FAST INTRA-MODE DECISION IN H.264 USING INTERBLOCK CORRELATION

Chou-Chen Wang, Tsung-Shien Chen and Chi-Wei Tung

Department of Electronic Engineering, I-Shou University, Kaohsiung, Taiwan
E-mail: chchwang@isu.edu.tw

ABSTRACT
In this paper, a fast intra mode decision algorithm for H.264 that exploits the interblock correlation in the intra-mode domain is proposed to reduce computational complexity. Four modes of neighboring coded macroblocks/blocks are considered as the good candidate intra modes of the current block. Experimental results show that the proposed method can efficiently save the computation cost with little degradation in the rate-distortion performance.

Index Terms— video coding, video codecs, H.264

1. INTRODUCTION
H.264 is the ITU-T’s newest video coding recommendation, which is also known as MPEG-4 Advanced Video Coding (AVC) [1]. The H.264 standard can achieve much higher coding efficiency than the previous standards such as MPEG-1/2/4 and H.261/H.263. This is mainly due to the fact that the H.264 encoder employs more complicated approaches in the coding procedure. One important approach is the technique of intra-prediction in the spatial domain. The H.264 standard exploits the directional spatial correlation between adjacent macroblocks (MBs) or blocks for intra prediction. In other words, the current MB/block is predicted by adjacent pixels in the upper and the left MBs/blocks that are previously decoded. H.264 offers a rich set of prediction patterns for intra prediction, i.e., nine prediction modes for 4×4 luma blocks, four modes for 16×16 luma MBs, and four modes for 8×8 chroma blocks, respectively. Each mode has its own direction of prediction and the predicted samples are obtained from a weighted average of decoded values of neighborhood MBs/blocks [2]. Figure 1 shows prediction samples and nine prediction modes for each 4×4 luma block. It can be seen that 4×4 block prediction is conducted for samples a-p of a block using samples A-Q. There are eight prediction directions in total and one DC prediction for 4×4 block prediction [1-2].

To take the full advantage of these modes, the H.264 encoder can select the best mode by using the rate-distortion optimization (RDO) calculations. The RDO mode decision exhaustively searches the best mode for each MB which produces the minimum rate-distortion cost (RDcost) given by

\[ J(s,c,mode) = \text{SSD}(s,c,mode) \cdot QP \]

where \( QP \) is the MB quantization parameter, \( \lambda_{\text{mode}} \) is the Lagrange multiplier for mode decision, and \( \text{SSD} \) represents the sum of the squared differences between the original block \( s \) and its reconstruction \( c \) and \( \text{mode} \) represents one of the potential prediction modes. According to the RDO procedure of intra prediction, the number of mode combinations for luma and chroma components in an MB is \( \sum_{\text{N},N_{8\times8},N_{4\times4},N_{16x16}} \), where \( N_{8\times8}, N_{4\times4}, \) and \( N_{16x16} \) represent the number of modes for 8×8 chroma blocks, 4×4 luma blocks and 16×16 luma MBs, respectively. It means that, for an MB, it has to perform 592 different RDO calculations before a best mode is determined. As a result, the complexity of the intra-mode decision is extremely high, which makes it difficult to achieve real-time implementations. To reduce the number of RDO computations, many fast mode decision methods have been proposed [3-5]. F. Pan, et al. proposed [3] a fast intra mode decision method based on analysis of edge direction histogram within the block so as to reduce the number of probable modes. J. Kim and J. Jeong proposed [4] a modified version based on Pan’s method [3] using simple directional masks and adjacent mode information to further...
speed up RDO procedure. Although Pan’s and Kim’s algorithms have reduced much complexity of intra prediction, they need extra pre-processing time to detect edge and analyze edge direction histogram. Therefore, the effects of both fast mode decision algorithms are reduced.

2. INTRA PREDICTION IN H.264

2.1 Best mode selection using RDO

As studied in the H.264 test model codec [6], the intra prediction procedures of luma and chroma components \( (Y,C_b) \) using RDO can be described as follows:

Step 1 Generate an \( 8 \times 8 \) predicted chroma block according to a mode.

Step 2 Determine the best intra mode for a \( 16 \times 16 \) MB among 4 modes. Code the chroma components with the given mode and compute the rate distortion of the MB for \( Y,C_b \) components \( \text{RDCost}_{16 \times 16} \).

Step 3 Select the 16 best intra modes for sixteen \( 4 \times 4 \) luma blocks among 9 modes. Code the chroma components with the given modes and compute the rate distortion of the MB for \( Y,C_b \) components \( \text{RDCost}_{4 \times 4} \).

