FAST BOTTOM-UP PRUNING FOR HEVC INTRAFRAME CODING

Han Huang, Yao Zhao, Chunyu Lin and Huihui Bai

Institute of Information Science, Beijing Jiaotong University, Beijing, China

ABSTRACT

In intraframe coding of the High Efficiency Video Coding (HEVC) standard, up to 35 modes are defined for intra prediction and the quadtree structure is used for adaptive block partition. While such flexibility leads to more efficient compression, it also dramatically increases the encoder complexity. In this paper, a simple yet effective fast bottom-up pruning algorithm is proposed to reduce the computational cost. Mode decision at a large coding unit (CU) is selectively skipped based on the block structures of its sub-CUs. Our experimental results show that the proposed scheme can effectively reduce the encoder complexity without compromising the compression efficiency.

***Index Terms*—** HEVC, fast mode decision, intra coding, video coding, bottom-up pruning

1. INTRODUCTION

The increased demand of higher resolution and better quality video inspires the ITU-T Video Coding Experts Group (VCEG) and ISO/IEC Moving Picture Experts Group (MPEG) to form the Joint Collaborative Team on Video Coding (JCT-VC) and develop the new High Efficiency Video Coding (HEVC) standard. In comparison to its prior standard H.264/AVC , approximately 50% bitrate reduction is achieved for equal perceptual quality [1].

The HEVC standard employs the same block-based hybrid coding structure as in H.264/AVC, but introduces many new technical features and characteristics. In HEVC intra coding, it follows the general quadtree-based *coding unit* (CU) structure and defines 35 intra prediction modes [2]. While the flexible block structure and more choices of prediction mode provides better coding performance, it dramatically increases the encoder complexity. Even for intraframe coding, it is still far away from real-time application [3]. Thus, reducing the encoder complexity is desirable. Some fast intra mode decision techniques have been proposed in [4–10]. Though fast CU size decision algorithms are also found in [11–14], they are usually designed for interframe coding. Therefore, they are not applicable in intraframe coding. In this paper, we focus on the CU size decision in intraframe coding. A novel fast bottom-up pruning technique is proposed. It utilizes the coding information obtained at the bottom nodes of the quadtree to selectively skip the mode decision process at a parent node. Therefore, the encoder complexity can be reduced. The proposed method is simple and effective. To the authors' knowledge, it is the first fast bottom-up pruning algorithm for HEVC intraframe coding in the literature.

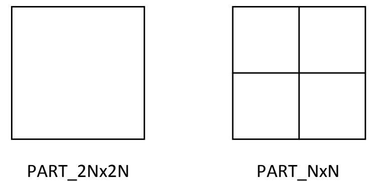

Fig. 1. PU types for intra prediction

This paper is organized as follows. The block structures in HEVC and the bottom-up pruning algorithm that is adopted in the reference software are described in Section 2. Then the proposed method is presented in Section 3. And experimental results are given in Section 4. It shows that the proposed method can skip a significant percentage of mode decision process at large coding unit. Up to 37.56% encoding time savings are achieved without noticeable loss in rate-distortion performance. Finally, conclusions are drawn in Section 5.

2. BACKGROUND

In the HEVC standard, a picture is first partitioned into nonoverlapping *coding tree units* (CTU), i.e. the largest CU. Each CU can be recursively split into a quadtree of sub-CUs. A leaf node can be further split into one, two or four *prediction units* (PU) depending on its PU type. In intra coding, the PU type can be either PART_2Nx2N or PART_NxN, as shown in Fig. 1. However, PART_NxN is only allowed for the minimum CU size, otherwise it is similar as four equal-size sub-CUs. Analogous to the quadtree of CU, the residual signal of a CU can be recursively split into multiple *transform units* (TU) to form a *residual quadtree* (RQT). More details on the block structures in HEVC can be found in [15].

To efficiently capture the structural information in images, HEVC specifies 35 intra prediction modes: planar mode, DC mode and 33 angular prediction modes. An overview of the intra coding in HEVC is provided in [2]. Exhaustive

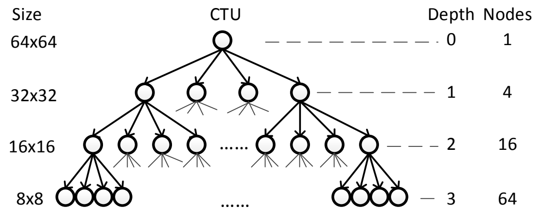

Fig. 2. Full quadtree

search of the best combination of CU structure, intra prediction mode and RQT is computational prohibitive. In the HEVC test model (HM) reference software [16], the RD optimized bottom-up pruning algorithm [17] is adopted to find the best quadtree partition.

