Fast Intra-Prediction Mode Decision Algorithm for H.264/AVC using Non-parametric Thresholds and Simplified Directional Masks

Young-ju Kim, *Member, KIMICS*

***Abstract*—**In the H.264/ AVC video coding standard, the intra-prediction coding with various block sizes offers a considerably high improvement in coding efficiency compared to previous standards. In order to achieve this, H.264/AVC uses the Rate-distortion optimization (RDO) technique to select the best intra-prediction mode for a macroblock, and it brings about the drastic increase of the computation complexity of H.264 encoder. To reduce the computation complexity and stabilize the coding performance on visual quality, this paper proposed a fast intra-prediction mode decision algorithm using non-parametric thresholds and simplified directional masks. The use of non-parametric thresholds makes the intra-coding performance not be dependent on types of video sequences and simplified directional masks reduces the compuation loads needed by the calculation of local edge information. Experiment results show that the proposed algorithm is able to reduce more than 55% of the whole encoding time with a negligible loss in PSNR and bitrates and provides the stable performance regardless types of video sequences.

Index Terms— H.264/AVC, Intra prediction mode decision, Non-parametric criteria, Simplified directional masks

I. INTRODUCTION

In the H.264/AVC video coding standard, the intra-prediction coding technique with various block sizes is one of new coding technologies making it possible a significant performance improvement over previous video coding standards such as H.263++ and MPEG-4, etc[1,2,3]. But, in the intra-prediction coding, H.264/AVC uses Rate-Distortion Optimization(RDO) technique to obtain the best result maximizing visual quality and minimizing bitrates. To choose the optimal intra-prediction mode, the standard H.264 encoder uses the full search method which calculates the RDcost(Rate Distortion cost) of every possible mode and chooses the mode having the minimum cost. Thus, the computation complexity is extremely increased compared to previous standards, so it makes H.264/ AVC difficult for applications with low computational capability, such as mobile devices[2,3,4].

To reduce the computation complexity of the intra-prediction coding in H.264/AVC, a variety of fast intra-prediction coding algorithms have been proposed for the last few years [5~11]. These algorithms mostly use the empirical thresholds being to be determined by experiments to decide the probable block size or intra-prediction mode, and accordingly the performance of these algorithms would be dependent on the types of video sequences. This is, the parametric algorithms may generate low performance on visual quality or bitrates for some types of video sequences. And in the calculation of local edge information applying directional masks to internal pixels of macroblocks, previous algorithms didn't consider boundary pixels of reconstructed neighboring blocks being used in the intra-prediction coding phase and also needed relatively high computation overheads.

This paper proposes a novel fast intra-prediction mode decision algorithm based on the simplified calculation of local edge information and the non-parametric decision method in H.264/AVC. The proposed algorithm performs the intra-prediction mode decision through two steps. The first step determines the most probable block size for intra-prediction coding using the block smoothness based on the normalized ratio of AC and DC coefficient energies and the non-parametric thresholds calculated by the correlation between intra-prediction modes of neighboring blocks. In the second step, only for the pre-determined block size, the most probable intra-prediction mode is determined by the simplified calculation of local edge information based on the representative pixels among internal ones of the current block and boundary ones of reconstructed neighboring blocks. In the proposed algorithm, the computation overhead for the intra-prediction coding

---

Manuscript received November 10, 2009 ; revised November 30, 2009.

Young-ju Kim is with the Division of Computer Information, Silla University, Busan, 617-736, Korea (Tel: +82-51-999-5709, Fax: +82-51-999-5657, Email: yjkim@silla.ac.kr)

is fall off by the removal of the computation load for unlikely block sizes and the simplified calculation of edge information, and the non-parametric decision method excludes the dependency of the intra-coding performance to types of video sequences due to the use of empirical parameters. One restriction in this paper is that since most profiles of H.264/AVC don't support intra 8x8 prediction for luma component[8,10], only intra 4x4 and 16x16 predictions are considered. Experiment results show that the proposed algorithm can speed up more than 55% of encoding time for encoding an all I-frame sequences with a negligible loss in PSNR and bitrates compared to the full-search-based intra-prediction coding method.

