A Fast Inter Mode Decision Algorithm in H.264/AVC for IPTV Broadcasting Services

Geun-Yong Kim*, Bin-Yeong Yoon*, and Yo-Sung Ho*
*Electronics and Telecommunications Research Institute, 161 Gajeong-Dong, Yuseong-Gu, Daejeon, 305-700, Korea
*Gwangju Institute of Science and Technology, 1 Oryong-Dong, Buk-Gu, Gwangju, 500-712, Korea

ABSTRACT

The new video coding standard H.264/AVC employs the rate-distortion optimization (RDO) method for choosing the best coding mode. However, since it increases the encoder complexity tremendously, it is not suitable for real-time applications, such as IPTV broadcasting services. Therefore we need a fast mode decision algorithm to reduce its encoding time. In this paper, we propose a fast mode decision algorithm considering quantization parameter (QP) because we have noticed that the frequency of best modes depends on QP. In order to consider these characteristics, we use the coded block pattern (CBP) that has “0” value when all quantized discrete cosine transform (DCT) coefficients are zero. We also use both the early SKIP mode and early 16x16 mode decisions. Experimental results show that the proposed algorithm reduces the encoding time by 74.6% for the baseline profile and 72.8% for the main profile, compared to the H.264/AVC reference software.

Keywords: H.264/AVC, fast mode decision, quantization parameter, IPTV

1. INTRODUCTION

The international video coding standard H.264/AVC is the latest version in a sequence of the video coding standards, such as H.261, MPEG-1, MPEG-2, H.263, and MPEG-4 part 2 [1]. H.264/AVC provides enhanced video compression performance in view of interactive applications like video telephony requiring a low latency system and non-interactive applications like storage, broadcast, and streaming of standard definition TV where the focus is on high coding efficiency. Recently, H.264/AVC has been used for IPTV broadcasting services.

While H.264/AVC still uses the concept of block-based motion compensation, it provides some significant changes which furnish better coding efficiency than previous standards. In order to reduce the block-artifacts, an adaptive deblocking filter is used in the prediction loop. The deblocked macroblock is stored in the memory and can be used to predict future macroblocks. Whereas the memory contains one video frames in previous standards, H.264/AVC allows storing multiple video frames in the memory. A prediction scheme is used also in intra mode that uses the image signal of already transmitted macroblocks of the same image in order to predict the block to code. The discrete cosine transform (DCT) used in former standards is replaced by an integer transform (4x4).

The coding tools of H.264/AVC allow for bit savings of about 50% compared to previous video coding standards like MPEG-4 part 2 simple profile and H.263+ for a wide range of bit rates and resolutions [2]. However, these savings come at the price of an increased complexity. The encoder complexity depends largely on the algorithms for motion estimation as well as for the rate-constrained encoder control for mode decision and motion estimation. H.264/AVC selects the best mode by reflecting distortion between the current and reference blocks as well as encoding bits. The problem arises from the rate-distortion optimization (RDO) function, since the H.264/AVC reference software performs motion estimation for all possible modes and all encoding processes to obtain encoding bits. It causes a large amount of encoding time and is not suitable for real-time applications.

In this paper, we propose a fast mode decision algorithm considering the quantization parameter (QP). We have found that the frequency of best modes depends on QP. In order to consider these characteristics, we use CBP which has “0” value when all quantized DCT coefficients are zero. We also use early SKIP mode decision and early 16x16 mode decision to avoid time consumption for selecting the best mode in P8x8.
2. MODE DECISION OF H.264/AVC

There are seven different block sizes (16×16, 16×8, 8×16, 8×8, 8×4, 4×8, and 4×4 blocks) in H.264/AVC, which are used for inter modes as shown in Fig. 1. The available coding modes for the macroblock in a I-slice include: Intra4×4 and Intra16×16 predictions for luminance samples and Intra8×8 prediction for chrominance samples. The SKIP mode represents the case where the block size is 16×16, but no motion or residual information is coded. In the P8×8 mode, each 8×8 block can be divided into smaller blocks, such as 8×8, 8×4, 4×8, and 4×4 blocks. Intra4×4 and Intra16×16 modes, and the other inter modes except for the SKIP, require the motion estimation operation.

