Journal of Universal Computer Science, vol. 18, no. 9 (2012), 1177-1193
submitted: 1/12/11, accepted: 28/4/12, appeared: 1/5/12 © J.UCS

Low Complexity H.264/AVC Intraframe Coding for Wireless Multimedia Sensor Network

Xingang Liu
(University of Electronic Science and Technology of China, Chengdu, China
hanksliu@uestc.edu.cn)

Jiantan Liu
(Yeungnam University, Kyungsan, South Korea
JiantanLiu@ynu.ac.kr)

Kook-Yeol Yoo
(Yeungnam University, Kyungsan, South Korea
kyoo@yu.ac.kr)

Haengrae Cho
(Yeungnam University, Kyungsan, South Korea
hrcho@yu.ac.kr)

Abstract: For the Wireless Multimedia Sensor Network (WMSN), the intraframe video coding is widely used for the robust transmission and computation complexity. Though the intraframe algorithm requires much smaller computational complexity than the interframe coding, the amount of computation of the intraframe should be reduced to use WMSN application. In this paper, we propose an intra mode decision algorithm to reduce the computation complexity of intraframe H.264/AVC encoders. The proposed algorithm determines the candidate modes and skips the remaining modes based on the smoothness and directional similarity of MB. The simulation results show that the proposed algorithm achieves 18% to 70% reduction in the computational complexity, compared with the various conventional methods..

Keywords: Wireless Multimedia Sensor, H.264/AVC, Intraframe Coding, Complexity
Categories: E.4, H.5.1, I.4.2, I.4.9, I.6.6

1 Introduction

With the advance of inexpensive hardware such as CMOS cameras and microphones that are able to ubiquitously capture multimedia contents from the environment has fostered the development of Wireless Multimedia Sensor Networks (WMSNs), i.e., networks of wirelessly interconnected devices that allow retrieving video and audio streams, still images, and scalar sensor data [Misra, 08]. Wireless multimedia sensor networks will not only enhance existing sensor network applications such as tracking, home automation, and environmental monitoring [Zhang, 11], but they will also enable several new applications such as multimedia surveillance sensor networks, storage of potentially relevant activities, traffic avoidance, enforcement and control systems, advanced health care delivery, automated assistance for the elderly and family monitors, environmental monitoring, person locator services, and industrial process control, etc [Akyildiz, 07].

For the video coding for WMSN networks, temporal prediction based on motion estimation and compensation requires too much complex processing, which lead to high energy consumption as explained in [Akyildiz, 07] and [Pudlewski, 10]. For the WMSN, the video data is mostly compressed by using the intraframe coding for the robust transmission and computational complexity [Zhang, 10].

For the intraframe coding, the H.264/AVC video compression standard [ITU, 10] has gained a huge interest in the video industries because it achieves about 50% performance improvement in coding efficiency over the conventional video compression standards, such as MPEG-4 Advanced Simple Profile codec [ISO, 04].

Though the H.264/AVC codec has successfully become a market leading video coding standard in a short period of time, its computational complexity, especially at the encoder side, creates a great bottleneck in bi-directional applications. Specifically, the encoding computational complexity is about five times more complex than that of the MPEG-4 Simple Profile encoder [ISO, 04]. There are several studies reducing the computational complexity by simplifying the intra mode decision procedures in [Yang, 04], [Park, 06]. In the H.264/AVC intraframe coding, three types of intra prediction methods are used, i.e., the INTRA16×16 and INTRA4×4 modes for luma components and INTRA8×8 modes for two chroma components [Liu, 10]. Since INTRA16×16 mode is used to predict the whole 16×16 luma block with a prediction direction selected from the four candidate directions, it is suitable for the smooth MB (Macro Block). On the other hand, INTRA4×4 mode is useful for the detail MB since each 4×4 luma sub-block, partitioned from 16×16 luma, is predicted separately with one selected direction selected from nine candidate directions. The INTRA8×8 mode is used to predict the whole 8×8 chroma blocks with one selected prediction direction.

Based on these characteristics, Yang proposed to branch the mode decision into INTRA16×16 mode or INTRA4×4 mode based on the MB smoothness [Yang, 04]. This branching method can significantly reduce the complexity, but there is no remedy for the computationally intensive INTRA4×4 mode decision.

Park showed that the prediction directions between INTRA16×16 and INTRA4×4 modes are correlated with each other [Park, 06]. In [Park, 06], the full search mode decision for the INTRA16×16 mode is performed, and the predefined candidate directions similar to the decided INTRA16×16 mode are then considered for the INTRA4×4 mode decision. For each 4×4 sub-block, the number of candidates is reduced from 9 to 6 or 7 [Park, 06]. For the detail MB with small objects or object boundaries, however, the similarity between INTRA16×16 and INTRA4×4 modes is very low. Therefore, the coding efficiency is significantly reduced for the sequences with small objects and complex textures, suffering from serious quality degradation.

Based on the two methods of [Yang, 04] and [Park, 06], a new intra mode decision algorithm is proposed in this paper. The proposed algorithm adopts the idea from [Yang, 04] for the branching of INTRA16×16 and INTRA4×4 modes. However, for the decision of candidate INTRA4×4 directions, the prediction direction of INTRA16×16 mode is derived from the MB smoothness without the full INTRA 16×16 mode decision, unlike [Park, 06]. For the detail MB, the candidate INTRA4×4 directions are derived from the statistical analysis for test sequences. Notice that the number of candidates in the proposed method is much smaller than that of [Park, 06].

