ON FAST CODING TREE BLOCK AND MODE DECISION FOR HIGH-EFFICIENCY VIDEO CODING (HEVC)

Hui Li Tan[1], *Fengjiao Liu*[2], *Yih Han Tan*[1], *Chuohao Yeo*[1]

[1]Signal Processing Department, Institute for Infocomm Research,
1 Fusionopolis Way, #21-01 Connexis, South Tower, Singapore 138632.
[2] School of Electrical & Electronic Engineering, Nanyang Technological University, Singapore

ABSTRACT

In the current HEVC test model (HM), a quad-tree based coding tree block (CTB) representation is used to signal mode, partition, prediction and residual information. The large number of combinations of quad-tree partitions and modes to be tested during rate-distortion optimization (RDO) results in a high encoding complexity. In this paper, we investigate and compare a variety of algorithms for fast CTB and mode decision. Experimental results from HM4-based implementations show that different strategies can provide a range of complexity-performance trade-offs. In particular, our proposed CU Depth Pruning algorithm can reduce encoding time by about 10% with only 0.1% coding loss, while a combination of our proposed Early Partition Decision and an early CU termination approach can reduce encoding time by about 40% with about 1% coding loss.

Index Terms— High Efficiency Video Coding (HEVC), Coding Tree Block (CTB), Fast Mode Decision.

1. INTRODUCTION

High Efficiency Video Coding (HEVC) is a new video coding standard currently under development by the Joint Collaborative Team - Video Coding (JCT-VC), targeting better compression performance than the H.264/AVC standard [1]. In the current HEVC test model (HM) [2], a quad-tree based coding tree block (CTB) representation is used, where a picture is first divided into non-overlapping largest coding units (LCU), which can be recursively divided into smaller coding units (CU).

A leaf CU, which is a CU that is not divided into sub-CUs, can either be SKIP, INTER or INTRA coded. An INTER or INTRA CU can then be further divided into prediction units (PU) of various partition sizes. In HM4, the possible PU partitions types for a CU of size $2N \times 2N$ are $2N \times 2N$, $N \times 2N$, $2N \times N$, $2N \times nU$, $2N \times nD$, $nL \times 2N$, and $nR \times 2N$ for INTER CUs, and $2N \times 2N$ and $N \times N$ for INTRA CUs, as illustrated in Figure 1.

Fengjiao Liu performed the work while at the Institute for Infocomm Research.

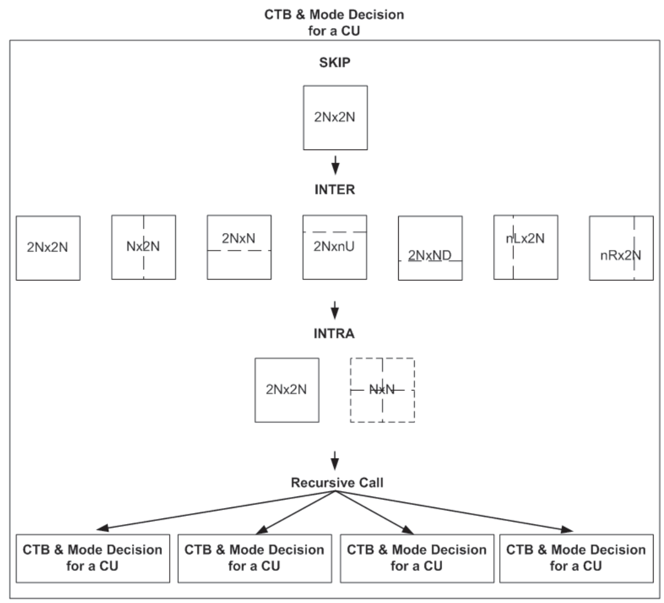

Fig. 1. Illustration of CTB, coding mode and PU partition type decision in the HM4 reference software encoder. The available partition types for INTER and INTRA CUs are also illustrated.