The rest of this paper is organized as follows. In Section II, we introduce the intra-prediction coding algorithm of H.264/AVC and then review previous works. Section III presents the proposed algorithm. Then, Section IV shows the results of performance evaluation. Finally, conclusions are given in Section V.

II. RELATED WORKS

In H.264/AVC, for luma component, three types of intra prediction are defined according to the intra-coding block size: 16x16, 8x8 and 4x4, and for chroma components, only the 8x8 intra prediction is defined. In the intra-prediction of luma component, 9's directional intra prediction modes are used for intra 4x4 and 8x8 predictions as illustrated in Table 1. Fig. 1 shows 8's prediction modes designed in a directional manner for 4x4 and 8x8 blocks except a DC prediction mode numbered as mode 2. In Fig.2, 16 pixels labeled from *a* to *p* are pixels of a 4x4 block belonging to a macroblock to be coded, and the pixels *A* to *M* selected from the neighboring blocks being already reconstructed are used in the computation of predictions for pixels from *a* to *p*.

The 16x16 intra-prediction modes for luma component are selected in relatively homogeneous area and only four prediction modes are supported, which are vertical(Mode 0), horizontal(Mode 1), DC(Mode 2) and plane(Mode 3) prediction modes. These modes are specified similar to prediction modes supported in the 4x4 intra prediction except the plane prediction mode. The plane prediction mode uses weighted combination of horizontal and vertical adjacent pixels. For chroma components, there are 4 prediction modes applied to the two 8x8 chroma blocks (U and V), which are very similar to the intra 16x16 prediction modes for luma component such as DC(Mode 0), horizontal (Mode 1), vertical (Mode 2), and plane (Mode 3).

Table 1 Intra-prediction modes for 4x4 and 8x8 sub-blocks of luma component

Mode num.	Type of intra prediction mode
0	Vertical
1	Horizontal
2	DC
3	Diagonal-down-left
4	Diagonal-down-right
5	Vertical-right
6	Horizontal-down
7	Vertical –left
8	Horizontal-up

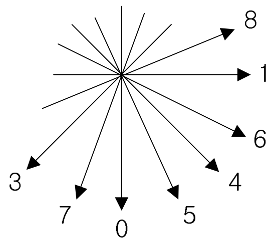

Fig. 1 Intra 4x4 and 8x8 prediction directions

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				

Fig. 2 Target and adjacent pixels for intra 4x4 prediction

To reduce the computation complexity of the intra prediction coding in H/264/AVC, several approaches have been proposed for the fast intra-prediction coding[5~11]. In [5] and [6], the local edge direction in a macroblock is calculated by using the edge directional histogram and the simple edge masks, respectively, and the probable prediction modes are determined by using the edge direction information. In [7], the smoothness of a macroblock is used to select the probable intra-coding block size, reducing the computation load needed by unlikely block sizes. The algorithm used the normalized ratio of AC and DC coefficient energies as the block smoothness and decided the probable block size by comparing with the threshold given by experiments. In [8] as an improved algorithm of [5], a probable block size is decided by comparing the amplitude of the edge histogram cell corresponding to the primary prediction mode with the given threshold. In [9], two primary amplitude thresholds and SATD of the 16×16 luma block are exploited to judge 4×4 or 16×16 block size for the intra-prediction. In [10], the boundary difference values between inner sub-blocks in a macroblock is calculated and compared with the given threshold to decide the probable block size for the intra-prediction. In [11], for the stabilization of visual quality performance, the non-parametric thresholds being calculated by the correlation between neighboring blocks is used to decide the probable block size.

To exclude the dependency of coding performance to video types and reduce the computation loads needed by the intra-prediction mode decision, this paper proposes a new fast intra-prediction mode decision algorithm that first, non-parametrically selects the most probable block size, and next, decides the best intra-prediction mode for the pre-determined block size based on the simplified computation of local edge information and the close correlation between intra-prediction modes of neighboring blocks.

