

Fast Intra CU Size Decision for HEVC Based on Machine Learning

Meng Wang^{1,a}, Junru Li^{2,b}, Xiaodong Xie^{2,c*}, Yuan Li^{2,d} and Huizhu Jia^{2,e}

¹ School of Electronic and Computer Engineering, Shenzhen Graduate School, Peking University, 518055, Shenzhen, China

² National Engineering Laboratory of Video Technology, Peking University, China

^a mengw@pku.edu.cn, ^b junru.li@pku.edu.cn, ^c donxie@pku.edu.cn, ^d yuanli@pku.edu.cn, ^e hzjia@pku.edu.cn

*The corresponding author, Xiaodong Xie is with Peking University, also with Cooperative Medianet Innovation Center and Beida (Binhai) Information Research

Keywords: HEVC, Intra prediction, CU size, Fast algorithm.

Abstract: High Efficiency Video Coding (HEVC) is the new generation of video coding standard. A quad-tree based Coding Unit (CTU) partitioning scheme is used to adapt to different video contents. However, it brings the dramatically increasing of coding complexity because there are a large amount of CU partition structure to traverse. In this paper, we proposed a fast CU size decision method based on machine learning. CU features is extracted and Support Vector Machine (SVM) model is trained to classify CU splitting or non-splitting. Experimental results show that our proposed method can achieve 40.23% encoding time saving on average and the BD-rate loss is only 0.83% under All Intra (AI) configuration.

1. Introduction

The development of video socialization applications brings enormous challenges to video coding technology. Joint Collaborative Team on Video Coding (JCT-VC) develops new generation video coding standard HEVC, which provides double coding efficiency compared with the previous coding standard. But it involves much higher computational complexity [1]. Reducing the encoding complexity is vital for this coding standard.

In HEVC, a video frame is divided into non-overlapping Coding Tree Units (CTU). CTU is further split into Coding Unit (CU) according to the quad-tree in a recursive manner. CU is the basic coding processing unit. The size of CU ranges from 64×64 to 8×8 in HEVC standard and CU will be continuously divided into Prediction Units (PU). The structure of CTU is shown in Figure 1. Besides, up to 35 prediction modes are adopted including DC mode, planar mode and 33 angular modes. With flexibly coding structures and multiple prediction modes, the coding complexity is dramatically increased, because encoder need to traverse all those modes and selects the best coding parameter set in rate distortion optimization way. To reduce the computational complexity of entire encoding process, it is necessary to optimize the traversing of all CU [1].

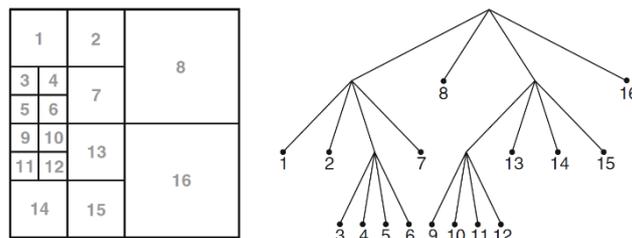


Figure 1 CTU structure

There have been many investigations on fast CU size decision. In [2], a fast CU size decision scheme is proposed by utilizing the depth correlation of neighbouring CTUs. [3] proposes a fast CU decision method with decision tree. A complexity reduction approach for intra mode HEVC is

proposed in [4], which learns a deep convolutional neural network model to predict CTU partition instead of RDO. In this paper, we propose a fast CU size decision algorithm for intra coding based on the SVM classification. Firstly, we present some effective features, such as block flatness (BF) and variation. Then based on those features, a SVM classifier is used to execute early termination and skip ahead. Finally, we can achieve effectually speed-up.

The rest of this paper is organized as follows. Section 2 elaborates the proposed method. Section 3 shows experimental results. The conclusion is given in section 4.

2. Proposed Method

According to the above analysis, HEVC adopts different CU sizes to achieve better coding gain. CU size decision is such a time-consuming procedure. This section will introduce our machine learning based fast CU size decision algorithm. Instead of designing with traditional statistical method, machine learning is utilized to avoid manually developing partition criteria. Feature is extracted from CU luminance data. SVM is learned to predict CU splitting and non-splitting. We use CPIH dataset [5], which is a dataset for HEVC CU splitting under intra mode encoding. The details is described as follows.