The CTB, based on a recursive quad-tree decomposition of the image plane [3], is an efficient representation of variable block sizes, so that regions of different sizes can be better coded. For regions of high stationarity and homogeneity, it is possible to encode them with a larger block size, resulting in a smaller side-information overhead. However, the flexibility of the variable block size structure greatly increases the search domain and hence the computational complexity of RDO at the encoder. For instance, consider an image of 704×576 pixels. Using a fixed CU size of 16×16 involves only $44 \times 36 = 1584$ CU mode decisions, while using a CTB structure of 64×64 LCU and a maximum quad-tree depth of 4 involves $11 \times 9 \times (1 + 4^1 + 4^2 + 4^3) = 8415$ CU mode decisions. To speed up the CTB and mode decision process, one possible approach is to reduce the search space by avoid-

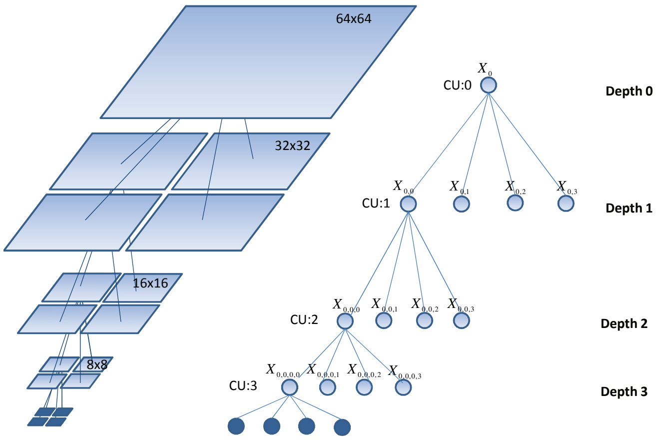

Fig. 2. Illustration of CTB, where a LCU X_{i_0} is recursively split into sub-CUs $X_{i_0,i_1,\cdots,i_m}$. A leaf CU can be further split into PUs. All leaf CUs are traversed in a depth-first manner.

ing full branching of the search tree, e.g. making early CTB termination or mode decisions.

As we will explain later, the CTB and mode decision process in the HM4 reference software encoder follows a depth-first traversal of the CTB, so prior fast CTB and mode decision approaches tend to be based on depth-first pruning of the search tree. In this paper, we also consider and propose fast mode decision approaches inspired by breadth-first search algorithms.

2. BACKGROUND AND RELATED WORKS

For ease of discussion, we shall use the following notation in this paper. Referring to Figure 2, each frame is divided into non-overlapping LCUs, where X_{i_0} denotes the i_0^{th} LCU within the frame. Let M denotes the maximum depth of the CTB representation, and m_0 denote a parameter for deciding the minimum CU size. Then, a CU at depth m, where $0 \leq m < M$, is of size $2^{m_0+(M-m)} \times 2^{m_0+(M-m)}$; therefore the LCU size is $2^{m_0+M} \times 2^{m_0+M}$ while the minimum CU size is $2^{m_0+1} \times 2^{m_0+1}$. In the HEVC common test conditions [11], $M = 4$ and $m_0 = 2$, i.e., the LCU size is 64×64, and the minimum CU size is 8×8.

Each CU at depth m is denoted by $X_{i_0,i_1,\cdots,i_m}$, where as mentioned above, i_0 indexes the location of the root LCU within the frame, while each subsequent index, $i_1, \cdots, i_m$ ($0 \leq i_1, \cdots, i_m \leq 3$), specifies the index of the CU within its parent. Therefore, $X_{i_0,i_1,\cdots,i_m}$ is the i_mth CU of $X_{i_0,i_1,\cdots,i_{m-1}}$. This notation would allow us to uniquely identify each CU within a frame. For convenience, we would also sometimes use $X_{i^{\bar{m}}}$ to denote $X_{i_0,i_1,\cdots,i_m}$, i.e., $i^{\bar{m}}$ denotes the list of indices $i_0, i_1, \cdots, i_m$.

2.1. HEVC reference encoder

In the HM4 reference software encoder, the mode decision process for each CU is carried out as shown in Figure 1, where its best coding mode and PU partition type are first determined assuming that the CU is not split before recursively repeating the same process for each of its sub-CUs. A final decision is then made as to whether the CU is to be split or not only after all the sub-CUs has been analyzed. This corresponds to a CTB traversed in a depth-first manner as shown in Figure 2. In a depth-first traversal of a tree, one starts at the root and traverses as deep as possible along each branch before backtracking [4]. Due to the recursive nature, the CTB decision is made bottom up.