Unlike [Park, 06], we further verified the strength of smoothness to check whether or not the prediction directions of 16×16 block and 4×4 blocks is similar. If the similarity is low, then a full INTRA4×4 mode decision is performed to avoid abrupt quality degradation, which is the serious problem in [Park, 06].

This paper is organized as follows. Chapter 2 explains the conventional intra mode decision algorithms. The proposed algorithm is presented in Chapter 3. Chapter 4 shows the simulation results that confirm the performance of our proposed algorithm. The conclusions are drawn in Chapter 5.

2 Overview of Intra Prediction in H.264/AVC

2.1 Full Search Algorithm (FSA)

Spatial DPCM (Difference Pulse Coded Modulation) is adopted for the intraframe coding in H.264/AVC. For the luma component of Macroblock (MB), the two prediction modes are used, such as INTRA16×16 and INTRA4×4 modes. The INTRA8×8 mode is used for the chroma components of MB.

The four prediction directions are available for the INTRA16×16 mode as shown in Figure 1. In the figure, the shaded and white areas represent the 16×16 luma pixels to be predicted and the reconstructed pixels, respectively. For the Mode 0 in figure 1, all pixels in the shaded area are predicted from the pixels in the upper white area. This prediction direction is effective if the pixels in the current block have little intensity change in the vertical direction like figure 3-(a). All the prediction directions in the INTRA16×16 mode assume that all the pixels in the current block are changing smoothly.

For the MB with detail texture, the INTRA4×4 mode achieves better performance in terms of prediction accuracy. For the INTRA4×4 mode, the 16×16 luma block is partitioned into sixteen 4×4 sub-blocks. For each 4×4 sub-block, different prediction methods can be used to achieve better prediction accuracy. Figure 2 illustrates the nine prediction directions of the INTRA4×4 mode. Prediction direction 2, which is not shown in figure 2, is the DC prediction. The current pixels in the small capitals are predicted from the reconstructed pixels in the large capitals. For example, in case of direction '0', the pixels of 'a', 'e', 'i', and 'm' are predicted by pixel 'A'. The INTRA8×8 mode is used for predicting the two 8×8 chroma components of MB. Prediction mode has the same four prediction directions as those of INTRA16×16 mode.

To determine the best prediction directions for each MB, all possible prediction directions are checked in the Full Search Algorithm (FSA). The procedure of the FSA algorithm is as follows:

Step 1: INTRA8×8 mode (chroma)

Four prediction blocks are generated according to the four prediction directions.

Step 2: INTRA16×16 mode (luma)

Four prediction blocks are generated according to the four prediction directions. Find the best INTRA16×16 luma prediction mode in the sense of Rate-Distortion (RD) cost.

Step 3: INTRA4×4 mode (luma)

For each 4×4 block, nine prediction blocks are generated according to the nine prediction directions. Find the best INTRA4×4 luma prediction mode in the sense of RD cost for each 4×4 block.

Step 4: Best luma prediction mode
Select the best luma prediction mode between the prediction modes from Steps 2 and 3.

Step 5: Decision of the MB intra prediction mode
Compute the combined costs of best luma prediction mode for each chroma mode. The combination of luma and chroma modes with minimum RD cost becomes the MB prediction mode.

H.264/AVC uses the Rate Distortion Optimization (RDO) technique for the improved coding efficiency. For the computation of the distortion in RDO, the actual encoding processes, such as transformation, VLC, etc, are performed on the possible combinations of prediction modes to determine the MB mode. Therefore, the computational complexity is extremely high, compared with that of previous standard video encoders.

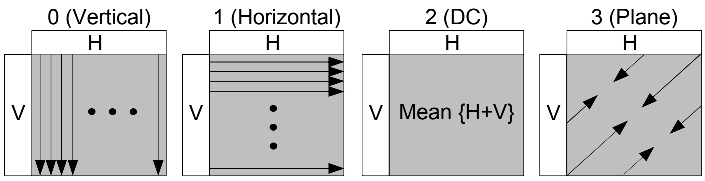


Figure 1: Four prediction directions of INTRA16×16 mode

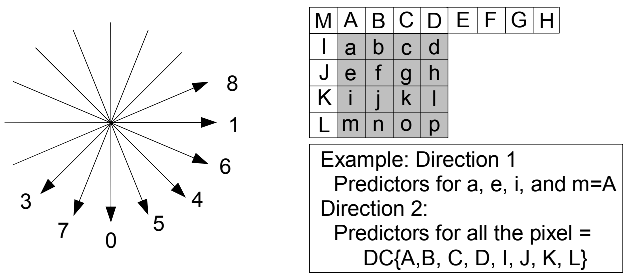


Figure 2: Nine prediction directions of INTRA4×4 mode

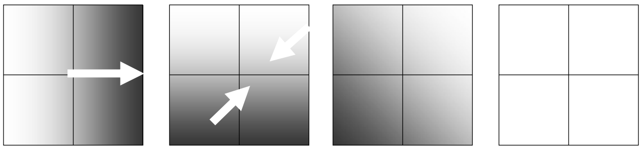


Figure 3: Example of Directional smoothness of 16×16 luma block: (a) vertical, (b) horizontal, (c) planar, (d) DC

2.2 Branching Method between INTRA16×16 and INTRA4×4 Modes Based on MB Smoothness Decision

The two luma prediction modes are closely related to the smoothness of MB. There is high possibility that the smooth MB is predicted by using the INTRA16×16 mode. Based on this fact, Yang [Yang, 04] proposed the branching method between INTRA16×16 and INTRA4×4 modes for the luma components. The four types of smoothness, as shown in figure 3, were checked. If the MB is decided as smooth MB, then only the INTRA16×16 mode is used and all the procedures associated with the INTRA4×4 mode are skipped. Otherwise, only INTRA4×4 mode is considered in the mode decision. This method reduces Steps 2 and 3 in the FSA algorithm into only one step.