III. PROPOSED FAST INTRA-PREDICTION MODE DECISION ALGORITHM

3.1 Intra-Coding Block Size Decision

First, this paper determines the intra-coding block size for the block to be predicted by using the non-parametric algorithm proposed in [11], which decides the best probable intra-coding block size by using the block smoothness and the non-parametric thresholds calculated based on the correlation between the current block and neighboring blocks.

The non-parametric algorithm adopts the normalized ratio of AC and DC coefficient energies as a measure for the smoothness of a block. To ease the calculation of AC/DC ratio, the 8x8 block is chosen as the basic unit by down-sampling from a 16x16 macroblock, and the following simple method is used to calculate the AC and DC coefficient energies:

$$DC\_energy = \frac{1}{8 \times 8}\left(\sum_{i=0}^{7}\sum_{j=0}^{7} a_{ij}\right)^2 \tag{1}$$

$$AC\_energy = \left(\sum_{i=0}^{7}\sum_{j=0}^{7} a_{ij}^2\right) - DC\_energy \tag{2}$$

And, the normalized ratio of AC and DC coefficient energies, AC_DC_ratio is calculated by

$$AC\_DC\_ratio = \frac{\log(AC\_energy)}{\log(DC\_energy \times 64)} \tag{3}$$

and is normalized between 0~1.

Usually, in Fig.3, the intra-prediction mode of the block *C* is correlated to reconstructed neighboring blocks *A* and *B*, and especially, the intra-coding block size of *A* or *B* can be the most probable block size for *C*. Thus, the non-parametric algorithm decides the intra-coding block size by directly comparing the normalized AC/DC ratio between the current block and neighboring blocks considering the block sizes of neighboring blocks. To ease the description, let *BS*(*K*) be the intra-coding block size for the block *K*, and *NR*(*K*) be the normalized AC/DC ratio for the block *K*. The decision process of the intra-coding block size in the non-parametric algorithm is shown in Fig. 4.

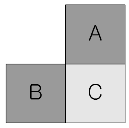

Fig. 3 Neighboring blocks for intra-prediction coding

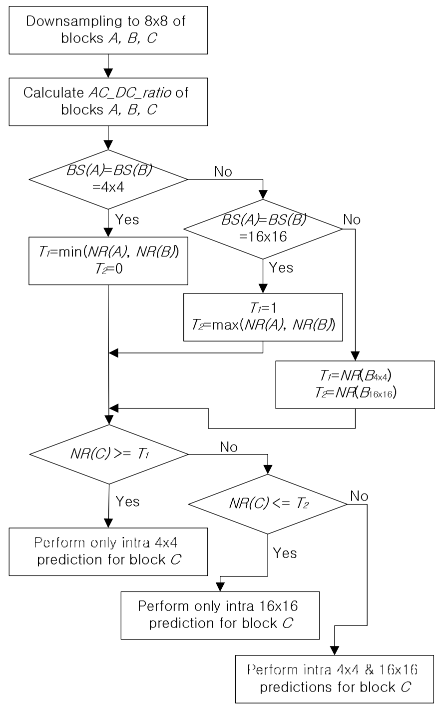

Fig.4 Intra-coding block size decision algorithm using non-parametric thresholds

This paper modified the non-parametric algorithm to return the differences of AC_DC_Ratio between the current block and neighboring blocks so as to use the information in the process for intra prediction mode decision.

3.2 Fast Intra Prediction Mode Decision

This paper proposed a novel fast algorithm deciding the intra-prediction mode for the pre-determined intra-coding block size. To reduce the computation loads by the intra-prediction mode decision, the proposed algorithm first determines candidate prediction modes by exploring two following features and decides the most probable prediction mode by executing RDO computation on only candidate modes: (i) the intra prediction mode has a close correlation with local edge directions of the block, and (ii) the intra prediction mode of the current block is highly correlated to ones of its neighboring blocks.

