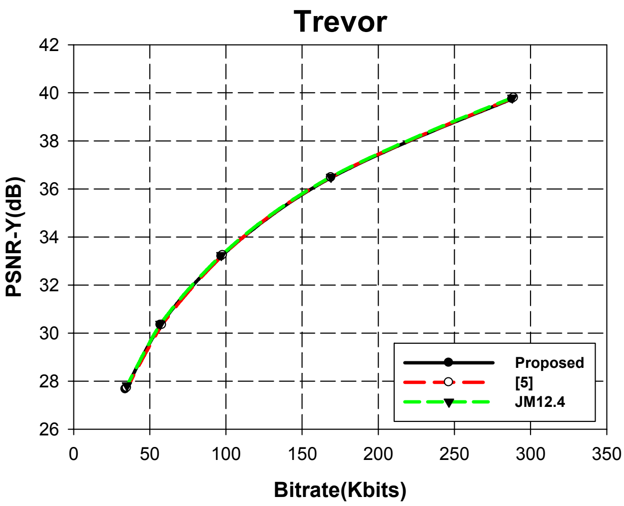

Figure 5. RD curve for Trevor(QP: 24, 28, 32, 36, and 40)

4. CONCLUSIONS

The main purpose of proposed algorithm in this paper is to efficiently detect large block size modes and reduce computational complexity of mode decision process of encoder. Actually in most sequences, the percentage of large block size mode as best mode is the largest. With this reason, we can save much time for encoding process using proposed three detection algorithm. As a result, we obtain time saving gain 39% with ignorable PSNR loss and 0.5% average bit rate reduction.

5. ACKNOWLEDGEMENT

This work was developed within the scope of Human Resource Development Project for IT SoC Architect.

REFERENCES

[1] JVT-K049, Joint model reference encoding methods and decoding concealment methods, ISO/IEC MPEG and ITU-T VCEG Joint Video Team, 2004.

[2] 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. Circuits Syst. Video Technol.*, vol. 15, no. 7, pp. 953-958, Jul. 2005.
[3] F. Pan, X. Lin, R. Susanto, K.-P. Lim, Z.-G. Li, G.-N. Feng, D.-J. Wu, and S. Wu, "Fast mode decision algorithm for intra prediction in JVT," presented at the 7th JVT-G013 Meeting, Pattaya, Thailand, Mar. 2003.
[4] B.-G. Kim, "Novel inter-mode decision algorithm based on macroblock (MB) tracking for the P-slice in H.264/AVC video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 18, no. 2, pp. 273-279, Feb. 2008.
[5] B. Zhan, B. Hou, and R. Sotudeh, "Temporal-spatial correlation based mode decision algorithm for H.264/AVC encoder," in *Proc. 4th IEEE DELTA 2008*, pp. 351-355.
[6] Y.-C. Lin, T. Fink, and E. Bellers, "Fast mode decision for H.264 based on rate-distortion cost estimation," in *Proc. IEEE ICASSP 2007*, pp. 1137-1140.
[7] A.-C.-W. Yu, G.-R. Martin, and H.-C. Park, "Fast inter-mode selection in the H.264/AVC standard using a hierarchical decision process," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 18, no. 2, pp. 186-195, Feb. 2008.
[8] L. Shen, Z. Liu, Z. Zhang, and X. Shi, "Fast inter mode decision using spatial property of motion field," *IEEE Trans. Multimedia*, vol. 10, no. 6, pp. 1208-1214, Oct. 2008.
[9] I.-C. Choi, J.-Y. Lee, and B.-W. Jeon, "Fast coding mode selection with rate-distortion optimization for MPEG-4 part-10 AVC/H.264," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 16, no. 12, pp. 1557-1561, Dec. 2006.
[10] M.-G. Sarwer, and L.-M. Po, "Fast bit rate estimation for mode decision of H.264/AVC," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 10, pp. 1402-1407, Oct. 2007.
[11] JVT reference software version JM 12.4 http://iphome.hhi.de/suehring/tml/download

Sequences	ΔPSNR(dB)		ΔBR(%)		ΔTime(%)	
	Prop.	*[5]*	*Prop.*	*[5]*	*Prop.*	*[5]*
Foreman	0.01	0	-0.03	0.5	**-23.3**	**-15.4**
Akiyo	-0.01	-0.48	-0.59	-2.81	**-61**	**-42.5**
Trevor	-0.03	-0.01	-0.01	0.09	**-25.7**	**-17.3**
News	-0.05	-0.01	-1.3	0.14	**-59.2**	**-27**
Table-tennis	-0.02	0.01	-0.33	0.1	**-31.2**	**-18.8**
Container	-0.07	-0.06	-1.54	-1.02	**-62.8**	**-39.7**
Silent	-0.01	-0.03	-0.56	0.71	**-53**	**-33.9**
Susie	-0.03	-0.01	0.16	1.5	**-38.1**	**-30.7**
Coastguard	-0.01	0.03	-0.01	0.68	**-24.3**	**-17.3**
Mobile	-0.06	0.01	-0.69	0.09	**-11.1**	**-13.6**
Average	-0.03	-0.06	-0.49	-0.001	**-39**	**-25.6**

Table 3. Results of proposed algorithm compared to [5] (QP=28)