Step 4 If \( \text{RDCost}_{16 \times 16} > \text{RDCost}_{4 \times 4} \), the block type \( 4 \times 4 \) is selected, otherwise the \( 16 \times 16 \) block type is selected in the given chroma mode. And the minimum cost is saved as \( \text{RDCost} \).

Step 5 Repeat step 1 to 4 for all chroma intra prediction modes, and choose the one with minimum \( \text{RDCost} \).

2.2 RDO Intra-mode map

H.264 is a block-based coding scheme, the frame is encoded block by block in a raster scan order, i.e., from the left to right and top to bottom. For a luma MB in an I-slice, RDO exhaustively searches the combinations of the predefined 13 intra modes to produce the best mode for this MB. Figure 1 shows part of the RDO intra-mode map of an I-frame (Hall-monitor video sequence) conducted by the JM 10.1 [6] with RDO procedure. Each point (location) in the two intra-mode maps corresponds to a \( 4 \times 4 \) block and a \( 16 \times 16 \) MB, respectively. Since MBs/blocks are highly correlated, many MBs/blocks in I-frame correspond to the same modes. In other words, many points in the map have same mode, as shown in Fig. 2. Therefore, the optimal modes of the neighboring MBs/blocks may be the good candidate modes for the current block. In this work, the four causal neighboring modes of the current MBs/blocks shown in Fig. 3 are first chosen to perform RDO computations. The modes A, B, C, and D represent the four candidate modes, which have been encoded by previous MB/blocks. A regular search scheme is employed to search for the causal neighboring modes.

3. PROPOSED FAST INTRA MODE DECISION

To determine whether a candidate mode is good enough for the current MB/block, we compute the \( \text{RDCost} \) by (1). After the candidate mode (one of the modes A, B, C and D) is found, we check it is good enough by comparing its \( \text{RDCost} \) with a threshold. If it is less than the threshold, the candidate is good enough for the current MB/block. Otherwise, it implies that the MB/block correlation is low and the other modes are needed to find the optimal prediction mode for the current MB/block. There may exist some repetition modes in the neighboring of the search path, as shown in Fig. 2. Due to this repetition, the proposed
is obtained from a lookup, while RDO procedure will yield redundant computation and thereby decrease the coding efficiency. Therefore, before RDO computation, we should determine whether the candidate mode is a repetition. If it is a repetition mode, the RDO computation for this candidate is skipped. The new algorithm for intra mode decision in H.264 can be summarized as follows:

**Intra-mode decision for 8×8 chroma blocks:**
Step 1 Check whether the first candidate mode \(A_{8,8}\) is ‘good’. If it is true, the mode is selected. Otherwise go to step 2.
Step 2 Repeat step 1 for the next priority neighboring block until a ‘good’ candidate mode is found or all neighboring modes are examined. If no good candidate is found, go to step 3.
Step 3 Perform the other modes to find the optimal mode.

**Intra-mode decision for 16×16 luma MBs:**
Step 1 Check whether the mode \(A_{16×16}\) is ‘good’. If it is true, the mode is selected. Otherwise go to step 2.
Step 2 Repeat step 1 for the next priority neighboring MB until a ‘good’ candidate mode is found or all neighboring modes are examined. If no good candidate is found, go to step 3.
Step 3 Perform the other modes to find the optimal mode.

**Intra-mode decision for 4×4 luma blocks:**
Step 1 Check whether the mode \(A_{4×4}\) is ‘good’. If it is true, the mode is selected and proceeds to the next block in the same manner. Otherwise go to step 2.
Step 2 Repeat step 1 for the next priority neighboring block until a ‘good’ candidate mode is found or all neighboring modes are examined. If no good candidate is found, go to step 3.
Step 3 Perform the other modes to find the optimal mode, and then go back to step 1 for the next block.

### 4. EXPERIMENTAL RESULTS

The proposed mode decision algorithm was tested using three QCIF video sequences (Hall-monitor, News and Container). For each sequence, 300 frames are encoded with I-frame only. The frame rate is 30 frames/s. The experiment was carried out on the JVT reference software JM 10.1 [6]. Context adaptive binary arithmetic coding (CABAC) was adopted as the entropy coding method. The Hadamard transform was enabled. Experiments were conducted for four quantization parameters \(QP = 28, 32, 36 \text{ and } 40\).