2.1 Rate-distortion optimization method

For each macroblock, the H.264/AVC reference software tries all possible inter modes. Motion estimation and RDO calculation are performed to find the best mode in each macroblock. The motion vector (MV) and the reference frame are selected to minimize the motion cost:

\[ J_{\text{motion}}(MV, REF | \lambda_{\text{motion}}) = SAD(s, r(MV, REF)) + \lambda_{\text{motion}} \cdot R(MV, REF) \]  

(1)

where \( MV \) is the motion vector, \( REF \) denotes the reference picture, and \( \lambda_{\text{motion}} \) is the Lagrangian multiplier which depends on QP. \( R(MV, REF) \) represents the bits used for coding motion vectors and the reference picture. \( s \) and \( r \) indicate the current and reference blocks, respectively. \( R(MV, REF) \) value is computed by the lookup table. \( SAD \) represents the sum of absolute differences defined by

\[ SAD(s, r(MV, REF)) = \sum_{x \in H, y \in V} |s(x, y) - r(x - m_x, y - m_y)| \]  

(2)

with the motion vector \((m_x, m_y)\), the horizontal block size \(H\), and the vertical block size \(V\).

The following Lagrangian function is used (a) to determine the sub-block mode for the P8×8 mode, (b) to decide the prediction mode in the intra mode decision, and (c) to select the best mode.

\[ J_{\text{mode}}(s, r, MODE | \lambda_{\text{mode}}) = SSD(s, r, MODE) + \lambda_{\text{mode}} \cdot R(s, r, MODE) \]  

(3)

where \( \lambda_{\text{mode}} \) is \( \lambda_{\text{motion}} \), and \( MODE \) can be a sub-block mode, a prediction mode, or the best macroblock mode. \( MODE \) indicates a mode out of ten possible macroblock modes: SKIP, 16×16, 16×8, 8×16, 8×8, 8×4, 4×8, 4×4, Intra4×4, and Intra16×16. \( R(s, r, MODE) \) is the number of bits associated with choosing \( MODE \) including the bits for the macroblock header, motion vectors, and all DCT coefficients. In the H.264/AVC reference software, this value is calculated for all possible modes, which means all encoding processes are performed. \( SSD \) denotes the sum of square differences between the original and reference blocks.

\[ SSD(s, r, MODE) = \sum_{x \in H, y \in V} (s(x, y) - r(x - m_x, y - m_y))^2 \]  

(4)

with the motion vector \((m_x, m_y)\), the horizontal block size \(H\), and the vertical block size \(V\).
2.2 Computational complexity of H.264/AVC

Figure 2 shows the run-time percentages of several major functional modules [3]. As can be seen, transform for cost generation and mode decision take the largest portion of computation and intra predictor generation is the second. In mode decision process, to calculate actual bits in Eq. (3), residue of each mode is processed by DCT/Hadamard transform, quantization, inverse quantization, inverse DCT/Hadamard transform, and entropy coding. Exp-Golomb VLC and CAVLC are bit-level processing with complex controlling, however their computational load is not large. The run-time percentage of CAVLC will change according to different QP values. Thus, in order to reduce the complexity of H.264/AVC encoder, we have to consider a fast mode decision algorithm.

![Diagram showing run-time percentage of several modules](image)

Fig. 2. Run-time percentage of several modules

2.3 Mode decision procedure of H.264/AVC

Figure 3 shows the mode decision procedure in the H.264/AVC reference software [4]. As shown in Fig. 3, the mode decision procedure includes inter mode and intra mode decision.

![Diagram showing mode decision procedure](image)

Fig. 3. Mode decision procedure of H.264/AVC
In general sequences, most modes are inter modes, since temporal correlation is much higher than spatial correlation. Thus, developing fast mode decision algorithm depends on how much fast we choose candidate modes for the best inter mode. From a point of view of complexity, the sub-block mode decision for P8×8 mode occupies much more time than 16×16, 16×8, and 8×16, since we have to perform a motion estimation and reference frame selection process for each sub-block in order to determine the P8×8 sub-block mode. Therefore, if we skip P8×8 sub-block mode decision efficiently, we can reduce the encoding time. Generally, P8×8 mode is not likely to be the best mode in the background or the smooth regions of the image, which provides a clue for development of a fast mode decision algorithm.