We will also find the following useful for understanding the CTB and mode decision process in the HM4 reference software encoder. Denote $F(X_{i^{\bar{m}}})$ to be the best RD cost computed for the CU, $X_{i^{\bar{m}}}$, assuming that $X_{i^{\bar{m}}}$ is not split into sub-CUs. Denote $C(X_{i^{\bar{m}}})$ to be the best RD cost computed for $X_{i^{\bar{m}}}$, i.e., without any restriction on whether it is split or not. Then, the HM4 encoder is essentially optimizing the RD cost for $X_{i^{\bar{m}}}$ by using the following recursive relationship:

$$C(X_{i^{\bar{m}}}) = \begin{cases} \min\{C_0 + F(X_{i^{\bar{m}}}), \\ \quad C_1 + \sum_{i_{m+1}=0}^{3} C(X_{i^{\bar{m}},i_{m+1}})\} & \text{if } m < M-1, \\ C_0 + F(X_{i^{\bar{m}}}) & \text{otherwise.} \end{cases} \tag{1}$$

where C_0 and C_1 represents the overhead of not splitting the CU and splitting the CU respectively. For each LCU, X_{i_0}, the HM4 encoder will thus try to compute the best RD cost starting from $C(X_{i_0})$.

2.2. Related work

As HEVC is currently under development, there has been limited work on fast CTB and mode decision techniques for it. Early Skip [5], CBF-based Early Termination [6] and Early CU Termination (ECU) [7] are some fast CTB and mode decision methods already present in the reference software. Early Skip is currently disabled but can be enabled by adding a line of code. Early Skip attempts to terminate the CU mode decision after checking the SKIP mode, by testing if the SKIP RD cost is less than a threshold. CBF-based Early Termination stops the CU mode decisions if there is no residual to be coded after checking an INTER mode with any PU partition types, i.e. the coding block flag (CBF) is zero. ECU stops the CTB decision by checking if the best coding mode for the current CU is SKIP. If so, it would not further check the case of dividing the current CU into four sub-CUs.

Other relevant works on fast mode decision are those developed for H.264. In [8], the number of Intra prediction modes after RD calculation is reduced based on statistics of a edge directional histogram. In [9], fast mode decision and motion estimation algorithms based on neighbouring modes, residue and distortion information are proposed. In [10], Early Skip mode decision and selective Intra mode decision are proposed to reduce the temporal and spatial redundancy of the encoding process respectively. These prior works are based on the depth-first pruning of the search tree.

Since we seek to judiciously prune the CTB search tree at the upper branches, we can also consider traversing the CTB tree in a breadth-first manner to avoid traversing down to the full depth of the CTB tree before deciding if the case of splitting a CU should be checked. In a breadth-first traversal of the CTB tree, one begins at the root node and first traverses all its child nodes before proceeding further down the tree. While a breadth-first traversal may lead to better search tree pruning, the dependencies between child CUs with the same parent CU makes the implementation of a breadth-first algorithms for CTB optimization non-trivial. In the next section, we propose two algorithms based on depth-first pruning and two algorithms inspired by breadth-first traversal for fast CTB and mode decision.

3. PROPOSED METHODS

3.1. Early Partition Decision

Early Partition Decision (EPD) attempts to terminate the mode decision process after checking the INTER mode with each of the PU partition types. At each stage, if the current best RD cost is less than a threshold, the mode decision process will stop. In our current implementation, the threshold is simply set to be the average RD cost of a number of previously coded SKIP CUs of the same CU size. Another possibility is to set the threshold as a scaled average instead.

The rationale for this approach is that when a CU achieves a RD cost comparable to the RD cost of CUs that have SKIP as the best mode, it is likely that there is no further need to check other possible modes, PU partition types, and further splitting of the CU. Hence, pruning of the search tree can be achieved.

It is also possible to combine this method with other fast mode decision methods. In particular, we found that combining EPD with ECU [7] leads to a useful complexity-performance operating point. In this combination, the Early Partition Decision approach is carried out as described above. In addition, as in [7], if the best CU mode is SKIP mode, the case of splitting the CU will not be checked.

3.2. CU Depth Pruning - Full Check

CU Depth Pruning - Full Check (CDP-F) attempts to terminate the CTB decision process by performing a one-level look-ahead. First, the best CU mode and PU partition types of the current CU, $X_{i^{\bar{m}}}$, is first determined. However, before the recursive call to perform CTB and mode decision of its sub-CUs, an early termination decision can be made by comparing the current best RD cost and the sum of best RD cost of the four sub-CUs, assuming the sub-CUs are not split further. That is, if the following holds:

$$\sum_{i_{m+1}=0}^{3} F(X_{i^{\bar{m}},i_{m+1}}) > F(X_{i^{\bar{m}}}), \tag{2}$$

the current CU is assumed to be a leaf CU, i.e., it is not split anymore. The intuition behind this is that splitting the CUs typically improves prediction at the cost of increasing overhead, and if splitting the CU does not result in any improvement in RD cost, then it is unlikely that a CTB which splits the current CU would result in a better RD cost.