(1) Intra Prediction Mode Decision for 4x4 Luma Blocks

Based on the above first feature, the proposed algorithm uses several directional masks to extract local edge directions. Unlike previous works[6,8,9], the proposed algorithm uses boundary pixels of reconstructed neighboring blocks together in the extraction of local edge directions, so that being able to explore the information used in the intra-prediction coding phase. And to reduce the computation loads needed by the masking operation, the proposed algorithm selects the small number of representative pixels among internal ones of the current block and boundary ones of neighboring blocks and determines edge directions by applying directional masks to only representative pixels. The process of intra-prediction mode decision for 4x4 luma sub-blocks is as followed:

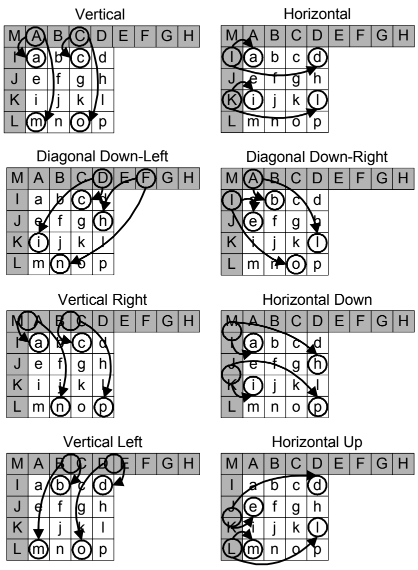

Fig. 5. Simplified directional masks for intra 4x4 prediction

Step 1. Calculate pixel variations between representative pixels of the current and neighboring blocks by applying 8’s directional masks depicted in Fig. 5. And select intra prediction modes corres-ponding to directional masks with minimum and secondarily small pixel variations as candidate modes.

Step 2. Being based on the second feature, select intra-prediction modes of neighboring blocks *A* and *B* as shown in Fig. 3 as candidate modes. Also select DC prediction mode being not included in 8’s directional masks as a candidate mode.

Step 3. Apply RDO computation to candidate modes to calculate RDCost(Rate-Distotion Cost) and decide the mode with the minimum RDCost as the best intra prediction mode for the 4x4 sub-block.

(2) Intra Prediction Mode Decision for 16x16 Luma Blocks

The proposed algorithm, like the intra-prediction mode decision for 4x4 sub-blocks, determines the intra-prediction mode by calculating pixel variations between representative pixels of the current block and neighboring blocks, and in the smooth area, there is a close correlation between intra prediction modes of the current block and neighboring blocks, so that the fact is applied in the intra prediction mode decision. The intra-prediction mode decision process for 16x16 blocks is as followed:

Step 1. Examine intra-coding block sizes of neighboring blocks *A* and *B*. If two block sizes are all 16x16, go to Step 2, or else go to Step 4.

Step 2. Examine intra-prediction modes of neighboring blocks *A* and *B*. If the modes are not the same, go to Step 3. If being the same, select the intra-prediction mode of a neighboring block as a candidate mode and go to Step 5.

Step 3. After determining the block with the minimum difference of AC_DC_Ratio among two blocks *A* and *B*, select the intra-prediction mode of the selected block as a candidate mode and go to Step 5.

Step 4. Calculate pixel variations between representative pixels of the current and neighboring blocks by using 3’s directional masks depicted in Fig. 6. And select the intra-prediction mode corresponding to the directional mask with the minimum variation as a candidate mode and go to Step 5.

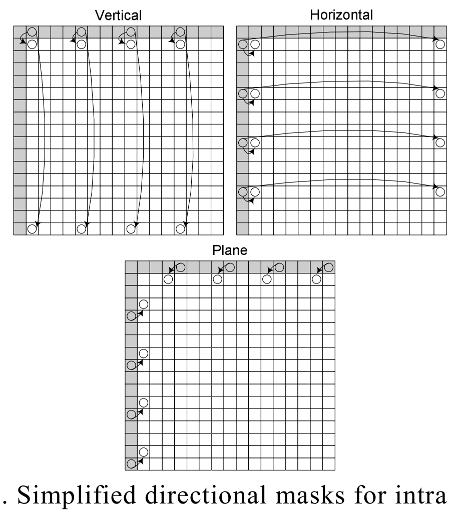

Fig. 6. Simplified directional masks for intra 16x16 prediction

Step 5. After applying RDO computation to the candidate mode and the DC mode, determine the mode with the minimum RDCost as the best intra-prediction mode for the 16x16 block.

