AN EFFICIENT INTRA CODING ALGORITHM BASED ON STATISTICAL LEARNING FOR SCREEN CONTENT CODING

Hao Yang, Liquan Shen, Ping An

School of Communication and Information Engineering, Shanghai University
Shanghai 200072, China

ABSTRACT

Screen content has different characteristics compared with natural content captured by cameras. To achieve more efficient compression, some new coding tools have been developed in the High Efficiency Video Coding (HEVC) Screen Content Coding (SCC) Extension, which also increase the computational complexity of encoder. In this paper, complexity analysis are first conducted to explore the distribution of complexities. Then, two classification trees, including early coding units (CU) partition tree (EPT) and CU content classification tree (CCT), are designed based on statistical characteristics and coding information. EPT is used to decide whether the CU skip the mode decision process of current depth level and CCT is used to classify the blocks into either natural blocks or screen blocks. Natural blocks will skip screen coding modes and screen blocks skip normal intra modes. Experimental results show the proposed algorithm can save 49% encoding time with 2.7% BD-rate increase on average for All Intra configuration under the SCC common test condition.

***Index Terms*—** Screen Content Coding, Machine Learning, Fast Algorithm, Intra Prediction.

1. INTRODUCTION

Unlike natural videos captured by cameras, Screen content video is generated by computer, which contain graphics, texts and mix content. To compress screen content more efficiently, Joint Collaborative Team on Video Coding (JCT-VC) developed screen content coding standard based on HEVC [2]. Two new intra coding modes are developed for SCC, which is known as Intra Block Copy (IntraBC) [3]and Palette (PLT) [4]. These tools get significant coding gain, however, additional rate-distortion optimization (RDO) procedure imposes enormous computation burden on encoder. Therefore, several fast algorithms have been proposed to reduce the complexity of encoder, which can mainly be categorized into three categories: fast intra mode decision, fast CU depth decision and fast search schedules.

The motivations for fast mode decision are based on the fact that the number of intra modes is large and most of them can be skipped. To overcome the heavy intra modes RDO process, HEVC adapts the rough modes decision (RMD) and most probable modes (MPMs) [5] strategy to construct the mode candidate list, and all the modes in the list will carry through RDO to choose the best intra prediction mode. A gradient-based fast mode decision method is proposed in [6], which uses the gradient information of each CU to reduce the number of intra candidates before mode selection. In [7], the average gradients in the horizontal direction and the vertical direction are calculated to decide the rough range of block modes.

In literatures [8] [9], Shen et al. propose the dynamic CU depth range and early termination methods on the basis of the CU information of neighboring and co-located blocks. Recently, machine learning methods are employed to solve the CU partition problem. Decision tree is used as a data classifier in [10]. In [11], a three-output joint classifier consisting of multiple SVM classifiers is utilized, and the parameters of SVM are derived by a sophisticated RD complexity model. A neural network (NN) based fast CU partition algorithm is proposed in [12], which uses the statistical features of CU to calculate CU partition decision.

Hash-based fast intra search algorithm has been adapted to SCC test model (SCM) [13], and the entire picture region is searched for the optimal block vector (BV) using hash search. Besides, [14] investigates a fast IBC search algorithm, which presents three fast IntraBC search methods based on the regular intra mode cost and CU activity.

The studies mentioned above have achieved significant coding performance, but most of them are proposed for HEVC, which do not consider the characteristics of screen content videos. For the methods proposed for SCC, their coding complexity is still too high to achieve real-time coding. In this paper, a more efficient method is proposed based on machine learning. Decision tree is chosen as the data classifier, and both screen content characteristics and encoding information are utilized as features. Firstly, a CU is classified as either partition block or non-partition block after the test of 2Nx2N intra mode. Partition block directly moves to next CU depth level, and non-partition block is categorized into either screen content block or natural content block. Screen content blocks only test screen coding modes, including IntraBC and

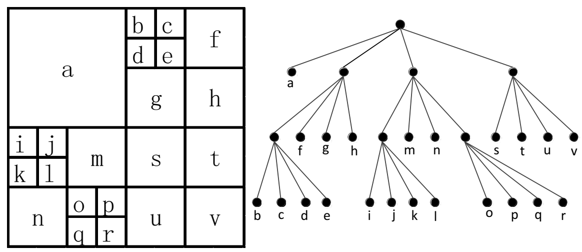

Fig. 1. Hierarchical Quadtree Structure

PLT, and natural blocks will skip screen coding modes.

The rest of the paper is organized as follows. In Section 2, we introduce the main coding tools in SCC that promote significant coding gain and complexity burden. Section 3 proposes two classifiers trained to accelerate encoding process by skipping unnecessary RDO calculation based on machine learning. Section 4 evaluates the performance of proposed algorithm and compares the proposed algorithm with the state-of-the-art fast algorithms. Section 5 concludes the paper.