3. PREVIOUS WORKS

As described above, a mode decision procedure is the bottleneck of H.264/AVC encoder. Recently, many fast mode decision algorithms have been proposed. Among them, we examine two proposed fast inter mode decision algorithms.

3.1 Fast inter mode decision algorithm of JM reference software (FMDRS)

Accurate detection of the SKIP mode type without performing the computationally intensive mode decision would be highly desirable for computational speed-ups. The computational gains for predictive SKIP mode schemes are indeed content dependent, since slow motion sequences or sequences with uniform motion fields tend to have more skipped macroblocks. Recently, the input document to JVT from Jeon et al. has been accepted for the fast inter mode method which is adopted in JM reference software 9.4 or later version [5].

In the FMDRS, a block can be predicted as having SKIP mode in a P-slice when the following set of four conditions is satisfied:

(1) The best motion compensation block size is 16×16.

(2) The reference frame is the previous frame.

(3) The best motion vector is the predicted motion vector.

(4) The transform coefficients of the 16×16 block size are all quantized to zero.

If this set of four conditions is satisfied, the best mode becomes SKIP mode and the remaining mode decision processes are terminated. In FMDRS, the conditions for SKIP mode are checked just from the result of J_{mode}(16×16). Figure 4 depicts the flowchart of FMDRS.

![Flowchart of FMDRS](image-url)

Fig. 4. Flowchart of FMDRS
3.2 Fast inter mode decision algorithm using neighboring information (FMDNI)

This algorithm has been proposed by C. Grecos and M. Y. Yang [6]. It initially exploits neighborhood information jointly with a set of SKIP mode conditions for enhanced SKIP mode decision. It subsequently performs inter mode decision for the remaining macroblocks by using a gentle set of smoothness constraints.

The main idea is that if the macroblocks on the top and left of the one to be encoded in the current picture and the macroblocks on the right and bottom of the co-located macroblock in the reference picture are all in SKIP mode, then this mode pattern can be a good indication that the current macroblock can be skipped. The elements of its spatio-temporal predictor are shown in Fig. 5. Co is the collocated macroblock in the reference picture, CoR is the macroblock to the right of the collocated, CoB is the one to its bottom, C denotes the macroblock to be mode predicted, L is the one to its left, and T is the one on its top in the current picture. To strengthen the accuracy of the SKIP mode prediction, the extra condition is applied; SAD between the current macroblock and its co-located one should be less than the average SAD among skipped macroblocks in the reference picture and their collocated predictors. These are the first SKIP conditions in FMDNI and the second SKIP conditions follow SKIP conditions in FMDRS.

![Fig. 5. Spatio-temporal predictor for SKIP mode decision](image)

![Fig. 6. Flowchart of FMDNI](image)

After SKIP mode decision, the following three smoothness conditions are applied for the remaining macroblocks.
(1) The $J_{\text{mode}16 \times 16}$ cost of the current macroblock should be less than the average $J_{\text{mode}16 \times 16}$ cost of the macroblocks in the reference frame.

(2) The co-located macroblock in the reference frame should be of SKIP or $16 \times 16$ mode.

(3) The SAD between the current and the co-located macroblock should be less than the average SAD among the SKIP mode macroblocks in the previous frame and their collocated predictors.

Figure 6 depicts the flowchart of FMDNI.

## 4. PROPOSED FAST INTER MODE DECISION ALGORITHM

We propose a fast inter mode decision algorithm considering QP for H.264/AVC (FMDQP). FMDRS examines four conditions for the early SKIP mode decision using only $J_{\text{mode}16 \times 16}$. In FMDRS, SKIP mode is determined with $16 \times 16$ motion vector, reference frame, and CBP value. Though FMDRS provides fast and accurate SKIP mode decision method, it does not consider P8×8 mode at early stage that consumes large part of the encoding time.

