Priority Classification Based Fast Intra Mode Decision for High Efficiency Video Coding

Hao Zhang, Qiang Liu
School of Information Science and Engineering
Central South University
Changsha, Hunan 410083 China
Email: hao@csu.edu.cn

Zhan Ma
Samsung Research America
Samsung Electronics
Richardson, TX 75082 USA
Email: zhan.ma@ieee.org

Abstract—The latest High Efficiency Video Coding (HEVC) video coding standard version 1 offers 50% bit rate reduction against the H.264/AVC at the same visual quality. However, HEVC encoder complexity is tremendously increased. It is therefore important to develop efficient encoding algorithms for the success of HEVC based applications.

In this paper, we propose a priority classification based fast intra mode decision to speed up the HEVC intra encoder. Each prediction unit (PU) is given a priority label out of four based on its spatial and temporal neighbor PU information as well as the predicted PU depth. Different processing strategy will be applied to different priority class, under the assumption that more computing resource should be allocated to the high priority class since its corresponding PU has the high potential to be chosen as the optima. Experiments are performed using all the common test sequences, and results show that, the encoder complexity is significantly reduced by about 46% for All Intra configuration with BD-Rate (Bjontegaard Delta Rate) increase less than 0.9% for luma component. Meanwhile, compared with several recent works, our proposed solution demonstrates the well trade-off between the coding efficiency and complexity reduction.

I. INTRODUCTION

The latest video coding standard HEVC promises the significant performance improvement over the H.264/AVC [1], [2]. The performance improvement is mainly due to the recursive tree structure, larger block transforms, fine-grain angular intra prediction, advanced motion prediction, sample adaptive offset and etc [1], [2], [3], [4], [5]. Instead of the macroblock defined in its predecessors, recursive quad-tree structured Coding Unit (CU), Prediction Unit (PU) and Transform Unit (TU) are introduced into the HEVC to massively exploit the spatial and/or temporal correlation. CU can be split into four sub-CUs. Each CU or sub-CU can be split into multiple PUs and associated TUs. More details regarding the HEVC tools could be found in [2].

Spatial Intra prediction in the HEVC has been improved noticeably compared with the H.264/AVC with more fine-grain directional modes [5]. In HEVC, up-to 35 prediction modes are used instead of up-to 9 modes in H.264/AVC. Obviously, these fine-grain directional modes would better explore the spatial correlation and lead to higher compression ratio, while increase complexity very much by selecting the optimal intra coding block and prediction mode via rate-distortion optimization (RDO). To reduce the encoder complexity, HEVC reference software HM adopts a four-step method1, where rough mode decision (RMD) is first performed using the Hadamard cost based criteria to choose few candidates (out of 35) [6]. After then, additional most probable modes (MPM) will be added into these few candidates if they are not included yet, given the assumption that MPM has the high probability to be chosen because of the neighbor correlation. RDO is applied to these few effective candidates with the maximum TU size. In the end, for the best mode chosen from the RDO, residual quadtree based TU selection (RQT) is performed to choose the optimal TU size through the same RDO manner. Here, RDO is implemented in HEVC reference software HM [7] using the rate-distortion optimized quantization (RDOQ) in default. HM also implements the conventional RDO rather RDOQ as a trade-off between coding efficiency and encoder complexity2.

Fig. 1 pictures the main idea of such four-step method. As we can see, it avoids checking various combination of CU, PU and TU via the expensive RDO. However, the complexity of intra prediction is still very high and there is quite some room for further improvement.

Fig. 1: Intra mode decision process in HEVC reference software: Ψ is the effective candidate set after performing the Hadamard cost ranking and MPM selection; $m_{opt}$ is the best mode chosen by the RDO from Ψ for subsequent RQT.

In this paper, we propose a novel priority classification based fast intra mode decision to speed up intra prediction in the HEVC. For each PU, four priority classes are defined as HH (high-high), HL (high-low), LH (low-high) and LL (low-low). HH stands for the highest priority, LL is the lowest priority class, HL and LH are intermediate classes but HL has

---

1It was a three-step method originally proposed in [6] and later refined with four-step with improved performance by adding the most probable modes.