For the decision of MB smoothness, the four cost functions are proposed in [Yang, 04] as follows:

- *DC smoothness*

  For the DC prediction of INTRA16×16 mode, the mean of the above and left pixels is used to predict all the pixels in a 16×16 luma block. The DC smoothness is evaluated by the Mean Absolute Difference (MAD) defined as follows:

  $$\mathrm{MAD_{DC}} = \frac{1}{256}\sum_{x=0}^{15}\sum_{y=0}^{15}|p(x,y) - m| \qquad (1)$$

  where $p(x, y)$ and m are the luma pixel value at (x, y) and the mean of the neighboring luma pixels in MB, respectively. If MAD_{DC} is smaller than the predefined threshold T_{DC}, then the MB is DC smooth.

- *Vertical smoothness*

  If MAD_V is smaller than a predefined threshold T_V, then MB is vertically smooth.

  $$\mathrm{MAD_V} = \frac{1}{256}\sum_{x=0}^{15}\sum_{y=0}^{15}|p(x,y) - m_x| \qquad (2a)$$

  $$m_x = \frac{1}{16}\sum_{x=0}^{15} p(x,-1) \qquad (2b)$$

  where m_x is the mean of the upper pixels of MB.

- *Horizontal smoothness*

  If MAD_H is smaller than the predefined threshold T_H, then MB is horizontally smooth.

  $$\mathrm{MAD_H} = \frac{1}{256}\sum_{x=0}^{15}\sum_{y=0}^{15}|p(x,y) - m_y| \qquad (3a)$$

  $$m_y = \frac{1}{16}\sum_{y=0}^{15} p(-1,y) \qquad (3b)$$

where m_y is the mean of the left pixels of MB.

- *Planar smoothness*
  MAD_{VV} of Plane mode is calculated as (2) but $Block_V(x,y)$ is used instead of $p(x,y)$. For MAD_{HH}, $Block_H(x,y)$ is used. If both MAD_{VV} and MAD_{HH} are smaller than a predefined threshold T_P, then the MB is planar smooth.

$$\mathrm{Block_V(x,y) = p(x,y+1) - p(x,y)} \quad (4a)$$
$$\mathrm{Block_H(x,y) = p(x+1,y) - p(x,y)} \quad (4b)$$

Compared with FSA, this algorithm significantly reduces the computational complexity, since only one prediction type is considered for the luma prediction mode decision. The additional computational complexity of the smoothness check is negligible.

2.3 Selective Mode Decision Algorithm

Park's algorithm [Park, 06] mainly focused on reducing the computational complexity of the INTRA4×4 mode decision. The key idea is that the INTRA4×4 prediction directions of the 4×4 sub-blocks in a MB are correlated with the INTRA16×16 prediction direction. For instance, if the INTRA16×16 mode for a given MB is decided as the direction 1 (Horizontal), then there is high possibility that the directions of the 4×4 sub-blocks inside MB are decided as directions 1, 6, or 8 (See figures 1 and 2). Based on this idea, the INTRA4×4 mode is classified into several candidate directions for a given INTRA16×16 mode. The candidate direction groups are summarized in Table 1. Notice that direction 2 (DC) is commonly included, since direction 2 occurs most frequently in the INTRA4×4 mode decision. U and L in the Table 1 represent the 4×4 luma prediction directions of the upper and left 4×4 sub-blocks, which are previously encoded sub-blocks.

16×16 mode	**Candidate 4×4 direction group**
0 (vertical)	7, 0, 5, 2, directions of U, L
1 (horizontal)	1, 6, 8, 2, directions of U, L
2 (DC)	0, 1, 3, 4, 2, directions, U, L
3 (plane)	0, 1, 3, 2, directions of U, L

Table 1: Candidates 4×4 directions according to 16x16 modes

The algorithm is performed in the following procedure: first, the INTRA16×16 mode is decided by using Step 2 of the FSA algorithm. The INTRA4×4 mode decision is performed for only one candidate group specified in Table 1. Therefore, this algorithm reduces nine directions into six or seven candidates.

For the INTRA8×8 chroma mode, the similarity in the intensity change of luminance and chrominance signals is used in [Huang, 05]. The candidate INTRA8×8 prediction directions of the 8×8 chrominance blocks are the best prediction direction of 16×16 luma mode and DC direction. Since the DC direction is the most probable direction in the INTRA8×8 chroma mode, it is always included in the candidate to provide robustness.

3 Proposed Mode Decision Algorithm

3.1 Analysis of the Previous Methods and Derivation of the Proposed Algorithm

Since the INTRA16×16 mode is introduced to predict the smoothly varying block, the method in [Yang, 04] is very effective for the smooth sequences by frequently skipping the computationally expensive INTRA4×4 mode decision. On the other hand, since the algorithm in [Park, 06] only considers the pre-defined candidate INTRA4×4 prediction modes for a given INTRA16×16 mode, it can reduce the computations of the INTRA4×4 mode decision. This method is effective for the detail sequences in which the INTRA4×4 mode is frequently selected [Park, 06]. In [Huang, 05], the INTRA8×8 mode decision is skipped by just taking the result from the INTRA16×16 mode decision.