(3) Intra Prediction Mode Decision for 8x8 Chroma Blocks

Since chroma information is generally down-sampled and coded, 8x8 chroma blocks have smooth and simple information, and therefore the intra-prediction mode decision for 8x8 chroma blocks is similar to that of 16x16 luma blocks. For 8x8 chroma blocks, the proposed algorithm modified and applied the intra-prediction mode decision method for 16x16 luma blocks described above. The modification includes the removal of intra-coding block size examination of neighboring blocks and the scale-down of 16x16 directional masks to 8x8 size.

IV. EXPERIMENTAL RESULTS

For performance evaluation, the proposed algorithm was implemented into JM 14.2, which is the H.264/ AVC reference software[12]. And, the proposed algorithm is tested by using three QCIF video sequences on the IBM-compatible PC equipped with Intel Pentium-4 2.8GHz and 1GB RAM. The test conditions are summarized in Table 2.

Table 2 Test conditions for performance evaluation

Video seq. type	Carphone, Foreman ,Football
Frame size	QCIF
Frame structure	All I frames
Number of frames	300
RD optimization	Enabled
Entropy coding	CABAC
QP	24, 28, 32

In this experiment, the proposed algorithm was compared with JM 14.2 and the results of performance comparison according to QP(Quantization Parameter) are shown in Table 3. In this table, a positive number means increasing and a negative one means decreasing. It is seen in Table 3 that the proposed algorithm is able to reduce the computation complexity considerably with a negligible loss in PSNR and bitrates. The amount of change in PSNR and bitrates is extremely low, indicating that the results of the proposed algorithm is very close to the full search method in JM 14.2 and the use of non-parametric thresholds supports the stable performance on visual quality regardless of types of video sequences. The time saving for the entire encoding process is from 48.95% to 63.58% and more than 55% on the average. When comparing with the performance results presented in previous works [6,8,9,10], the time saving is relatively less than previous works, and the performance difference is produced due to the calculation of non-parametric thresholds in the block size decision process.

Table 3 Performance comparison results between the proposed algorithm and JM 14.2

QP	Video Sequence	ΔTime (%)	ΔPSNR (dB)	ΔBits (%)
24	Carphone	48.95	0.0	0.47
	Foreman	51.33	0.0	0.26
	Football	58.12	-0.14	1.28
28	Carphone	53.21	0.0	0.65
	Foreman	54.73	-0.02	0.38
	Football	59.58	-0.2	1.72
32	Carphone	55.86	-0.1	0.81
	Foreman	59.32	-0.1	0.46
	Football	63.58	-0.3	1.89

Fig. 7 and 8 show the R-D(Rate-Distortion) curves and one of result frames of Football video sequences, respectively, indicating that there is almost no difference in the subjective visual quality. From Fig. 7 and 8, it can be observed that the performance of the proposed algorithm is similar to that of JM 14.2 with the full-search-based intra prediction coding.

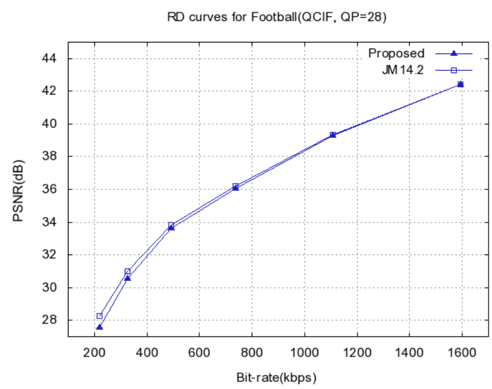

Fig. 7 RD curves of Football (QCIF, QP=28)