2Other realizations of the RDO are also possible given the implementation constraints. Here, we only consider the high-complexity RDOQ and low-complexity conventional RDO operations.
the higher priority than LH. During the encoding, each intra PU at certain depth will be assigned with a proper priority class and each class is associated with the corresponding processing strategy. Intuitively, a HH PU is considered to have the highest probability to be the optima. After examining the necessary steps (i.e., transform, quantization and entropy encoding), it will jump to the next PU directly; while a LL PU has the very small chance to be chosen, hence the processing operations can be completely skipped. To save the computational power, it is reasonable to spend the most computing time (or resource) on the HH PUs, while the least time (or resource) for LL PUs. Meanwhile, HL and LH PUs are likely to be selected, thus computing resource allocation will be between HH and LL classes. As will be shown in subsequent sections, our proposed priority classification based fast intra mode decision demonstrates the 0.9% BD-Rate increase while providing 46% encoding time reduction compared with the default intra coding using the latest HEVC reference software HM [7].

The remainder of this paper is organized as follows. Section II presents a brief literature review of the fast intra prediction algorithms for HEVC, while Section III details the proposed priority classification based scheme. Experimental results are presented in Section IV and the concluding remarks are drawn in Section V.

II. FAST INTRA MODE DECISION FOR HEVC: A REVIEW

Generally speaking, fast intra mode decision is trying to derive the subset of the candidates to avoid the brute force RDOQ based mode selection, which is basically a trade-off between the coding efficiency and encoder complexity. So far, there are already many relevant publications for HEVC. For instance, a fast RQT algorithms is proposed by Tan et al. [8], where less candidates are selected for full RDOQ while all the transform depths are checked for each candidate. Teng et al. propose a scheme to early terminate the merge or split process of RQT with fast rate and distortion predictions [9]. Meanwhile, early zero-blocks decision are also used to skip some TU levels.

Zhao et al. [10] propose to use a smaller number of candidates for each CU depth to conduct full RDOQ. Their results show negligible quality loss with 20% speed-up. They also experiment the trade-off between complexity and coding efficiency with respect to the number of effective modes. Choi et al. [11] propose to avoid CU split if SKIP mode is chosen for the current CU level. But this method can only be applied to the inter prediction. A gradient based method is developed by Jiang et al. [12], where only a few modes along the the gradient directions are checked. Tian and Goto [13] propose a content adaptive method by downsampling the LCU (largest coding unit) and analyzing texture complexity, where some CU levels are skipped according to the complexity measurement. Recently, Shen et al. [14] also develop a fast mode decision algorithm based on CU level prediction and early termination of motion estimation. Neighboring CUs’ levels are weighted as the prediction for the level of the current CU. However, this paper mainly deals with inter mode decision.

Besides, we have proposed a novel three-step fast intra prediction method in [15]. In RMD phase, the original Hadamard transform is replaced by a 2:1 downsampled Hadamard transform, followed by a gradual progressive mode search. Meanwhile, an early termination method is also included to reduce the number of modes for the subsequent RDOQ. Another piece of our work includes a set of early termination techniques for fast intra mode decision [16]. The rate-distortion (R-D) cost of four sub-CUs and sub-TUs are predicted by the Hadamard costs. The predicted cost is then compared with the R-D cost in the upper level to decide whether the CU split is necessary.

III. PRIORITY CLASSIFICATION BASED FAST INTRA MODE DECISION ALGORITHM

For each LCU, an optimal block split will be determined after going through all possible candidates. Ideally, all the computing power is wasted among non-optimal modes. However, it is quite difficult to predict the optimal block split before the LCU encoding. Alternatively, we try to predict whether current PU has the high or low probability to be selected as the optima.

A. Priority Classification

As aforementioned, four different priority classes are defined in this paper, i.e., HH, HL, LH and LL, representing the probability to be chosen as the optimal PU from the highest to the lowest level.