Typically, the CTU size is 64x64 and the maximum number of depth levels of the quadtree is $D = 4$. Consider the full quadtree as shown in Fig. 2, we locate a node by its depth level from top to bottom and its position from left to right. Let (i, j) be the index of jth node at depth level i, then $j < 4^i$. Denote $\mathcal{X}_{i,j}$ as a CU at node (i, j) without further splitting and $\mathcal{C}_{i,j}$ as the optimal tree after rate-distortion optimized pruning. Let $J(\cdot)$ be the operation to calculate the best RD cost. The bottom-up pruning algorithm traverses the full quadtree in a depth-first order. The sub-CUs of node (i, j) is pruned if and only if $J(\mathcal{X}_{i,j}) <= \sum_{k=0}^{3} J(\mathcal{C}_{i+1,4j+k})$.

3. PROPOSED METHOD

3.1. Analysis of bottom-up pruning algorithm

The bottom-up pruning algorithm described in Section 2 performs mode decision at each node of the full quadtree. Therefore, a total of $\sum_{i=0}^{D-1} 4^i$ for each CTU. However, we notice that the coding information obtained at the sub-CU nodes can be utilized to avoid unnecessary operations at their parent node.

Given a CU at node (i, j), the best partition and prediction modes of its sub-CUs $\mathcal{C}_{i+1,4j+k}, k = 0, 1, 2, 3$ are known by bottom-up pruning. Let $Q(\cdot) = 0$ indicating a leaf node and $Q(\cdot) = 1$ indicating a tree. If $Q(\mathcal{C}_{i+1,4j+k}) = 1$, i.e. the kth sub-CU is further split, it suggests that the structural information in the kth sub-CU is complex. Hence, the current CU is likely to be split by rate-distortion optimization. An example is shown in Table 1. It shows the probability of a CU is split when $\sum Q(\mathcal{C}_{i+1,4j+k}) = n$. The results are obtained by coding the first second (50 frames) of HEVC test sequence *BasketballDrill*. Similar statistics are found in coding the other sequences. Note that at level 3, it reaches the leaf node of the quadtree, thus the CU doesn't have any sub-CUs. However, HEVC allows PART_NxN type PU at the bottom level, in which case the CU can also be considered as a tree.

Table 1. The probability of a CU at node (i, j) is split when $\sum Q(\mathcal{C}_{i+1,4j+k}) = n$.

i	$n = 0$	$n = 1$	$n = 2$	$n = 3$	$n = 4$
0	0.97	0.98	0.99	1.00	1.00
1	0.49	0.76	0.88	0.95	0.98
2	0.33	0.69	0.86	0.92	0.97

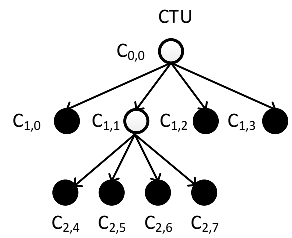

Fig. 3. Example of a pruned quadtree.

3.2. Fast bottom-up pruning algorithm

Based on the observations described in Section 3.1, we can conclude that there is a high probability that a CU remains split if its sub-CUs are sub-trees. In other words, lots of mode decision process in a large CU is unnecessary. Therefore, they could be avoided to save the computational complexity. In this paper, $\sum_{k=0}^{3} Q(\mathcal{C}_{i+1,4j+k}) >= i + 1$ is set as the necessary condition to skip the intra mode decision process at node (i, j). An example is shown in Fig. 3. If $C_{1,1}$ is a tree, then the sub-CUs of node $(0, 0)$ are not pruned. Implementation details of the proposed fast bottom-up pruning algorithm is described by the pseudocode in Algorithm 1.