2.1.1. CPIH Dataset

CPIH dataset [5] provides a large-scale dataset for CU splitting of HEVC intra-mode. There are 2000 images with resolution of 4928*3264 selecting from Raw Image Dataset. Those raw images are divided as training set (1700 images), testing set (200 images) and validation set (100 images). Besides, down-sampling is executed on the 4928*3264 frames so as to generate different resolutions and ensure sufficient data for training. Next, all images are encoded by HM 16.0 with QP equals to 22, 27, 32 and 37 to get the splitting (1) and non-splitting (0) flags. In total, the CPIH dataset contains 12 sub-datasets with the combination of 4 QPs and 3 CU sizes (64×64 , 32×32 , 16×16).

2.1.2. Feature Selection

Adjacent regions of an image have high local similarity. Intra prediction fully takes advantage of the local similarity of image and predicts the pixels of current coding block with neighbour reconstructed pixels. The residual coefficients are generated from the original image and prediction image. Besides, residual coefficients are more efficient to encode than the original data. Better prediction brings less residual coefficients in a CU and the encoder prefers to choose larger size CU to save bit rates in this case. Blocks with flat and simple content are easier to get better prediction result. Figure 2 shows the CU structure of one frame in “RaceHorses”, which is encoded by HM16.0. The flat area with less texture information, like the grass and part of horses, is encoded as large CU, while the rough area with rich texture is partitioned into smaller units.

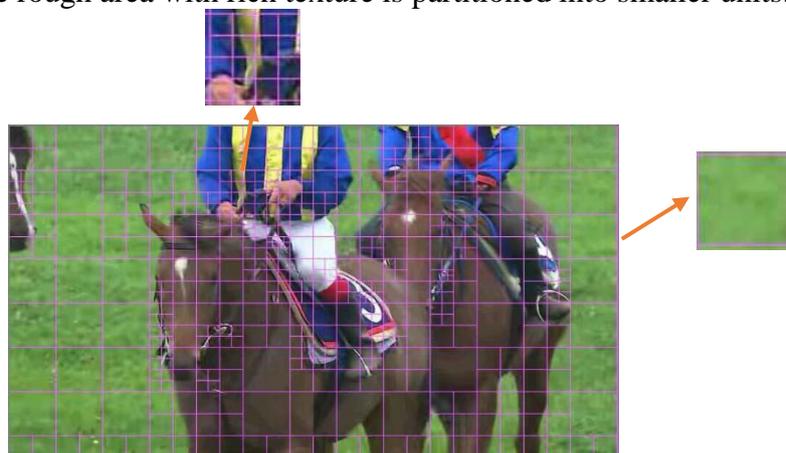


Figure 2 CU structure of “RaceHorses”

According to the former analysis, we select two main features of CU. One feature is variation, which is calculated in Eq.1. There have been many algorithms based on variance characteristics for video complexity analysis on H.264/AVC coding standard.

$$Var = \frac{1}{N^2} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} (X(i, j) - \bar{X}(i, j))^2 \quad (1a)$$

$$\bar{X}(i, j) = \frac{1}{N^2} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} X(i, j) \quad (1b)$$

where N represents CU size. $X(i, j)$ is the original pixel of luminance components. $\bar{X}(i, j)$ denotes the average result of the original pixels in a CU. Another feature is BF [6], which can also be interpreted as the ratio of DC component in Eq.2, Eq.3, Eq.4 and Eq.5.

$$DC = \frac{\sum_{i=0}^{N-1} \sum_{j=0}^{N-1} X(i, j)}{N^2} \quad (2)$$

$$Energy(DC) = \frac{(\sum_{i=0}^{N-1} \sum_{j=0}^{N-1} X(i, j))^2}{N^4} \quad (3)$$

$$Energy(Total) = \frac{\sum_{i=0}^{N-1} \sum_{j=0}^{N-1} X^2(i, j)}{N^2} \quad (4)$$

$$BF = \frac{Energy(DC)}{Energy(Total)} = \frac{(\sum_{i=0}^{N-1} \sum_{j=0}^{N-1} X(i, j))^2}{(\sum_{i=0}^{N-1} \sum_{j=0}^{N-1} X^2(i, j)) \times N^2} \quad (5)$$

Figure 3 shows four CUs with different BF. When BF is infinitely close to 1, CU is flat. If BF is much lower than 1, CU contains more texture.