We are first motivated by the assumption that, current PU should have the high probability to use the same block size as its neighbors, such as upper and left PU, and its temporal co-located PU in the previous frame (which can be noted as the temporal neighbor as well). We then first use the following condition to identify the HH and HL classes, i.e.,

- HH: current PU has the same size as its upper and left PU;
- HL: current PU has the same size as its upper or left or temporal co-located PU

Apart from above scenarios, it has the many cases that current PU has the different block size as its spatial and temporal neighbors. Therefore, we first try to predict a probable depth for current PU. Towards this goal, we calculate the averaged sum of absolute difference (SAD) per pixel between current PU and its upper, left and previous temporal co-located PU using the original pixel, and derive the predicted depth of current PU via

\[
\tilde{d}_{\text{cur PU}} = \frac{a \cdot d_{\text{left PU}} + b \cdot d_{\text{upper PU}} + c \cdot d_{\text{prev PU}}}{a + b + c},
\]

where \(d\) is the actual PU depth and \(\tilde{d}\) is the predicted depth. \(a = 1/\delta_{\text{left PU}}, b = 1/\delta_{\text{upper PU}}, c = 1/\delta_{\text{prev PU}}\) with \(\delta\) indicating the averaged SAD per pixel. In our implementation, if the difference between the actual current PU depth and predicted one is less than 1 (i.e., \(|\tilde{d}_{\text{cur PU}} - d_{\text{cur PU}}| \leq 1\)), this PU is classified as the LH category. Otherwise, the PU is labeled as the LL type if all the above conditions are not satisfied. The overall classification method is summarized in Algorithm 1.
Algorithm 1 Method of priority classification for each PU

1: PriorityPUClass ← 0 //initialization
2: Derive the \( d_{\text{curPU}} \) according to Eq. (1)
3: if \( (d_{\text{curPU}} = d_{\text{leftPU}}) \) and \( (d_{\text{curPU}} = d_{\text{upperPU}}) \) then
4: PriorityPUClass ← HH
5: else if \( (d_{\text{curPU}} = d_{\text{leftPU}}) \) or \( (d_{\text{curPU}} = d_{\text{upperPU}}) \) or \( (d_{\text{curPU}} = d_{\text{prevPU}}) \) then
6: PriorityPUClass ← HL
7: else if \( |d_{\text{curPU}} - d_{\text{curPU}}| \leq 1 \) then
8: PriorityPUClass ← LH
9: else
10: PriorityPUClass ← LL
11: end if

Fig. 2: Illustration of the associated operations for each priority class.

B. Associated Processing for Each Priority Class

As shown in Fig. 1, basic operations for the intra prediction includes rough mode decision (RMD), MPM selection, RDO and RQT. In RMD, few candidates are selected based on the Hadamard cost ranking. As the default implementation in HM [6], it produces 8, 8, 3, 3, and 2 modes for squared PU sizes from \( 4 \times 4 \) to \( 64 \times 64 \), respectively. However, these modes still have redundancy. We have noticed that the mode has the high probability to be optimal one if its corresponding Hadamard cost is small. Therefore, we order the modes from the RMD process along with its Hadamard cost in ascending order, and only select few ones in the beginning for later RDO process. In this work, we use 3, 3, 2, 2, and 1 modes for PU sizes from \( 4 \times 4 \) to \( 64 \times 64 \). As see, we can further reduce the encoder complexity by using less effective candidates at different PU level. It is also noted that our findings are consistent with the work published in [10], where other settings (rather only 3, 3, 2, 2, 1) are also experimented to demonstrate the trade-off between complexity and coding efficiency.

In addition, most probable mode (MPM) scheme is also adopted in the HEVC to leverage the correlation between spatial neighbors. MPM candidates will be included into the \( \Psi \) if they are not there yet. This step could be skipped if we find that current PU belongs to the LL class.

RDO is used to choose the R-D optimal mode from the \( \Psi \) and the best TU size in RQT phase. RDO could be realized using the default RDOQ or the conventional low-complexity RDO. Therefore, for PUs (such as LH class) with small chance to be chosen as the optimal one, we propose to use the low-complexity RDO instead of the default RDOQ. Besides, for the PUs with the least probability to be the optimum, we propose to completely skip all the steps (i.e., RMD, MPM selection, RDO and RQT).