Algorithm 1 Fast bottom-up pruning algorithm

function MODEDECISION($\mathcal{X}_{i,j}, \mathcal{C}_{i,j}$)
  if $i < D - 1$ **then**
    Generate sub-CUs $\mathcal{X}_{i+1,4j+k}, k = 0, 1, 2, 3$
    for $k = 0 \to 3$ **do**
      MODEDECISION($\mathcal{X}_{i+1,4j+k}, \mathcal{C}_{i+1,4j+k}$)
    end for
    $\mathcal{C}_{i,j} \leftarrow \cup_{k=0}^{3} \mathcal{C}_{i+1,4j+k}$
    $J(\mathcal{C}_{i,j}) \leftarrow \sum_{k=0}^{3} J(\mathcal{C}_{i+1,4j+k})$
  end if
  if $i == D - 1$ or $\sum_{k=0}^{3} Q(\mathcal{C}_{i+1,4j+k}) < i + 1$ **then**
    Do intra mode decision and Calculate $J(\mathcal{X}_{i,j})$
    if $i == D - 1$ or $J(\mathcal{X}_{i,j}) \leq J(\mathcal{C}_{i,j})$ **then**
      $\mathcal{C}_{i,j} \leftarrow \mathcal{X}_{i,j}$
      $J(\mathcal{C}_{i,j}) \leftarrow J(\mathcal{X}_{i,j})$
    end if
  end if
end function

It should be noted that the proposed fast bottom-up pruning algorithm does not introduce much extra operations when compared to the original bottom-up algorithm. The only additional operations are the storing of 4 tags for the sub-CUs, one summation operation and one comparison operation. Therefore, the proposed algorithm is simple and easy to implement.

4. EXPERIMENTS AND DISCUSSION

The proposed method was evaluated in the HEVC reference software HM8.2 [16]. The main profile intra-only encoder configuration was used to code the HEVC test sequences in classes A to E [18]. Maximum CU size (CTU size) was 64x64, the maximum number of CU depth levels was 4, and maximum number of TU levels was 3. Rate-distortion optimized quantization, sample adaptive offset, transform skipping, and fast transform skipping were on. Four quantization parameters (QP) $\{22, 27, 32, 37\}$, as suggested in [18], were used to encode each sequence. The proposed fast bottom-up pruning (FBUP) algorithm is compared with the optimal bottom-up pruning algorithm (BUP) in the original HM8.2 coder.

To demonstrate the encoding time saving (ETS) of the proposed method, an isolated PC with Intel Core 2 3.0GHz CPU and 2.0GB RAM was used to encode the first second of each sequence. The ETS, in percentages, and the corresponding BD-rate [19] for FBUP are shown in Table 2. Generally, FBUP achieves more ETS at low QP values (high bitrates) but less at high QP values (low bitrates). It is expected since larger block size is more favorable at low bitrate coding and the chance of a CU has sub-CUs is lower. The max ETS is 37.56% and the max bitrate increase is 1.25%. Averaging by all sequences, ETS ranges from 16.49% to 24.25% depending on the QP values, and the average bitrate increase is only 0.46%. When coding high resolution sequences, ETS is relatively smaller since large block is more frequent. However, note that more than 20% ETS is also observed.

The ETS can only tell us a general idea of the performance of FBUP, since it depends on the condition of the machine that is used for experiments. And the reference software is not optimized. To better demonstrate the efficiency of the proposed FBUP algorithm, the average percentages of skipped nodes at each level and the BD-rate of Y component are shown in Table 3. The entire clip (10s) of each sequence was used for experiment. We can observe that significant percentages of CU nodes can be skipped by the FBUP algorithm without noticeable bitrate increase. The number varies by different depth levels. That is because the node at top levels is more likely to have subtrees. Note that the FBUP is not applicable at the maximum depth level 3. Averaging over the top 3 levels, 37.94% nodes are skipped and the bitrate increase is also only 0.46%. In coding sequences *NebutaFestiva*, *SteamLocomotive* and *Kimono*, the performance of FBUP is less significant. In these sequences, large similar color regions would result in choosing more large blocks. For example, the green trees and the white coat in sequence *Kimono*, the gray smoke, white snow and black train in *SteamLocomotive*, and the carton texture in *NebutaFestiva*. Overall, the FBUP algorithm can skip a large percentage of mode decision process at large CUs. Thus, the computation complexity is reduced.

Table 2. Encoding time saving (%) and corresponding Y BD-rate of FBUP (%).