### 4.1 Variation of mode distribution according to QP

In order to determine the best mode, H.264/AVC uses RDO method with Eq. (3), where $J_{\text{mode}}$ cost consists of the distortion (SSID) and rate term (R). In Eq. (3), $\lambda$ is the weighting factor representing how much the rate term affects $J_{\text{mode}}$. The Lagrangian multiplier $\lambda$ is given by

$$\lambda_{\text{mode}} = 0.85 \times 2^{0.6/3}$$

(5)

If QP is larger, $\lambda$ will be larger. This result makes the rate term more important, whereas the distortion term less important since the distortion will be small due to large QP in DCT and IDCT processes. It is the reason that the frequency of SKIP and $16 \times 16$ modes increases at large QPs. Figure 7 shows the change of mode distribution in case of QP 28 and 40 for FOREMAN sequence in QCIF format.

![Mode distribution according to QP](image)

**Fig. 7. Mode distribution according to QP**

Our proposed algorithm uses the CBP value based on these characteristics. CBP represents that the number and location of 8×8 block with non-zero coefficient in a macroblock. If a CBP value is zero, it denotes that there are no prediction residues to be coded. Therefore, we do not need to consider small block size. In our algorithm, we decide whether to consider P8×8 mode depending on the CBP value of $16 \times 16$ mode.

### 4.2 Observation of $J_{\text{motion}}$ and $J_{\text{mode}}$

The H.264/AVC reference software includes the sub-optimal mode decision method, where the best mode is the mode with minimum $J_{\text{motion}}$. The result of the sub-optimal mode decision is worse than that of the rate-distortion method in PSNR value and bit rates, since it does not count the actual coding bits. However, this observation provides a clue for the fast mode decision.

Figure 8 tells the correlation between the sub-optimal mode by $J_{\text{motion}}$ and the optimal mode by $J_{\text{mode}}$. In the experiment, 50 frames for each test sequence are used with QP 28. {0, 1, 2, 3, 8, 9, 10} denotes {SKIP, $16 \times 16$, $16 \times 8$, $8 \times 16$, $8 \times 8$,}
Intra4×4, Intra16×16}. It is found that the SKIP mode or 16×16 mode is the best mode for macroblocks in the background or smooth regions of the image. On the other hand, P8×8 mode is the favored choice for macroblocks that are on the boundary of objects or in the fast moving regions of the image. Thus, if a macroblock is in the background or smooth region in the image, J_{motion} cost will be small, which means we do not need to check J_{motion} or J_{mode} cost for small blocks. However, if a macroblock is in the boundary of objects or the fast moving regions of the image, J_{motion} cost will be large, thus we have to check J_{mode} cost for small blocks such as 8×8 and 4×4. From Fig. 8, when the best mode by sub-optimal mode decision is 16×16, the best mode by RDO method is likely to be SKIP or 16×16. Otherwise, if the sub-optimal best mode is P8×8, the best mode is not likely to be SKIP.

![Graphs showing mode distribution](image)

Fig. 8. The correlation between J_{motion} and J_{mode} for FOREMAN sequence

4.3 Implementation of proposed algorithm

An early SKIP mode decision is essential part of fast mode decision algorithm. For our proposed fast mode decision algorithm, we borrow the early SKIP mode decision scheme of FMDRS since it provides fast and accurate scheme. For more details, after performing motion estimation of 16×16 block, calculating J_{mode(16×16)} and finding SKIP motion vector, we determine the best mode as SKIP mode at early stage when the following three conditions are satisfied:

1. The reference frame is the previous frame.
2. The best motion vector of 16×16 mode is the same as that of SKIP mode.
3. The CBP value is zero.

Even though SKIP conditions are not satisfied, a macroblock could be in the smooth region or part of the object moving the same direction. Fast decision of 16×16 mode should be considered at early stage to avoid time consumption when QP is large or a macroblock is in the homogeneous region. In general, a large part of image is coded as the 16×16 mode hence we build the early 16×16 mode decision strategy by the following conditions:

1. The sub-optimal mode is 16×16 among 16×16, 16×8, and 8×16.
2. CBP (16×16) is zero.
3. SKIP motion vector is the same as 16×16 motion vector.
If the above three conditions are satisfied, we skip the procedure of sub-block mode decision for P8×8. We also inactivate 16×8, 8×16, and P8×8 modes for the best mode decision. Thus, RDO calculations are not required for these modes, that is, inter mode is determined as the 16×16 mode at early stage. This strategy could be thought as the homogeneous detection method without overhead. Even though a macroblock is not determined as 16×16 mode, if the sub-optimal best mode is 16×16 and the CBP value of 16×16 is zero, we do not include P8×8 mode for the best mode decision process.