On the other hand, if early termination for the current CU is not done, then the same process needs to be repeated for each of its sub-CUs. Unfortunately, this also means that the best CU mode and PU partition type of each sub-CU are searched twice: the first time when the best prediction mode and PU partition is checked as a sub-node at depth m; and the second time when the best CU mode and PU partition types is checked as a node at depth $m+1$. To mitigate this redundancy, we do the following. First, this early termination decision is done only for CUs of depths $0 \leq m < M-2$, since at level $m = M-2$, we already know that splitting of the CU cannot occur at the next depth. Second, after the best prediction mode and partition type of the first sub-CU, $X_{i^{\bar{m}},0}$, is determined during the early termination check, they are saved, and loaded from memory when they are next needed. Note that, this can only be done for $i_{m+1} = 0$, because the remaining sub-CUs have coding dependencies on the first sub-CU. Therefore, the mode decision and PU partition type decision still needs to be performed twice for them.

3.3. CU Depth Pruning - Partial Check

To eliminate the repetitions in mode decision and PU partition type checks, we modify the CDP-F approach in the following way. For each CU that is the 4th sub-CU of its parent, i.e., $X_{i^{\bar{m}},3}$, if the sum of the RD cost of its sibling sub-CUs and itself is larger than the best current RD cost of its parent CU, i.e., if the following holds:

$$F(X_{i^{\bar{m}},3}) + \sum_{i_{m+1}=0}^{2} C(X_{i^{\bar{m}},i_{m+1}}) > F(X_{i^{\bar{m}}}), \tag{3}$$

then branching is terminated for $X_{i^{\bar{m}},3}$. This method will be referred to as CDP-P.

4. EXPERIMENTAL RESULTS

Our algorithms are implemented on HM4, and tested using the common test conditions [11]. The HEVC test sequences cover a range of resolutions, including Class A (2560x1600

	RA-HE		RA-LC		LB-HE		LB-LC	
Method	Luma BD-Rate(%)	Enc Time(%)	Luma BD-Rate (%)	Enc - Time(%)	Luma BD-Rate (%)	Enc Time(%)	Luma BD-Rate(%)	Enc Time(%)
Early SKIP	4.7	50	5.2	42	5.4	48	5.7	41
CBF	1.1	66	1	60	1.3	64	1.2	60
ECU	0.5	65	0.6	61	0.2	66	0.3	63
EPD	0.7	66	0.7	61	0.8	66	0.8	63
EPD+ECU	1.1	58	1.2	51	1.1	57	1.1	54
CDP-F	1.3	91	1	87	1	95	0.6	100
CDP-P	0.1	93	0.1	90	0.1	92	0.1	91

Table 1. Coding losses (Luma BD-Rate), reflected as positive BD-Rate percentages, and encoding times (Enc Time), reflected as percentage of HM4 anchor encoder runtimes, for all combinations of Random Access (RA) and Low Delay (LD) settings and High Efficiency (HE) and Low Complexity (LC) configurations.

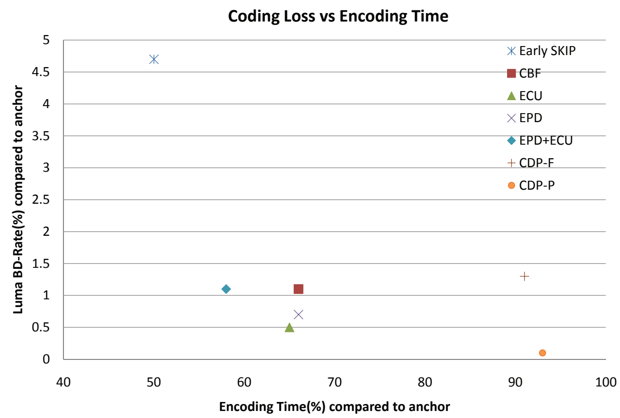

Fig. 3. Plot of coding loss vs encoding time for various fast CTB and mode decision methods under Random Access, High Efficiency configuration.

pixels), Class B (1920x1080 pixels), Class C (832x480 pixels), Class D (416x240 pixels) and Class E (1280x720 pixels). Using HM4 as the anchor, the performances of our algorithms in Random Access (RA) and Low Delay (LD) settings, for both High Efficiency (HE) and Low Complexity (LC) configurations are assessed. For each method and configuration, a video sequence is coded at 4 QPs, $\{22, 27, 32, 37\}$. The coding performance of the proposed methods are measured by Bjontegaard Delta PSNR (BD-PSNR) [12], and their encoding time are presented as a percentage of the anchor encoding run-times.