How to determine the thresholds is the key issue to the proposed method in H.264. It is clear that the larger the thresholds are, the more search modes can be skipped and the encoding complexity can be further reduced. However, more modes are incorrectly selected at the same time, which results in more significant loss in image quality and more bits used in encoding prediction error signals. Table I shows the relationship between rate-distortion (R-D) performance and complexity for “Hall-monitor” sequence using \(QP = 32\).

In practice, preventing loss in video quality may be more important than a minor increase in complexity. Therefore, we regard video quality as the guide in determining thresholds. In this work, the average loss of video quality is limited under 0.1 dB. We have conducted several experiments with different candidate sets of thresholds based on different degradation of video quality for some video sequences, and then select an appropriate set which provides a good tradeoff between R-D performance and computational complexity in practice. The set of the threshold \((TH_{8×8}, TH_{16×16}, TH_{4×4})\) is an important parameter which affects the R-D curve and complexity of our method. Furthermore, experiments show that different QP values result in different thresholds. This is due to the fact that the RDCost value is related to QP. In (1), \(J\) is directly related to \(\lambda\), while \(\lambda\) is obtained from a lookup table indexed by QP. Therefore, we need to adjust the thresholds according to the value of QP. In our method, we select the threshold set for \(QP = 32\) to be the principal thresholds, and then the threshold sets for other QP values can be set as follows:

\[
TH_{8×8} = 35000 + (\lambda_{QP} - \lambda_{32}) \times 100 \\
TH_{16×16} = 15000 + (\lambda_{QP} - \lambda_{32}) \times 3 \\
TH_{4×4} = 1200 + (\lambda_{QP} - \lambda_{32}) \times 6
\]  

The performance of the proposed algorithm is compared with Pan’s fast algorithm [3] and JM 10.1 with RDO procedure [6] in terms of the average PSNR of luma component (PSNR\(Y\)), bitrate and encoding time. All the concerning methods are implemented using Visual C++ and the Pentium IV 1.3G PC. The test results are shown in Tables II-III. From Table II, we can see that the encoding time of the proposed algorithm is obviously less than JM10.1 and Pan’s method in all kinds of QP with little degradation in the R-D performance. Additionally, as compared with Pan’s method, our approach reduces the encoding time by 24.3% to 43.3%. This is because there is not any on-line pre-processing time that is needed by our method.

The computational complexity comparisons, the number of candidate modes selected for RDO computation, are summarized in Table III. From Table III, we may find that our method has less intra modes than Pan’s method with close to R-D performance. Taking the “Hall-monitor” under \(QP = 32\) as an example, the encoder with the proposed method carries out only \(1.54 \times (3.68 \times 16 + 1.82) = 93.5\) RDO computations on the average, which are much less than that of Pan’s fast algorithm used in H.264 video coding, i.e., 132 or 198 RDO computations.
5. CONCLUSIONS

A fast intra mode decision algorithm for H.264 standard that take advantage of the correlation between MBs/blocks is proposed. The motivation is to predict the potential intra modes before conducting the rate-distortion optimized intra-prediction. Experimental results show our method is superior to Pan’s method in computational complexity due to no on-line computational overhead.

REFERENCES


Table I. Comparison of the R-D performance and complexity with different threshold sets for “Hall-monitor” using \( QP=32 \).

<table>
<thead>
<tr>
<th>Threshold sets ((TH_{8x8}, TH_{16x16}, TH_{4x4}))</th>
<th>R-D performance</th>
<th>Complexity (Average # of modes)</th>
</tr>
</thead>
<tbody>
<tr>
<td>((0, 0, 0))</td>
<td>35.03, 477.52</td>
<td>9, 4, 4</td>
</tr>
<tr>
<td>((17500,7500,600))</td>
<td>35.00, 486.94</td>
<td>4.77, 2.4, 2.57</td>
</tr>
<tr>
<td>((35000,15000,1200))</td>
<td>34.97, 496.16</td>
<td>3.68, 1.82, 1.54</td>
</tr>
<tr>
<td>((70000,30000,2400))</td>
<td>34.94, 518.54</td>
<td>2.52, 1.11, 1</td>
</tr>
<tr>
<td>((144000,60000,4800))</td>
<td>34.84, 608.76</td>
<td>1.27, 1, 1</td>
</tr>
</tbody>
</table>

Table II. Comparison of the test results in terms of the PSNR(dB), bitrate (Kbps) and encoding time (sec).