Though each method of [Yang, 04], [Park, 06], and [Huang, 05] has advantage for a certain case, they produce poor results for sequences with different characteristics from their assumptions. For [Yang, 04], once one type of INTRA prediction mode is selected, it should perform the full search mode decision for a selected INTRA mode type. The method in [Park, 06] produces low coding efficiency for detail sequences, because it only considers the candidate INTRA4×4 directions, whether or not the current MB is non-homogeneous in the sense of the directional correlation. For instance, 4×4 sub-blocks in MB's with multiple object boundaries or small objects, textured MB's, etc, may have different directional correlation from the prediction direction of INTRA16×16 mode. In this case, it would result in poor coding efficiency to predict the 4×4 sub-blocks with similar directions of the INTRA16×16 mode.

To take advantage of mode branching in [Yang, 04] and the reduced candidates in the INTRA4×4 mode decision in [Park, 06] while maintaining the coding efficiency, we propose a hybrid INTRA mode decision algorithms by combining the concepts from these methods and [Huang, 05] and by introducing the special full search INTRA4×4 mode decision of the non-homogeneous MBs.

Since most computations in INTRA mode decision are used for the INTRA4×4 mode decisions, further reduction of candidates for mode decision is highly necessary. Notice that the number of candidate INTRA4×4 directions in [Park, 06] is reduced from 9 to 6 or 7, only removing the 2 or 3 candidates. In the subsequent section, we empirically investigate the relationship between the smoothness of a given MB and the INTRA4×4 mode to reduce the number of candidates for INTRA4×4 mode decision.

3.2 Investigation on the relationship between the MB smoothness and INTRA4×4 mode

To know the relationship between the smoothness of the 16×16 block and INTRA4×4 prediction directions of the sub-blocks, the conditional probabilities of INTRA4×4 prediction directions for a given smoothness of 16×16 block are measured. The decision of smoothness type is calculated by using the cost functions of (1)-(4). The INTRA4×4 mode is decided by using the FSA algorithm described in Chapter 2. The results are summarized in Table 2. For the MB determined as vertically smooth MB

(V), the most probable INTRA4×4 prediction directions of the sub-blocks in the MB are ‘7’, ‘0’, ‘5’, and ‘2’.

Notice that the conditional probability in Table 2 shows that 32.28% MBs are decided as the 4×4 directions different from the candidate directions. It implies that the FSA mode decision is still necessary when the 4×4 sub-blocks have low directional correlations. For instance, several sub-blocks and others in 16×16 block simultaneously have strong correlation in the horizontal and vertical directions, respectively. For this example, if the smoothness of 16×16 block is decided as horizontal smooth block, the sub-blocks with vertical smoothness cannot be predicted accurately by using one candidate group. Notice that these kinds of situations frequently occur at the boundaries of objects. For this case, the quality can be significantly degraded. This is the reason why the algorithm of [Park, 06] suffers from low coding efficiency.

Identification of such non-homogeneous areas in directional correlation is very important for coding efficiency. According to the MADs used for the smoothness test of the 16×16 block, MB can be classified into three categories; (i) If the MADs are sufficiently small, then MB can be predicted by using 16×16 mode, (ii) else, if the MADs are smaller than the certain threshold, then most 4×4 sub-blocks have similar directional correlation with INTRA16×16 mode, and they can be predicted with the candidate INTRA4×4 directions, (iii) If the MADs are large, then it cannot be predicted accurately by using the candidate directions. In case of (iii), all the prediction directions are checked to decide the final INTRA4×4 mode, like FSA.

Candidate group	**16×16 block Smoothness**	**Candidate 4×4 mode group**	**Conditional probability**
V	0 (vertical)	7, 0, 5, 2	68.0%
H	1 (horizontal)	1, 6, 8, 2	65.7%
DC	2 (DC)	0, 1, 3, 4, 2	78.3%
P	3 (Plane)	0, 1, 3, 2	58.9%

Table 2: Candidates 4x4 directions for 16×16 block smoothness

3.3 Procedure of the Proposed Mode Decision Algorithm

The proposed algorithm decides the MB prediction mode for luminance components according to the following procedure:

Step 1: Smoothness decision

If $MAD_{DC} \le T_{DC}$, go to Step2
Else if $MAD_V \le T_V$, go to Step2
Else if $MAD_H \le T_H$, go to Step2
Else if ($MAD_{HH} \le T_P$ && $MAD_{VV} \le T_P$), go to Step2
Else go to Step3

Step 2: 16×16 prediction mode decision

Select the best 16×16 prediction direction among the four directions and terminate the mode decision

Step 3: 4×4 prediction mode decision

If min{MAD_V, MAD_H, MAD_{DC}, MAD_P } < T_S, then go to Step 3.1
Otherwise Step 3.2

Step 3.1: Search the mode among the candidate group

$g = \min\{ MAD_g \}$, where g = V, H, DC, P

Note that candidate 4×4 directions are shown in Table 2. After 4×4 mode decision, terminate the mode decision

Step 3.2: Search the best direction among all 4×4 prediction directions

The mode decision algorithm for chrominance components is as follows:

Step 1: If MB is not smooth, then only use the DC direction for chroma mode decision and terminate, otherwise go to Step 2.

Step 2: If MB is V-smooth, then the candidates are vertical and DC directions. For H-smooth case, the candidates are horizontal and DC directions. For P-smooth case, the candidates are planar and DC directions. For DC-smooth case, the candidate is DC direction.