Simulation results of the four proposed methods as well as existing methods are summarized in Table 1. EPD reduces encoding time by 34% with 0.7% coding loss, while its combination with ECU (EDP+ECU) reduces encoding time by 42% with 1.1% coding loss. On the other hand, the CDP algorithms give smaller encoding time savings. CDP-F reduces encoding time by 9% with 1.3% coding loss, while CDP-P reduces encoding time by 7% with 0.1% coding loss.

Figure 3 shows a plot of coding losses against encoding times for the RA-HE configuration. The plots for the other 3 encoding configurations are similar. Operating points towards the lower left corner of the graph are desirable since they represent fast CTB and mode decision methods that achieve low encoding time with minimal coding loss. Both EDP+ECU and CDP-P offer competitive complexity-performance trade-offs, since they lie on the convex hull of the operating points.

5. CONCLUSION AND FUTURE WORKS

The use of CTB in HEVC greatly increases the search domain for the RDO testing of all possible combinations of quad-tree partitions and coding modes. We have investigated and compared a variety of fast coding tree block and mode decision algorithms. Simulation results show that different strategies can provide a variety of complexity-performance trade-offs. In particular, a combination of our proposed Early Partition Decision with an early CU termination algorithm, as well as CU Depth Pruning, offer competitive complexity-performance trade-off. Our future works include looking into other quad-tree optimization techniques to prune the CTB search for better encoder speed-ups.

6. REFERENCES

[1] "ITU-T rec. H.264ISO/IEC 14496-10. Advanced video coding for generic audio-visual services," Joint Video Team, Doc. JVT-V202, 2005.

[2] B. Bross, W.-J. Han, J.-R. Ohm, G. J. Sullivan, T. Wiegand, "WD4: Working Draft 4 of High-Efficiency Video Coding" in JCT-VC of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, JCTVC-F803, 6th Meeting: Torino, IT, 14-22 July, 2011.

[3] H. Samet, "The Quad Tree and Related Hierarchical Data Structures," ACM Computing Surveys, Vol. 16 No. 2, June 1984, pp. 188-260.

[4] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein, "Introduction to Algorithms," Second Edition. MIT Press and McGraw-Hill, 2001. ISBN 0-262-03293-7. Section 22.3: Depth-first search, pp. 540549.

[5] W.-J. Han et al., "Improved Video Compression Efficiency Through Flexible Unit Representation and Corresponding Extension of Coding Tools", in TCSVT, Dec 2010.

[6] R. H. Gweon, Y.-L. Lee, J. Lim, "Early Termination of CU Encoding to Reduce HEVC Complexity," in JCT-VC of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, JCTV -F045, 6th Meeting: Torino, 2011.

[7] K. Choi, S. H. Park, E. S. Jang,"Coding tree pruning based CU early termination" in JCT-VC of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, JCTV-F092, 6th Meeting: Torino, 2011.

[8] D. Wu, F. Pan, K.P. Lim, S. Wu, Z.G. Li, X. Lin, S. Rahardja, C.C. Ko, "Fast intermode decision in H.264/AVC video coding," TCSVT, vol.15, no.7, pp. 953- 958, July 2005.

[9] X. Lu, A.M. Tourapis, P. Yin, J. Boyce, "Fast mode decision and motion estimation for H.264 with a focus on MPEG-2/H.264 transcoding," in ISCAS 2005, pp. 1246- 1249 Vol. 2, 23-26 May 2005.

[10] I. Choi, J. Lee, B. Jeon,"Fast Coding Mode Selection With Rate-Distortion Optimization for MPEG-4 Part-10 AVC/H.264," TCSVT, vol.16, no.12, pp.1557-1561, Dec. 2006.

[11] F. Bossen, "Common conditions and software reference configurations," in JCT-VC of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, JCTVC-F900, 6th Meeting: Torino, IT, Jul., 2011.

[12] G. Bjontegaard, "Calculation of average PSNR differences between RD-curves," 13th VCEG-M33 Meeting: Austin, Texas, USA, 2-4 April, 2001.