<table>
<thead>
<tr>
<th>QP</th>
<th>Methods</th>
<th>PSNR</th>
<th>Bitrate</th>
<th>Time</th>
<th>PSNR</th>
<th>Bitrate</th>
<th>Time</th>
<th>PSNR</th>
<th>Bitrate</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>28</td>
<td>JM 10.1</td>
<td>38.12</td>
<td>674.79</td>
<td>429.20</td>
<td>37.72</td>
<td>791.02</td>
<td>452.22</td>
<td>37.18</td>
<td>720.34</td>
<td>435.37</td>
</tr>
<tr>
<td></td>
<td>Pan’s</td>
<td>38.09</td>
<td>689.25</td>
<td>170.91</td>
<td>37.69</td>
<td>809.99</td>
<td>180.53</td>
<td>37.12</td>
<td>735.95</td>
<td>167.94</td>
</tr>
<tr>
<td></td>
<td>Proposed</td>
<td>38.05</td>
<td>711.03</td>
<td>96.98</td>
<td>37.69</td>
<td>859.73</td>
<td>111.74</td>
<td>37.16</td>
<td>754.43</td>
<td>108.68</td>
</tr>
<tr>
<td>32</td>
<td>JM 10.1</td>
<td>35.03</td>
<td>477.57</td>
<td>370.43</td>
<td>34.53</td>
<td>563.25</td>
<td>389.87</td>
<td>34.3</td>
<td>487.39</td>
<td>385.3</td>
</tr>
<tr>
<td></td>
<td>Pan’s</td>
<td>34.99</td>
<td>489.65</td>
<td>152.53</td>
<td>34.51</td>
<td>579</td>
<td>157.93</td>
<td>34.25</td>
<td>501.18</td>
<td>148.99</td>
</tr>
<tr>
<td></td>
<td>Proposed</td>
<td>34.97</td>
<td>496.16</td>
<td>101.02</td>
<td>34.49</td>
<td>597.75</td>
<td>112.92</td>
<td>34.27</td>
<td>506.05</td>
<td>112.66</td>
</tr>
<tr>
<td>36</td>
<td>JM 10.1</td>
<td>31.95</td>
<td>325.91</td>
<td>330.19</td>
<td>31.52</td>
<td>382.73</td>
<td>353.48</td>
<td>31.49</td>
<td>319.24</td>
<td>334.58</td>
</tr>
<tr>
<td></td>
<td>Pan’s</td>
<td>31.91</td>
<td>338.4</td>
<td>138.05</td>
<td>31.52</td>
<td>397.52</td>
<td>141.7</td>
<td>31.44</td>
<td>331.09</td>
<td>136.10</td>
</tr>
<tr>
<td></td>
<td>Proposed</td>
<td>31.88</td>
<td>338.94</td>
<td>95.01</td>
<td>31.37</td>
<td>400.96</td>
<td>104.89</td>
<td>31.46</td>
<td>330.38</td>
<td>97.67</td>
</tr>
<tr>
<td>40</td>
<td>JM 10.1</td>
<td>29.05</td>
<td>219.21</td>
<td>297.76</td>
<td>28.66</td>
<td>256.5</td>
<td>306.8</td>
<td>28.91</td>
<td>210.45</td>
<td>296.09</td>
</tr>
<tr>
<td></td>
<td>Pan’s</td>
<td>29.01</td>
<td>229.24</td>
<td>129.32</td>
<td>28.64</td>
<td>267.98</td>
<td>131.26</td>
<td>28.89</td>
<td>220.3</td>
<td>123.04</td>
</tr>
<tr>
<td></td>
<td>Proposed</td>
<td>28.95</td>
<td>227.17</td>
<td>84.41</td>
<td>28.62</td>
<td>269.12</td>
<td>92.6</td>
<td>28.83</td>
<td>218.33</td>
<td>85.00</td>
</tr>
</tbody>
</table>

Table III. Comparison of the number of RDO computations (Hall-monitor sequence).

<table>
<thead>
<tr>
<th>Block type</th>
<th>Total # of modes by JM 10.1</th>
<th># of modes selected by Pan’s</th>
<th>Average # of modes by ours ((QP=32))</th>
</tr>
</thead>
<tbody>
<tr>
<td>4x4 luma</td>
<td>9</td>
<td>4</td>
<td>3.68</td>
</tr>
<tr>
<td>16x16 luma</td>
<td>4</td>
<td>2</td>
<td>1.82</td>
</tr>
<tr>
<td>8x8 chroma</td>
<td>4</td>
<td>3 or 2</td>
<td>1.54</td>
</tr>
</tbody>
</table>