In summary, for each priority class, we define its associated operations as follows and illustrated in Fig. 2. This kind of assignment for each class is based on the assumption that we would like to allocate more computing resource to the modes which have the high potential to be selected as the optimal.

- HH: apply MPM selection, RDOQ based RDO process and RQT, then jump to the next PU;
- HL: apply MPM selection, RDOQ based RDO process and RQT, then continue the evaluation of other block split in current PU;
- LL: apply MPM selection, low-complexity (LC) RDO based RDO process and RQT, then continue the evaluation of other block split in current PU;
- LH: skip all steps and continue the evaluation of other block split in current PU;
- HH: skip all steps and continue the evaluation of other block split in current PU;

### Table I: Coding efficiency and complexity reduction for proposed fast intra prediction on HM10.0 for Class A-E sequences

<table>
<thead>
<tr>
<th>Class</th>
<th>Speedups</th>
<th>Chroma-U</th>
<th>Chroma-V</th>
</tr>
</thead>
<tbody>
<tr>
<td>Class A</td>
<td>Traffic</td>
<td>0.8%</td>
<td>1.5%</td>
</tr>
<tr>
<td></td>
<td>PeopleOnStreet</td>
<td>1.0%</td>
<td>1.7%</td>
</tr>
<tr>
<td></td>
<td>Nebuta</td>
<td>0.4%</td>
<td>0.9%</td>
</tr>
<tr>
<td></td>
<td>SteamLocomotive</td>
<td>0.6%</td>
<td>5.3%</td>
</tr>
<tr>
<td>Class B</td>
<td>Kimono</td>
<td>0.6%</td>
<td>1.3%</td>
</tr>
<tr>
<td></td>
<td>ParkScene</td>
<td>1.0%</td>
<td>1.9%</td>
</tr>
<tr>
<td></td>
<td>Cactus</td>
<td>1.1%</td>
<td>3.6%</td>
</tr>
<tr>
<td></td>
<td>BasketballDrive</td>
<td>0.9%</td>
<td>2.4%</td>
</tr>
<tr>
<td>Class C</td>
<td>BasketballDrill</td>
<td>0.9%</td>
<td>2.1%</td>
</tr>
<tr>
<td></td>
<td>BQMall</td>
<td>1.0%</td>
<td>1.7%</td>
</tr>
<tr>
<td></td>
<td>PartyScene</td>
<td>0.9%</td>
<td>1.3%</td>
</tr>
<tr>
<td></td>
<td>RaceHorses</td>
<td>0.9%</td>
<td>1.5%</td>
</tr>
<tr>
<td>Class D</td>
<td>BasketballPass</td>
<td>0.9%</td>
<td>1.6%</td>
</tr>
<tr>
<td></td>
<td>BQQSquare</td>
<td>0.9%</td>
<td>1.2%</td>
</tr>
<tr>
<td></td>
<td>BlowingBubbles</td>
<td>0.9%</td>
<td>1.7%</td>
</tr>
<tr>
<td></td>
<td>RaceHorses</td>
<td>1.0%</td>
<td>1.7%</td>
</tr>
<tr>
<td>Class E</td>
<td>FourPeople</td>
<td>0.8%</td>
<td>1.3%</td>
</tr>
<tr>
<td></td>
<td>Johnny</td>
<td>1.0%</td>
<td>2.1%</td>
</tr>
<tr>
<td></td>
<td>KristenAndSara</td>
<td>0.9%</td>
<td>1.7%</td>
</tr>
<tr>
<td></td>
<td>Ave</td>
<td>0.9%</td>
<td>1.9%</td>
</tr>
<tr>
<td>Encoding Time [%]</td>
<td>54%</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### IV. Performance Evaluation and Comparison

In this section, experiments are carried out to verify the performance of the proposed fast intra mode decision. Our proposed fast intra method is compared with the default algorithm in HM10.0, following the common conditions defined in [17]. All intra encoder setting is simulated to demonstrate the performance. Class A (4Kx2K), B (1080p), C (WVGA), D (QHD), and E (HD).
TABLE II: Performance comparison between proposed method and existing algorithms (with respect to the luma BD-Rate loss \(\Delta R\) and encoding time reduction \(\Delta T\))