Sequence		QP=22	QP=27	QP=32	QP=37	Y BD-rate
Class A 2560x1600	Traffic	23.46	22.43	18.41	13.88	0.54
	PeopleOnStreet	24.41	20.60	12.34	19.90	0.40
	NebutaFestival	10.04	9.40	13.89	14.41	0.44
	SteamLocomotive	14.38	4.93	13.91	14.28	1.25
	Average	**18.07**	**14.43**	**14.64**	**15.62**	**0.66**
Class B 1920x1080	Kimono	10.05	5.13	6.75	7.05	0.79
	ParkScene	25.73	20.25	15.93	15.57	0.73
	Cactus	26.33	21.89	18.12	13.98	0.54
	BQTerrace	19.39	21.79	21.68	18.71	0.32
	BasketballDrive	17.30	14.43	11.82	13.50	0.60
	Average	**19.76**	**16.70**	**14.86**	**13.76**	**0.60**
Class C 832x480	RaceHorese	27.90	25.17	21.20	18.49	0.38
	BQMall	31.73	28.04	22.33	20.96	0.27
	PartyScene	35.43	32.28	32.61	25.70	0.07
	BasketballDrill	31.43	26.81	22.15	17.38	0.43
	Average	**31.62**	**28.08**	**24.57**	**20.63**	**0.29**
Class D 416x240	RaceHorese	27.81	26.26	24.11	19.95	0.20
	BQSquare	36.44	34.39	30.28	25.81	0.00
	BlowingBubbles	37.56	27.96	24.26	18.68	0.14
	BasketballPass	26.09	19.57	17.85	11.30	0.37
	Average	**31.98**	**27.05**	**24.13**	**18.94**	**0.18**
Class E 1280x720	FourPeople	22.98	20.41	17.94	14.33	0.36
	Johnny	17.41	16.65	10.19	13.47	0.89
	KristenAndSara	19.11	16.99	13.10	12.42	0.43
	Average	**19.83**	**17.99**	**13.74**	**13.41**	**0.56**
Total Average		**24.25**	**20.76**	**18.44**	**16.49**	**0.46**

5. CONCLUSION

In this paper, a fast bottom-up pruning algorithm for HEVC intraframe coding is presented. The essential idea is to utilize the coding information obtained at the sub-CUs. The proposed method is effective and easy to implement. Future work for further improvements includes combining with top-down approaches, adaptive CU range selection and fast intra prediction mode decision.

Acknowledgements

This work was supported in part by the National Natural Science Funds for Distinguished Young Scholar, under Grant 61025013; the 973 Program, under Grant 2012CB316400; and Program for Changjiang Scholars and Innovative Research Team in University.

6. REFERENCES

[1] G. J. Sullivan, J.-R. Ohm, W.-J. Han, T. Wiegand, and T. Wiegand, "Overview of the High Efficiency Video Coding (HEVC) standard," *IEEE Trans. Circuits Syst.*

Table 3. Average percentages of skipped nodes at each depth level and BD-rate (%) of FBUP.

Sequence	level 0	level 1	level 2	BD-rate
Traffic	95.39	49.79	6.59	0.54
PeopleOnStreet	96.94	59.95	8.22	0.41
NebutaFestival	59.33	15.56	1.59	0.37
SteamLocomotive	68.95	17.54	1.37	1.28
Average	**80.15**	**35.71**	**4.44**	**0.65**
Kimono	71.52	19.75	2.61	0.95
ParkScene	91.22	52.87	10.74	0.67
Cactus	92.65	49.68	9.49	0.54
BQTerrace	91.46	56.78	4.45	0.28
BasketballDrive	85.58	36.12	2.54	0.51
Average	**86.49**	**43.04**	**5.97**	**0.59**
RaceHorese	93.73	54.57	10.50	0.46
BQMall	95.40	62.94	12.38	0.28
PartyScene	99.91	88.93	33.75	0.08
BasketballDrill	99.37	69.88	14.93	0.43
Average	**97.10**	**69.08**	**17.89**	**0.31**
RaceHorese	97.56	68.87	14.00	0.30
BQSquare	98.45	78.15	37.75	0.12
BlowingBubbles	99.97	91.13	34.67	0.08
BasketballPass	94.30	56.21	12.47	0.15
Average	**97.57**	**73.59**	**24.72**	**0.16**
FourPeople	90.21	46.73	6.92	0.37
Johnny	66.14	26.37	3.42	1.04
KristenAndSara	74.09	29.20	4.29	0.43
Average	**76.81**	**34.10**	**4.87**	**0.61**
Total Average	**88.11**	**51.55**	**11.63**	**0.46**

Video Technol., vol. 22, no. 12, pp. 1649 –1668, Dec. 2012.