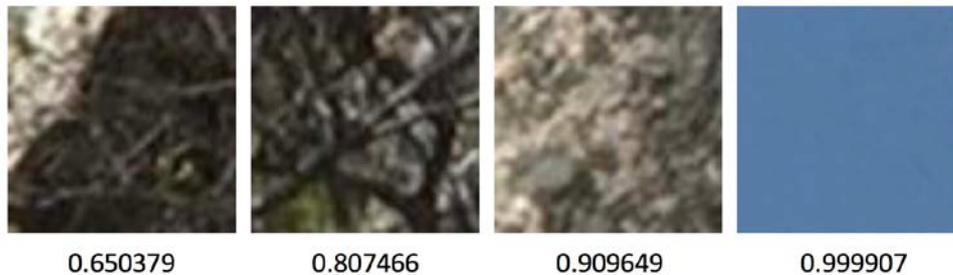


Figure 3 BF of different CUs

2.1.3.SVM and CU Size Decision

SVM is a widely used machine learning tool and it is a supervised classification algorithm based on finding the optimal margin. A decision boundary, also called the separating hyperplane is generated to classify samples into different classes in an N -dimensional space. In this paper, CU size decision problem can be simply regarded as a binary classification problem, and the hyperplane is describe as Eq.6.

$$w^T x + b = 0 \quad (6)$$

where x denotes features and w is the coefficients vector of features. b is an intercept term. The hyperplane is trained from the training set. Then training samples label is “1” and “0”. “1”denotes splitting the current CU and sub-CU might be a better choice, thus the processing of current CU can be skipped. “0” represents non-splitting, which means the further CU recursive can be avoided. In general, splitting or non-splitting may not be perfectly separable in the feature space. Misclassification will directly cause encoding performance loss.

Assuming that splitting is positive samples and non-splitting is negative samples. Table 1 shows the notations of FNR and FPR. The Euclidean distance of (FNR, FPR) is calculated to represent the classification error rate. To further analysis the relationship between classification error rate and coding performance, firstly, we train a SVM binary classifier of CU size decision and calculate the classification error rate. Secondly, we integrated it to HM16.0. 18 JCT-VC recommended sequences is tested to get the performance loss (BD-rate). Finally we get the scatter plot in Figure 4 of classification error rate of CU size decision and corresponding coding performance loss of the CU size decision algorithm. Higher error rate directly leads to worse coding performance.

Table 1 Notations to describe error rate

Notations	Full Name	Descriptions
TP	True Positive	True decision of splitting
TN	True Negative	True decision of non-splitting
FP	False Positive	False decision of splitting
FN	False Negative	False decision of non-splitting
FNR	False Negative Rate	$FN/(FN+TP)$
FPR	False Positive Rate	$FP/(FP+TN)$

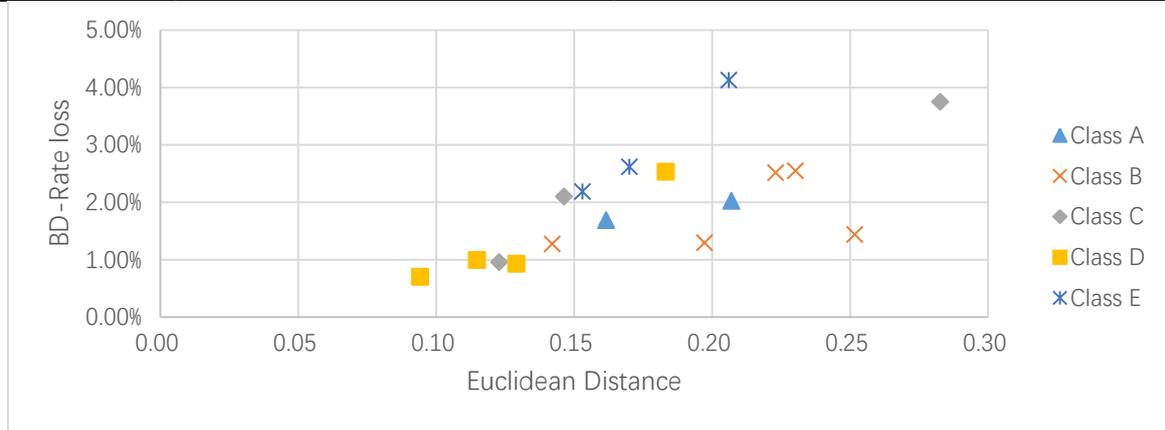


Figure 4 Relationship between Euclidean distance and BD-rate loss

To balance the performance of encoding and computational complexity, we choose weighted SVM with linear kernel. Penalty parameter is introduced to minimize the classification error separately. Equation of weighted SVM is shown as Eq. 7, which is also a three-output model.