Figure 9 shows the structure of FMDQP. As can be seen, we determine the motion vector and reference frame for 16×16 at first and calculate \( J_{\text{model}}(16\times16) \) and we find the SKIP motion vector. The SKIP motion vector is predicted by median filter of the motion vectors for the neighboring three blocks; left, top, top-right. If some adjacent block does not exist, it follows the rule specified in the H.264/AVC draft specification [1]. The next step is to examine the SKIP conditions. If a macroblock satisfies them, we terminate the mode decision process and the best mode is determined as SKIP. This makes it possible to skip mode decision and motion estimation processes for 16×8, 8×16, and P8×8. If the sub-optimal best mode is 16×16 among 16×16, 16×8, 8×16 modes, and the CBP value of 16×16 is zero, we assume that the current macroblock is in the smooth region of the image or in part of the object moving the same direction.

Even if not, if QP is large, we do not need to split a macroblock into small blocks since the CBP value tends to be zero. Hence, we skip the sub-block mode decision procedure including motion estimation for P8×8. If motion vectors of SKIP and 16×16 modes are not the same, we perform the mode decision process for the remaining modes. The early 16×16 mode decision is applied when the motion vectors of SKIP and 16×16 modes are the same. Thus, we decide the best inter mode as 16×16 and compare it with intra modes. The early SKIP mode decision and the early 16×16 mode decision are the most important part of our proposed algorithm since SKIP and 16×16 are the popular modes in general sequence.

![Flowchart of FMDQP](image)

**5. EXPERIMENTAL RESULTS**

We have implemented our proposed algorithm into JM the reference software 9.5. In our experiments, we have used a variety of video sequences in QCIF format adopted as test sequences in MPEG standard. The experiment was performed on a Pentium-IV (2.80 GHz). Test sequences are coded with the Baseline and Main profiles. The simulation conditions are shown in Table 1 [7]. For performance comparison among FMDRS, FMDNI and, our algorithm, we have used the Bjontegaard delta bit rates and Bjontegaard delta PSNR (BDBR) [8]. The experimental results are shown in Table 2 and Table 3. All comparisons are with respect to results of the original H.264/AVC without fast motion estimation and fast mode decisions.
Table 1. Simulation conditions

<table>
<thead>
<tr>
<th>Profile</th>
<th>Baseline and Main Profiles</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hadamard Transform</td>
<td>Used</td>
</tr>
<tr>
<td>RDO Mode</td>
<td>Fast High Complexity Mode</td>
</tr>
<tr>
<td>Search Range</td>
<td>±16</td>
</tr>
<tr>
<td>Reference Frames</td>
<td>5</td>
</tr>
<tr>
<td>Quantization Parameters</td>
<td>28, 32, 36, 40</td>
</tr>
<tr>
<td>Fast Motion Estimation Algorithms</td>
<td>UMHexagonS, CBFPS [9]</td>
</tr>
</tbody>
</table>

Table 2. Performance comparison for Baseline profile

<table>
<thead>
<tr>
<th>Sequences</th>
<th>FMDRS</th>
<th>FMDNI</th>
<th>Proposed Algorithm</th>
</tr>
</thead>
<tbody>
<tr>
<td>AKIYO</td>
<td>-0.04</td>
<td>-0.11</td>
<td>80.0</td>
</tr>
<tr>
<td>NEWS</td>
<td>-0.03</td>
<td>-0.22</td>
<td>79.8</td>
</tr>
<tr>
<td>SILENT</td>
<td>-0.02</td>
<td>-0.19</td>
<td>74.0</td>
</tr>
<tr>
<td>CONTAINER</td>
<td>-0.03</td>
<td>-0.2</td>
<td>83.3</td>
</tr>
<tr>
<td>COASTGUARD</td>
<td>-0.02</td>
<td>-0.25</td>
<td>57.7</td>
</tr>
<tr>
<td>CARPHONE</td>
<td>-0.05</td>
<td>-0.09</td>
<td>65.0</td>
</tr>
<tr>
<td>FOREMAN</td>
<td>-0.11</td>
<td>-0.6</td>
<td>60.1</td>
</tr>
<tr>
<td>MOBILE</td>
<td>-0.02</td>
<td>-0.45</td>
<td>49.3</td>
</tr>
<tr>
<td><strong>Average</strong></td>
<td><strong>-0.04</strong></td>
<td><strong>-0.27</strong></td>
<td><strong>68.7</strong></td>
</tr>
</tbody>
</table>