3.4 Simplified Analysis on the Computational Complexity

The numbers of candidate directions for several methods are summarized in Table 3. Since the mode decision is branched into 16×16 or 4×4 prediction modes in [Yang, 04] and the proposed method, the two cases are separately represented in the table. Table 3 indicates that the proposed algorithm requires the minimum number of mode searches, compared with all methods for both smooth and non-smooth MBs. Compared with the methods in [Liu, 2009], [Yang, 04] and [Park, 06], the performance can be varied according to the ratio of smooth MBs for each frame and image characteristics, which can affect the RD optimization. In case of RD optimization, the total number of candidates is determined by multiplying the numbers of luma and chroma components. For instance, the number of RD cost checks in FSA algorithm is (Luma: 4+9×16)×(Chroma: 4) = 592. On the other hand, the number of cost checks without RD optimization procedure is (Luma: 4+9×16)+(Chroma: 4) = 152.

For better understanding, the average number of cost checks for [Yang, 04] and [Park, 06] according to the RD optimization procedure are represented as follows:

$$\bar{n}_{\mathrm{FSA}} = 592 \tag{5a}$$

$$\bar{n}_{\mathrm{C1}} = \{4P_{\mathrm{S}} + 16 \times 9(1 - P_{\mathrm{S}})\} \times 4 \tag{5b}$$

$$\bar{n}_{\mathrm{C2}} = \left\{4 + 16 \times \left(6P_{\mathrm{C2}}^{4,6} + 7P_{\mathrm{C2}}^{4,7}\right)\right\} \times \left(1P_{\mathrm{C2}}^{8,1} + 2P_{\mathrm{C2}}^{8,2}\right) \tag{5c}$$

$$\bar{n}_{\mathrm{PR}} = \left\{4P_{\mathrm{S}} + \left[16 \times \begin{pmatrix} \left(4P_{\mathrm{PR}}^{4,4} + 5P_{\mathrm{PR}}^{4,5}\right)\left(1 - P_{\mathrm{PR}}^{4,9}\right) \\ + \\ 9P_{\mathrm{PR}}^{4,9} \end{pmatrix} \times \; (1 - P_{\mathrm{S}})\right]\right\} \times \{2P_{\mathrm{S}} + (1 - P_{\mathrm{S}})\} \tag{5d}$$

where $\bar{n}_{\mathrm{FSA}}$, $\bar{n}_{\mathrm{C1}}$,$\bar{n}_{\mathrm{C2}}$ and $\bar{n}_{\mathrm{PR}}$ represent the average number of mode searches for FSA, [Yang, 04], [Park, 06], and the proposed algorithm, respectively. P_{S} is the probability that the current MB is decided as smooth MB. $P_{\mathrm{C2}}^{4,6}$ and $P_{\mathrm{C2}}^{4,7}$ represent the probabilities that the 4×4 mode in [Park, 06] has 6 and 7 candidates, respectively. According to Table 1, $P_{\mathrm{C2}}^{4,7}$ means the probability that the 16×16 mode in [Park, 06] is decided as 'DC'. Similarly, $P_{C2}^{8,1}$ and $P_{C2}^{8,2}$ are defined for the 8×8 mode. The probability $P_{PR}^{4,9}$ represents the case that MB is decided as non-smooth MB and has no correlation between the 16×16 and 4×4 modes. For this case, the FSA algorithm is applied to the 4×4 mode.

For the simplicity of analysis, it is assumed that the directions for a given mode type is uniformly distributed. Based on Tables 1 and 2, $P_{C2}^{4,6} = 1 - P_{C2}^{4,7} = 3/4, P_{C2}^{8,1} = 1/4 = 1 - P_{C2}^{8,2}$, and $P_{PR}^{4,4} = 1 - P_{PR}^{4,5} = 3/4$. Therefore, (5) can be rewritten as follows:

$$\bar{n}_{C1} = 576 - 560P_S \quad (6a)$$

$$\bar{n}_{C2} = 165 \quad (6b)$$

$$\bar{n}_{PR} = 72 + 72P_{PR}^{4,9} + 4P_S - \left\{68 + 72\left(P_{PR}^{4,9}\right)^2\right\}(P_S)^2 \quad (6c)$$

The results of (6) are depicted in figure 4 with respect to P_S and $P_{PR}^{4,9}$. For [Yang, 04], as expected in Table 3 and (6a), the encoding time is linearly decreased to the smoothness ratio of MBs, P_S. According to (6a) and (6b), the same performance of [Yang, 04] and [Park, 06] methods in average mode checks occurs at $P_S = 73.39\%$. It means that most MBs should be decided as homogeneous MBs for [Yang, 04] to perform better than [Park, 06], which is very rare in real images. It can be interpreted that [Yang, 04] can perform better than [Park, 06] for the synthetic image, where most objects in the image are represented in single color. For the proposed method, the performance is significantly better than both [Yang, 04] and [Park, 06] methods for the wide range of P_S. Notice that the $P_{PR}^{4,9}$ value in Table 2 is 32.28%. Performance of the proposed method is monotonically improved with respect to the increase of P_S and the decrease of $P_{PR}^{4,9}$. For the performance variation to P_S, [Yang, 04] is very sensitive, while [Park, 06] and the proposed method are relatively insensitive. Therefore, the performance of [Yang, 04] can be highly dependent on the characteristics of the test sequences.