(a) JM 14.2 (b) Proposed Algorithm

Fig. 8 Comparison of subjective visual quality in Football video sequence

IV. CONCLUSIONS

This paper proposed and evaluated a fast intra-prediction mode decision algorithm in H.264/AVC. The proposed algorithm first decides the most probable intra-coding block size by using the normalized ratio of AC and DC coefficient energies as block smoothness and calculating non-parametric thresholds. And next, for the pre-determined block size, the proposed algorithm decides the intra-prediction mode by using the local edge information calculated by simplified directional masks and the close correlation between intra prediction modes of neighboring blocks. In the proposed algorithm, the use of non-parametric thresholds in the block size decision excludes the possibility of the dependency of intra coding performance on video types. Also, in the intra-prediction mode decision, the use of boundary pixels of reconstructed neighboring blocks is able to utilize the coding information used in the intra prediction coding phase and the simplified directional masks based on representative pixels among internal ones of the block to be coded and boundary ones of neighboring blocks reduces the computation loads needed. Experiment results show that compared to the H.264/AVC reference encoder, the proposed algorithm is able to achieve the time saving more than 55% with a negligible loss in PSNR and bitrates.

REFERENCES

[1] ITU-T Recommendation H.264 & ISO/IEC 14496-10 (MPEG-4) AVC, "Advanced Video Coding for Generic Audiovisual Services," (version 1: 2003, version 2: 2004, version 3: 2005).

[2] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC Video Coding Standard," IEEE Trans. on Circuits Syst. Video Technol., vol. 13, pp.560-576, 2003.

[3] Iain E. G. Richardson, "H.264 and MPEG4 Video Compression: Video Coding for Next Generation Multimedia", John Wiley and Sons, 2003.

[4] G. J. Sullivan, P. Topiwala, and A. Luthra, "The H.264/AVC advanced video coding standard: Overview and introduction to the fidelity range extensions," SPIE Conf. on applications of digital image processing XXVII, vol. 5558, pp. 53-74, 2004.

[5] F. Pan, X. Lin, S. Rahardja, K. Pang Lim, Z. G. Li, D. Wu, and S. Wu, "Fast mode decision algorithm for intraprediction in H.264/AVC video coding," IEEE Transactions on Circuits and Systems for Video Technology, vol. 15, Issue 7, pp.813-822, 2005.

[6] J. Kim, and J. Jeong, "Fast intra-mode decision in H.264 video coding using simple directional masks," Visual Communications and Image Processing 2005, Proceedings of SPIE, vol. 5960, pp.1071-1079, 2005.

[7] Yu-Kun Lin, and Tian-Sheuan Chang, "Fast Block Type Decision Algorithm for Intra Prediction in H.264 FRext," IEEE International Conference on Image Processing (ICIP 2005), vol. 1, pp. 585-8, 2005.

[8] LIU Qiong, HU Rui-min, and ZHU Li, et al. "Improved fast intra prediction algorithm of H.264/AVC," Journal of ZHEJIANG University Science A, vol. 7, pp. 101-105, 2006.

[9] ZHANG Kun, YUAN Chun, LI Qiang, and ZHONG Yuzhuo, "A Fast Block Type Decision Method for H.264/AVC Intra Prediction," The 9th International Conference on Advanced Communication Technology, vol. 1, pp. 673-676, 2007.

[10] Sungmin Kim, Takgi Lee, and Kidong Chung , "An Efficient and Fast Block Size Decision Exploiting Boundary Information on Inner Block for H.264/AVC Intra Prediction," IEEE Southwest Symposium on Image Analysis and Interpretation (SSIAI 2008), pp.101-104, 2008.

[11] Y. J. Kim, "A Non-parametric Fast Block Size Decision Algorithm for H.264/AVC Intra Prediction," International Journal of Maritime Information and Communication Sciences, vol.7, no.2, pp.193-198, 2009.

[12] H.264/AVC JM Reference Software Version 14.2, Available: http://iphome.hhi.de/suehring/tml/download/.

Young-ju Kim received his B.S., M.S., and Ph.D. degree in Dept. of Computer Science from Pusan National University in 1988, 1990, and 1999 respectively. From 1990 to 1995, he worked at ASRC(Application System Research Center) in Qnix Computer as a computer system development staff. From 2000, he joined the Division of Computer Information in Silla University, where he is presently an associate professor. His research interest is in the area of Multimedia Communication and Embedded System.