Table 3. Performance comparison for Main profile

<table>
<thead>
<tr>
<th>Sequences</th>
<th>FMDRS</th>
<th>FMDNI</th>
<th>Proposed Algorithm</th>
</tr>
</thead>
<tbody>
<tr>
<td>AKIYO</td>
<td>-0.09</td>
<td>-0.15</td>
<td>79.3</td>
</tr>
<tr>
<td>NEWS</td>
<td>-0.05</td>
<td>-0.21</td>
<td>78.3</td>
</tr>
<tr>
<td>SILENT</td>
<td>-0.02</td>
<td>-0.13</td>
<td>72.2</td>
</tr>
<tr>
<td>CONTAINER</td>
<td>-0.03</td>
<td>-0.2</td>
<td>81.2</td>
</tr>
<tr>
<td>COASTGUARD</td>
<td>-0.03</td>
<td>-0.47</td>
<td>56.5</td>
</tr>
<tr>
<td>CARPHONE</td>
<td>-0.09</td>
<td>-0.55</td>
<td>63.1</td>
</tr>
<tr>
<td>FOREMAN</td>
<td>-0.09</td>
<td>-0.62</td>
<td>58.3</td>
</tr>
<tr>
<td>MOBILE</td>
<td>-0.02</td>
<td>-0.14</td>
<td>46.8</td>
</tr>
<tr>
<td><strong>Average</strong></td>
<td><strong>-0.05</strong></td>
<td><strong>-0.31</strong></td>
<td><strong>67.0</strong></td>
</tr>
</tbody>
</table>

Where,

 Avg.\text{PSNR} = \frac{\text{PSNR}(28) + \text{PSNR}(32) + \text{PSNR}(36) + \text{PSNR}(40)}{4} \quad [dB] \quad (6)
\[
\text{Avg.Bitrate} = \frac{\text{Bitrate}(28) + \text{Bitrate}(32) + \text{Bitrate}(36) + \text{Bitrate}(40)}{4} \quad [\text{kbit/s}]
\]

\[
\Delta T = \frac{\text{Time(\text{reference})} - \text{Time(\text{proposed})}}{\text{Time(\text{reference})}} \times 100 \quad [%]
\]

\[
\text{Avg.} \Delta T = \frac{\Delta T(28) + \Delta T(32) + \Delta T(36) + \Delta T(40)}{4} \quad [%]
\]

### 6. CONCLUSIONS

H.264/AVC provides high compression efficiency compared to previous video coding standards, such as MPEG-4 and H.263, mainly by supporting variable block-size macroblock modes. H.264/AVC has seven different block sizes: 16×16, 16×8, 8×16, 8×8, 8×4, 4×8, and 4×4. A motion estimation process is performed for every possible mode, thus it requires much higher computational complexity than the other standards.

In order to reduce the complexity, FMDRS sets four conditions for the early SKIP mode decision using only \( J_{\text{mode}(16\times16)} \). It decides SKIP mode based on the 16×16 motion vector, reference frame, and CBP. Though FMDRS provides the fast and accurate SKIP mode decision, it does not consider P8×8 mode at early stage. FMDNI uses neighborhood information jointly with a set of SKIP mode conditions for enhanced SKIP mode decision to overcome FMDRS, however the P8×8 mode is not treated at early stage and it does not consider QP.

In this paper, we have proposed a fast mode decision algorithm considering QP. We have focused on the frequency of the best mode according to QP. In order to consider these characteristics, we check CBP which reflects the number of quantized DCT coefficients. We also consider the early SKIP mode decision and develop the early 16×16 mode decision using motion vectors of 16×16 and SKIP modes, reference frame, and the sub-optimal best mode.

Experimental results show that the proposed algorithm reduces encoding time by 74.6% for the Baseline profile and 72.8% for the Main profile compared to H.264/AVC reference software. We have also verified that our algorithm maintains its performance regardless of entropy coding techniques, such as CABAC and CAVLC.

### REFERENCES