In case of $P_{PR}^{4,9} = 30\%$, about 30% of MBs cannot be predicted accurately by the 4×4 candidate directions. It means that [Park, 06] can result in poor performance in the coding efficiency; in other words, low visual quality at a given bitrate. But the proposed method can maintain visual quality by the FSA mode search for the non-homogeneous MBs, and also the computational complexity is still low due to the merits of smooth MB and other MBs using 4×4 prediction candidates.

From these simplified analyses, it can be easily interpreted that the proposed algorithm can significantly reduce the computational complexity and guarantees the visual quality of non-homogeneous MBs.

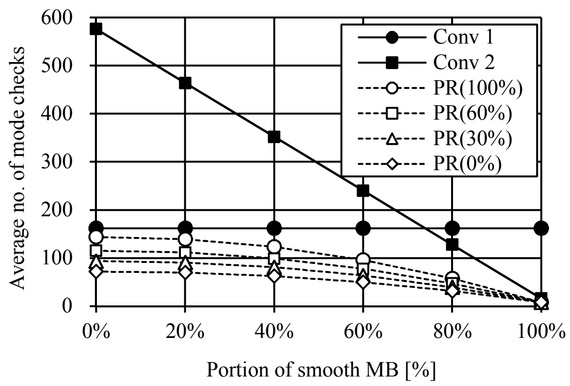


Figure 4: Average number of cost checks; xx in PR(xx%) represents the proposed method with $P_{PR}^{4,9} = xx\%$; Conv 1: conventional branching method, Conv2: conventional selective method

Method	Luma		Chroma
	16×16 candidates	**4×4 candidates**	**8×8 candidates**
FSA	4	9	4
Conv 2	4	6 or 7	1 or 2
Conv 1	4 (smooth)	0	4
	0(non-smooth)	9	4
Proposed method	4 (smooth)	0	2
	0(non-smooth)	{4 or 5} or 9	1

Table 3: Comparison of number of candidates in mode decision

4 Simulation Results and Comparison

All the simulations are performed on a 3.0GHz PC and the simulation conditions are as follows:

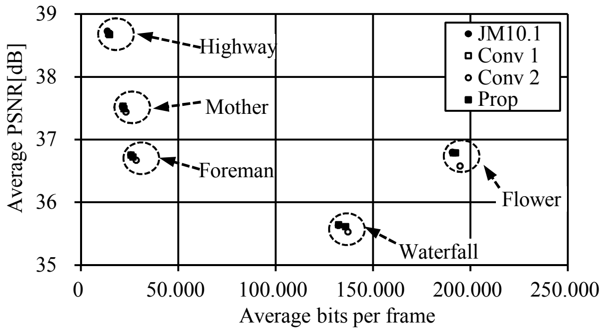
- JM 10.1 software codec
- H.264 Baseline profile
- RD optimization
- CABAC
- GOP structure : 100 I-frames
- Test sequences: 'Foreman', 'Highway', and 'Mother' in QCIF formats and 'Waterfall' and 'Flower' in CIF formats.

Since the proposed algorithm is mainly for Intra prediction, all the frames are encoded as I-picture. The performances are evaluated in terms of PSNR (Peak SNR), bit-rate and computational complexity (encoding CPU time) for the four algorithms, such as FSA, conventional branching method (Conv 1) of Section 2.2, selective method (Conv 2) of Section 2.3, and proposed algorithm.

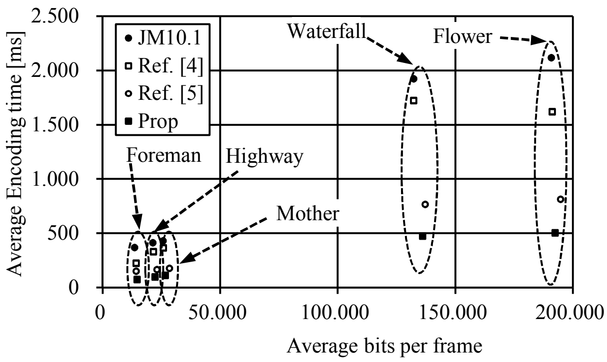
4.1 Performance Evaluation for the Various Sequences

Conventionally, the coding efficiency of video encoder is evaluated in the terms of Rate-Distortion (RD) theory. In other words, for a given quality, the bitrates are compared. Reversely, for a given bitrate, the qualities are compared.

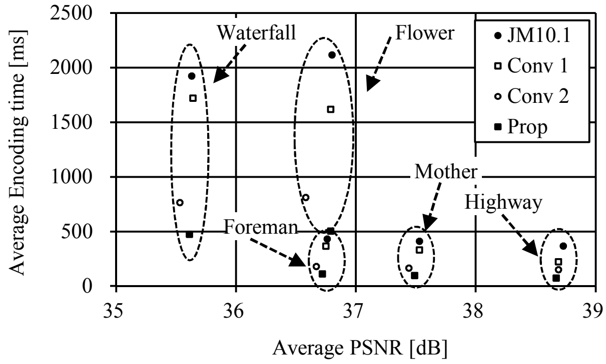
Since the method in this paper should be evaluated in three cost functions, the performance of the methods should be evaluated in various ways. The simulation is performed for the fixed Quantization Parameter (QP) of 28. In figure 5-(a), the RD curve implicates that all algorithms showed similar performance, except the minor low coding efficiency of conventional Conv 2 method. As previously analyzed in Chapter 3, this low performance of Conv 2 method comes from the fact that the 4×4 mode decision is applied to only candidate directions correlated with the direction of 16×16 mode, even though there is no correlation between the 16×16 and 4×4 modes for some MBs.