<table>
<thead>
<tr>
<th>Class</th>
<th>(\Delta R)</th>
<th>(\Delta T)</th>
<th>(\Delta R)</th>
<th>(\Delta T)</th>
<th>(\Delta R)</th>
<th>(\Delta T)</th>
<th>(\Delta R)</th>
<th>(\Delta T)</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>1.6%</td>
<td>37%</td>
<td>1.4%</td>
<td>19%</td>
<td>0.6%</td>
<td>20%</td>
<td>0.7%</td>
<td>44%</td>
</tr>
<tr>
<td>B</td>
<td>2.3%</td>
<td>41%</td>
<td>1.3%</td>
<td>18%</td>
<td>0.7%</td>
<td>20%</td>
<td>0.9%</td>
<td>45%</td>
</tr>
<tr>
<td>C</td>
<td>0.4%</td>
<td>32%</td>
<td>-</td>
<td>-</td>
<td>0.8%</td>
<td>19%</td>
<td>0.9%</td>
<td>48%</td>
</tr>
<tr>
<td>D</td>
<td>0.3%</td>
<td>26%</td>
<td>-</td>
<td>-</td>
<td>0.9%</td>
<td>19%</td>
<td>0.9%</td>
<td>47%</td>
</tr>
<tr>
<td>E</td>
<td>1.7%</td>
<td>51%</td>
<td>-</td>
<td>-</td>
<td>0.8%</td>
<td>20%</td>
<td>0.9%</td>
<td>44%</td>
</tr>
</tbody>
</table>

D (QWVGA) and E (720p) sequences are all used for performance verification. BD-Rate performance and encoder time reduction are shown in Table I. On average, our proposed solution achieves 46% encoding time reduction for all intra coding with 0.9% BD-Rate increase (of luma component).

Meanwhile, we have implemented several other algorithms from the recent published works [18], [19], [12]. We have found the results of the these works are almost similar as claimed in their papers at the latest HM platform according to our experiments (even though different HM platforms were used in these works). It might be due to the fact that intra prediction part of the HEVC does not involve huge changes after the publication of these papers. We abstract their results and list in Table II for the side-by-side comparison with our proposed algorithm. As we can see, our proposed priority classification based method gives the decent trade-off between the coding efficiency and encoding complexity.

It is also worth to point out that, our solution has slightly larger BD-Rate drop on the chroma components in comparison to the luma channel, as highlighted in Table I. Even though human eye tends to be relatively insensitive to the chroma channels, we would like to investigate this issue as the next step and try to improve the chroma performance.

V. CONCLUSION

In this paper, we have proposed a novel priority classification based fast intra mode prediction for HEVC. Our solution assigns the one priority class out of four (i.e., HH, HL, LH, LL) to the current PU according to its neighbor PU size as well as the predicted PU depth. For the HH PUs, which are presumed to have the highest probability to be chosen as the optimum, we perform the same process as in the default HM and jump to the next PU directly; For the HL PUs, almost all the operations are the same as the HH PUs. However, instead of jumping to the next PU directly, we continue the evaluation of other depth and block sizes for the current PU; For the LH PUs, we replace the high-complexity RDQ based mode decision and RQT with low-complexity RDO solution; Eventually for those LL PUs, which are considered to have the least chance to be selected, all operations are skipped.

There are several advantages for the proposed algorithm. One aspect is that the proposed solution doesn’t require any training to derive the thresholds like other works. Hence it is general applicable to all contents. One more aspect is that proposed algorithm can be harmonized with other published works, such as fast CU/PU size decision [14], early RQT termination [9], to further boost the encoding speed. As the future study, we will also study this priority classification on the inter mode decision and form the integrated solution for the HEVC encoder.

OPEN SOURCE

We would like to open our source code for the algorithm implementation on top of HM10.0 at (URL: http://vision.poly.edu/~zma03/opensrc/pcs_sourceHM10.zip). Any comments, bugfix, and further implementations are more than welcome.

REFERENCES