$$\min \left\{ \frac{1}{2} w^t w + C (W^+ \sum_{i=1}^{n^+} \xi_i + W^- \sum_{j=1}^{n^-} \xi_j) \right\} \quad (7)$$

$$s. t. y_i (w^T x + b) \geq 1 - \xi_i, \xi_i \geq 0 \forall (x_i, y_i) \in n$$

where W^+ and W^- represent the weighted for positive error and negative error, respectively. C is a penalty parameter [7]. The flow of CU size decision is designed as Figure 5. Experimental results are shown in chapter 4

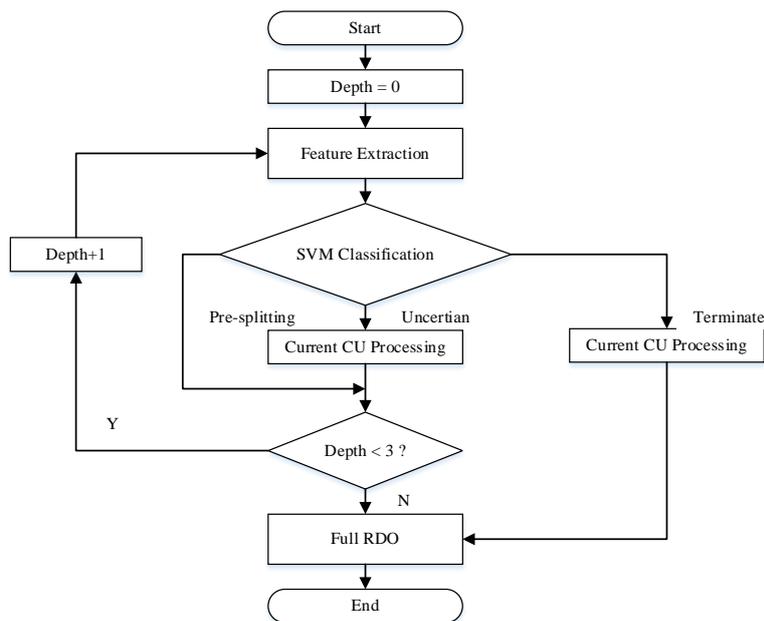


Figure 5 Flowchart of proposed fast cu decision method

3. Experimental Results

Performance of the proposed fast CU size decision method is verified on HEVC reference software HM16.0. JCT-VC recommended sequences (Class A to Class E) are tested under the common test conditions in [8]. All Intra (AI) configurations are applied. QP is set as 22, 27, 32 and 37. To evaluate the coding performance, we use the popular method in [9] to calculate the difference between two rate-distortion curves, which is also well known as BD-rate. Time Saving (TS) of quantization process is defined as Eq.8.

$$TS = \frac{Time_{anchor} - Time_{proposed}}{Time_{anchor}} \times 100\% \quad (8)$$

where $Time_{anchor}$ and $Time_{Proposed}$ denote encoding time consumptions of the method in original HM16.0 and the proposed methods, respectively. We also use an efficiency factor E to measure the coding efficiency of the proposed algorithm, which is calculated as Eq.9.

$$E = \frac{TS}{BD_rate} \quad (9)$$

where TS indicates the encoding complexity reduction and BD_rate denotes the coding performance loss. Efficiency factor E can reflect the encoding time saving per BD-rate loss [10]. The method with more computational complexity reduction under the same BD-rate loss tends to have better efficiency.

Table 2 shows the performance of our proposed method. A negative BD-rate relates to coding gain against the anchor and vice versa. A positive TS represents the time saving. Compared with HM 16.0 anchor model, our proposed method reduces the average encoding time by 40.23%, with only 0.83% BD-rate loss on average. The efficiency factor E shows better result when compared with the approach in [4].

Table 2 Performance of the proposed method and that in [4].