(a) bits/frame vs. PSNR



(b) bits/frame vs. encoding time



(c) PSNR vs. encoding time

Figure 5: Performance comparison for various sequences; average data for 100 I-frames for each sequence; QP=28

According to figures 5-(b) and (c), the performance differences in the computational complexity are evident. The low performance of Conv 1 is due to the fact that the portion of smooth MBs is low for the 'Waterfall' and 'Flower' sequences.

Figure 6 shows the encoding time reduction relative to the FSA for the various sequences; in other words, the different portion of smooth and non-homogeneous MBs in the frame. The proposed and Conv 2 methods are relatively insensitive to the change of sequences. The figure also shows that the Conv 1method is very sensitive to the image characteristics, as expected from the result of analysis in the previous section.

The average data for all sequences are summarized in Table 4. The proposed method reduces encoding time by 15.32 % over the Conv 2 method with a minor improvement in the coding efficiency. The proposed method achieves the great improvement over the Conv 1 method, i.e., 56.95% reduction in the encoding time. In summary, the proposed algorithm can achieve great computational reduction while maintaining almost the same coding efficiency.

Method	**PSNR-Y [dB]**	**Bits per frame**	**ΔBits**	**Time [ms]**	**ΔTime [%]**
JM10.1	37.09	76,716	-	1052	-
Conv 1	37.08	77,069	0.46%	854	-18.82%
Conv 2	36.98	79,526	3.66%	416	-60.45%
Proposed	37.06	78,410	2.21%	255	-75.77%

Table 4: Average performance comparison for various sequences, (Average of 100 I-frames, QP = 28)

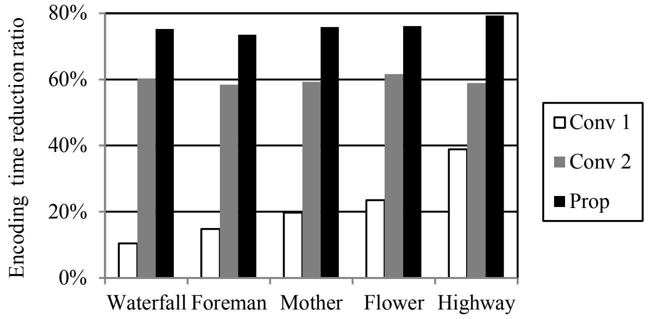
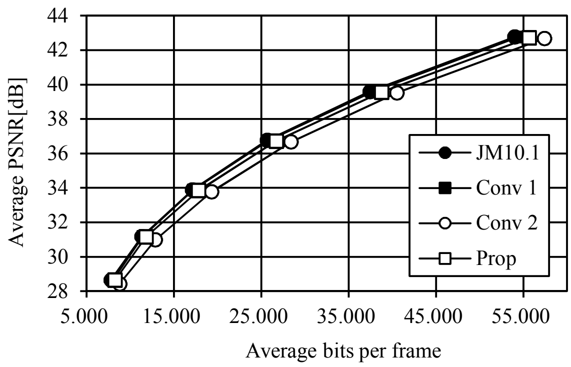


Figure 6: Encoding time reduction ratio relative to JM10.1 method; average data for 100 I-frames for each sequence; QP=28

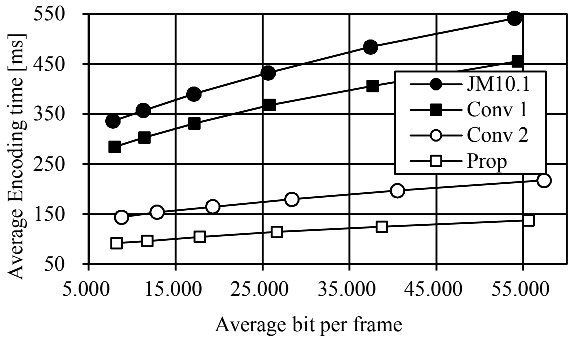
4.2 Performance Evaluation for the Various Quantization Parameters

To evaluate the performance variation to the QPs, the 'Foreman' sequence is used because it shows middle performance in coding efficiency in the previous section. The QP values used in this simulation are 20, 24, 28, 32, 36, and 40 to know the performance over the wide range of the bitrates. The other simulation conditions are the same as the ones used in the previous section.

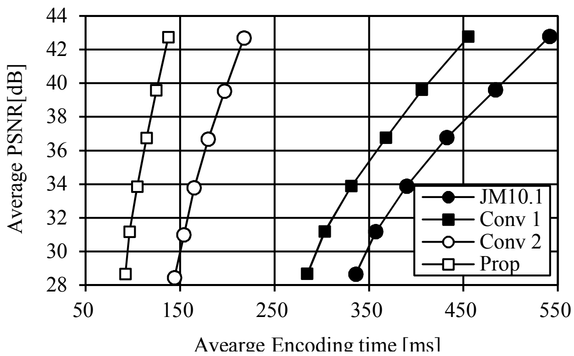
Figure 7-(a) depicts the performance of coding efficiency. The results show similar characteristics as shown in the previous section. The Conv 2 method shows rather low performance due to no remedy for the non-homogeneous MB. As the bit allocated to each frame is increased, the performance gaps between the groups of 'JM10.1 and Conv 1' and 'Conv 2 and the proposed method' are increased. This characteristic comes from the fact that the 4×4 modes are selected more frequently because of the increased available bits. Figures 7-(b) and 7-(c) show that the proposed method can achieve the great reduction in the computational complexity while almost maintaining same coding efficiency. Also, the proposed method is relatively insensitive to the variation of bitrates, compared with JM10.1 and Conv 1 methods. The average data for all the QP values are summarized in Table 5.