[2] J. Lainema, F. Bossen, Woo-Jin Han, Junghye Min, and K. Ugur, "Intra coding of the HEVC standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1792 –1801, Dec. 2012.

[3] F. Bossen, B. Bross, K. Suhring, and D. Flynn, "HEVC complexity and implementation analysis," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1685 –1696, Dec. 2012.

[4] Yinji Piao, Junghye Min, and Jianle Chen, "Encoder improvement of unified intra prediction," in *document JCTVC-C207*, Oct. 2010.

[5] Liang Zhao, Li Zhang, Siwei Ma, and Debin Zhao, "Fast mode decision algorithm for intra prediction in HEVC," in *Proc. IEEE Visual Commun. Image Process.*, Nov. 2011, pp. 1 –4.

[6] T.L. da Silva, L.V. Agostini, and L.A. da Silva Cruz, "Fast HEVC intra prediction mode decision based on edge direction information," in *Proc. the 20th European Signal Process. Conf. (EUSIPCO)*, aug. 2012, pp. 1214 –1218.

[7] H. Zhang and Z. Ma, "Early termination schemes for fast intra mode decision in High Efficiency Video Coding," in *IEEE Int. Symp. Circuits and Syst. (ISCAS)*, may. 2013.

[8] Wei Jiang, Hanjie Ma, and Yaowu Chen, "Gradient based fast mode decision algorithm for intra prediction in hevc," in *Consumer Electronics, Communications and Networks (CECNet), 2012 2nd International Conference on*, 2012, pp. 1836–1840.

[9] Yongfei Zhang, Zhe Li, and Bo Li, "Gradient-based fast decision for intra prediction in hevc," in *Visual Communications and Image Processing (VCIP), 2012 IEEE*, 2012, pp. 1–6.

[10] Heming Sun, Dajiang Zhou, and S. Goto, "A low-complexity hevc intra prediction algorithm based on level and mode filtering," in *Multimedia and Expo (ICME), 2012 IEEE International Conference on*, 2012, pp. 1085–1090.

[11] Xiaolin Shen, Lu Yu, and Jie Chen, "Fast coding unit size selection for HEVC based on bayesian decision rule," in *Picture Coding Symposium (PCS)*, May 2012, pp. 453 –456.

[12] Hui Li Tan, Fengjiao Liu, Yih Han Tan, and Chuohao Yeo, "On fast coding tree block and mode decision for High-Efficiency Video Coding (HEVC)," in *IEEE Int. Conf. Acoust., Syst., Signal Process.*, March 2012, pp. 825 –828.

[13] L. Shen, Z. Liu, X. Zhang, W. Zhao, and Z. Zhang, "An effective CU size decision method for hevc encoders," *IEEE Trans. Multimedia*, vol. 15, no. 2, pp. 465 –470, Feb. 2013.

[14] Kiho Choi and Euee S. Jang, "Fast coding unit decision method based on coding tree pruning for high efficiency video coding," *Optical Engineering*, vol. 51, no. 3, pp. 030502–1–030502–3, 2012.

[15] I.-K. Kim, J. Min, T. Lee, W.-J. Han, and J. Park, "Block partitioning structure in the HEVC standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1697 –1706, Dec. 2012.

[16] HEVC test model reference software HM8.2 [Online]. Available:, "`https://hevc.hhi.fraunhofer.de/svn/svn_HEVCSoftware/tags/HM-8.2/`," .

[17] G.J. Sullivan and R.L. Baker, "Efficient quadtree coding of images and video," *IEEE Trans. Image Process.*, vol. 3, no. 3, pp. 327 –331, May 1994.

[18] F. Bossen, "Common test conditions and software reference configurations," in *document JCTVC-I1100*, May 2012.

[19] Gisle Bjontegaard, "Improvements of the BD-PSNR model," Tech. Rep. VCEG-AI11, ITU-T Video Coding Experts Group(VCEG), Berlin, Germany, July 2008.