Sequences		Previous [4]		Proposed	
		<i>BD-Rate</i>	<i>TS</i>	<i>BD-Rate</i>	<i>TS</i>
Class A	Traffic	2.76%	70.04%	0.97%	40.17%
	PeopleOnStreet	2.16%	62.98%	0.97%	41.12%
Class B	Kimono	2.15%	83.64%	0.85%	52.09%
	ParkScene	2.13%	70.79%	0.58%	38.54%
	Cactus	2.11%	66.37%	1.34%	40.35%
	BasketballDrive	4.60%	75.78%	0.84%	45.85%
	BQTerrace	1.47%	60.04%	0.74%	43.82%
Class C	BasketballDrill	2.97%	58.20%	1.34%	37.65%
	BQMall	1.27%	54.69%	1.17%	31.96%
	PartyScene	0.50%	40.17%	0.29%	23.60%
	RaceHorses	1.75%	57.50%	0.29%	32.52%
Class D	BasketballPass	2.20%	57.50%	0.58%	37.68%
	BQSquare	0.19%	46.98%	0.41%	24.26%
	BlowingBubbles	0.68%	47.11%	0.33%	37.32%
	RaceHorses	1.23%	50.93%	0.31%	27.99%
Class E	FourPeople	3.12%	67.49	1.39%	44.70%
	Jhonny	3.60%	76.32%	1.51%	58.08%
	KristenAndSara	3.21%	74.01%	1.08%	56.60%
Average		2.12%	62.25%	0.83%	40.23%
E		29.36		48.47	

4. Conclusions

In this paper, we propose a machine learning based approach to simplify the computational complexity of CU size decision for HEVC all intra encoding. CU features are extracted and then we

train a three-output classification model for CU size decision, which ensures higher classification accuracy. The classification model is utilized in HM16.0. Experimental results show that our proposed method can achieve 40.23% encoding time saving on average and the BD-rate loss is only 0.83%. Besides, when considering the time saving of per BD-rate loss, our proposed method shows better efficiency compared with other fast CU size decision method.

Acknowledgements

This work is partially supported by National Key Research and Development Program of China under contract No. 2016YFB0401904.

References

- [1]Sullivan G J, Ohm J R, Han W J, et al. Overview of the High Efficiency Video Coding (HEVC) Standard[J]. *IEEE Transactions on Circuits & Systems for Video Technology*, 2012, 22(12):1649-1668.
- [2]L. Shen, Z. Zhang, and P. An, "Fast CU size decision and mode decision algorithm for HEVC intra coding," *IEEE Transactions on Consumer Electronics*, vol.59, iss.1, pp.207-213, Feb. 2013
- [3]D. Ruiz-Coll, V. Adzic, G. Fernández-Escribano, H. Kalva, J. L. Martínez and P. Cuenca, "Fast partitioning algorithm for HEVC Intra frame coding using machine learning," 2014 *IEEE International Conference on Image Processing (ICIP)*, Paris, 2014, pp. 4112-4116.
- [4]T. Li, M. Xu and X. Deng, "A deep convolutional neural network approach for complexity reduction on intra-mode HEVC," *2017 IEEE International Conference on Multimedia and Expo (ICME)*, Hong Kong, 2017, pp. 1255-1260.
- [5] Tianyi Li, Mai Xu and Xin Deng. A Deep Convolutional Neural Network Approach for Complexity Reduction on Intra-mode HEVC. *IEEE International Conference on Multimedia and Expo (ICME)*, 2017.
- [6]J. Li, F. Luo, Y. Zhou, S. Wang, M. Wang and S. Ma, "Content Based Fast Intra Coding for AVS2," *2017 IEEE Third International Conference on Multimedia Big Data (BigMM)*, Laguna Hills, CA, 2017, pp. 94-97.
- [7]Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin, "Liblinear: A library for large linear classification," *The Journal of Machine Learning Research*, vol. 9, pp. 1871–1874, 2008.
- [8]G. Correa, P. Assuncao, L. Agostini and L. Cruz, "Performance and Computational Complexity Assessment of High Efficiency Video Encoders", *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no.12, pp. 1899-1909, 2012.
- [9]G. Bjontegaard, "Calculation of average PSNR difference between RD-curves," in *Proc. ITU-T Q.6/SG16 VCEG 13th Meeting*, Austin, TX, Apr. 2001, document VCEG-M33.
- [10]S. Wang, S. Ma, X. Jiang, J. Fan, D. Zhao and W. Gao, "A fast intra optimization algorithm for HEVC," 2014 *IEEE Visual Communications and Image Processing Conference*, Valletta, 2014, pp. 241-244.