(a) bits/frame vs. PSNR



(b) bits/frame vs. encoding time



(c) Encoding time vs. PSNR

Figure 7: Performance comparison for 'Foreman' sequences; average data for 100 I-frames; QP=20, 24, 28, 32, 36, and 40;

Method	**PSNR-Y [dB]**	**Bits per frame**	**ΔBits**	**Time [ms]**	**ΔTime [%]**
JM10.1	35.47	25,554	-	423	-
Conv 1	35.47	25,774	1.15%	358	-15.38%
Conv 2	35.34	27,885	10.75%	176	-58.20%
Proposed	35.44	26,495	4.14%	112	-73.50%

Table 5: Average Performance Comparison for Various QPs ('Foreman', Average of 100 I-frames, QP = 20, 24, 28, 32, 36, and 40)

5 Conclusion

In this paper, we proposed an intra mode decision method to reduce the computational complexity of the H.264/AVC Intraframe coding for both luminance and chrominance components for the WMSN application. From the statistical analysis, (i) the luma 4×4 mode can be classified into several candidate groups, (ii) the chroma 8×8 mode can be reduced to 1 or 2 candidate directions, and (iii) the full 4×4 mode search is necessary for the non-homogeneous MBs. By combining the two conventional methods and introducing the partial full search mode, the proposed method can greatly reduce the computational complexity while maintaining the coding efficiency. Performance of the proposed algorithm is verified by the statistical and numerical analyses and simulation with test sequences. The proposed algorithm can be a very good tool for future bi-directional applications using the H.264/AVC codec.

Acknowledgements

The corresponding author, K.-Y. Yoo thanks the anonymous reviewers for their invaluable comments. This research was supported by the Yeungnam University research grants in 210-A-380-131.

References

[Akyildiz, 07] Akyildiz, I. F., Melodia, T., Chowdhury, K. R.: A survey on wireless multimedia sensor networks, In Computer Networks, Elsevier, Vol. 51, No. 4, pp. 921-960, 2007

[Huang, 05] Huang, Y.-W, Hsieh, B.-Y, Chen, T.-C., Chen, L.-G.: Analysis, fast algorithm, and VLSI architecture design for H.264/AVC intra frame coder. In IEEE Trans. on Circ. and Syst. for Video Technol., Vol.15, No. 3, pp. 378-401 March 2005.

[ITU, 10] ITU-T Recommendation H.264: Advanced video coding for generic audiovisual services, March 2010

[ISO, 04] ISO/IEC 14496-2, Information technology - Coding of audio-visual objects -Part 2: 3rd Ed., June 2004

[Kandrisa, 11] Kandrisa, D., Tsagkaropoulosb, M., Politisb, I., Tzesb, A., Kotsopoulosb, S.: Energy efficient and perceived QoS aware video routing over Wireless Multimedia Sensor Networks, In Multimedia Ad Hoc and Sensor Networks, Elsevier, Vol. 9, No. 4, pp. 591-607, June, 2011

[Liu, 09] Xiangyang L., Hongtao L. and Daqiang Z., “Head Pose Estimation Based on Manifold Embedding and Distance Metric Learning,” in Proc. of the 9th Asian Conference on Computer Vision, 2009, pp. 120 – 131.

[Liu, 10] Liu X., Yoo K.Y., Kim S.W.: Low complexity Intra prediction algorithm for MPEG - 2 to H.264/AVC transcoder, In IEEE Trans. on Consumer Electronics, Vol.56, No. 2, pp. 987-994 May 2010.

[Misra, 08] Misra, S., Reisslein, M., Xue, G.: A survey of multimedia streaming in wireless sensor networks, In IEEE Communications Surveys and Tutorials, vol. 10, no. 4, 2008.

[Pudlewski, 10] Pudlewski, S., Melodia, T.: On the Performance of Compressive Video Streaming for Wireless Multimedia Sensor Networks, in Proc. IEE Int'l Conf. on Commun., pp. 1-5, Cape Town, 2010

[Park, 06] Park, J.S., Song, H.-J: Selective intra prediction mode decision for H.264/AVC encoders. In Proceedings of World Academy of Science, Engineering, and Technology, Vol. 13, pp. 51-55, May 2006.

[Seema, 11] Seema, A., Reisslein, M.: Towards Efficient Wireless Video Sensor Networks: A Survey of Existing Node Architectures and Proposal for A Flexi-WVSNP Design, In IEEE Communications Surveys and Tutorials, Vol. 13, No. 3, pp. 462-486, 2011

[Yang, 04] Yang, C.-L., Po, L.-M., Lam, W.-H.: A fast H.264 intra prediction algorithm using macroblock properties. In Proc. of IEEE ICIP-2004, Vol. 1, pp. 461-464, Oct. 2004

[Zhang, 10] Daqiang Zhang, Minyi Guo, Jingyu Zhou, Dazhou Kang and Jiannong Cao, Context Reasoning Using Extended Evidence Theory in Pervasive Computing Environments, Future Generation Computer Systems (FGCS), vol. 26, pp. 207 - 216, 2010.

[Zhang, 11] Daqiang Zhang, Jingyu Zhou, Minyi Guo and Jiannong Cao, TASA: Tag-Free Activity Sensing Using RFID Tag Arrays, IEEE Transactions on Parallel and Distributed Systems (TPDS), vol. 22, pp. 558 – 570, April, 2011.