2. SCM OVERVIEW

2.1. Hierarchical Quadtree Structure

To adapt to the diversity of image content, flexible quadtree partition is adopted into HEVC which enable the use of coding unit (CU), prediction unit (PU) and transform unit (TU). As an extension of HEVC, SCC shares the same recursive partitioning structure. Firstly, images are divided into many coding tree units (CTU), and each CTU can be partitioned into four square CUs, which can be further recursively partitioned until smallest CU size is reached, as shown in Fig.1.

PU is the basic unit to determine the optimal encoding modes for each CU level. There are two size of PU for intra CUs, i.e. 2Nx2N and NxN. It should be noted that CUs with different depth level may be coded in different prediction modes and different partition modes. Therefore, a "try all and find the best" method is employed to derive the optimal coding parameters. Obviously, this violent search strategy introduces significant complexity on SCM's encoder.

2.2. SCC New Coding Tools

In screen content compression, intra prediction process is improved by adopting three coding tools, known as Intra Block Copy, Palette Mode and Adaptive Color Transform (ACT) [15].

IntraBC finds the matching block from the reconstructed areas of current frame, and Block Vector between current block and matching block is signaled to decode side. Therefore, IntraBC can be treated as motion compensation within the current frame. In current HEVC-SCC test models, IntraBC and inter mode are unified, and IntraBC blocks are coded as inter PUs. This tool is very efficient for coding repeated patterns in text and graphics content within the same picture. But the search strategy introduces high computational complexity to encoder.

PLT is designed to suit the characteristics of screen contents where usually contain few color values with sharp edges. It enumerates these colors by building a color table, and an index is sent to indicate which color each pixel belongs to. ACT is used to remove inter-color component redundancy. An image block in the RGB or YUV color space can be coded directly, or adaptively converted to YCoCg color space during coding.

Combining the traditional intra modes with these new intra coding tools, HEVC-SCC is capable of performing with more than twice compression efficiency of HEVC [3]. However, the obtained coding efficiency is at the cost of high computational complexity due to the extra RDO process and matched blocks searching.

3. THE PROPOSED APPROACH

In HEVC, homogenous regions can be coded well by large size CU, and complex regions always be coded in small size CU. But for SCC, IntraBC and PLT rely on the patterns and colors that appears previously, and the local homogeneity cannot provide sufficient precision to decide whether a CU should be further split or not. To solve this problem, coding information is also under consideration in our algorithm. Since normal intra modes and screen coding tools are designed for different contents, we decide to separate natural contents and screen contents before testing corresponding modes in each CU depth level. According to the above analysis, two fast encoding strategies are proposed, and the flowchart is shown in Fig.2.

The proposed encoder framework includes two classifiers. For a coding block, fast IntraBC mode and 2Nx2N normal intra mode will be check first. Then, the features are extracted as the input of two classifier. The first classifier separates input blocks into partition blocks (P) and non-partition blocks (NP). The partition category completely bypasses the remaining mode selection process and directly moves to next CU depth level. For non-partition category, the input blocks will be classified into either screen content blocks (SCB) or natural content blocks (NCB) by the second classifier. Natural content blocks continue to check remaining normal intra modes and skip screen modes (IBC and PLT). For screen content blocks, only screen modes are considered at current depth level.

3.1. Features Selection

The CU depth decision mainly depends on the texture of the video contents. Similarly, mode selection has high correlation with image pattern and distinct color numbers. Since the purpose of the paper is to reduce the computational complexity, the features extraction process should be with low complexity to avoid the computation overheads. Finally, ten features are chosen as follows.

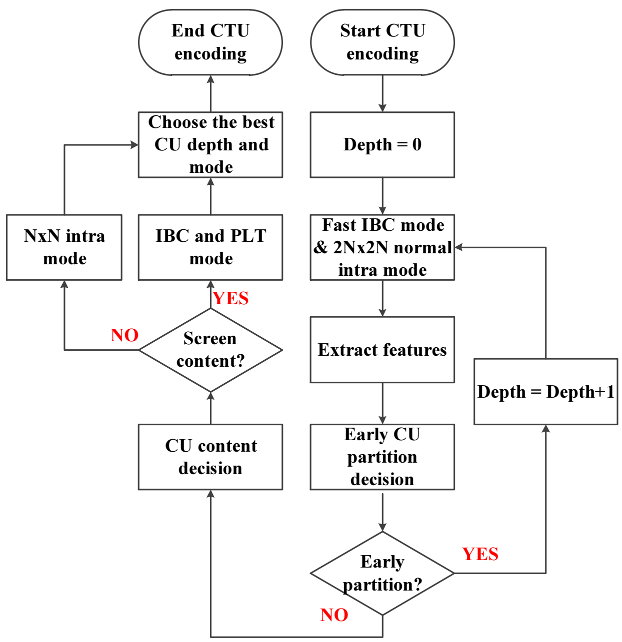

Fig. 2. Flowchart of the proposed algorithm

3.1.1. *Statistical Information of Current CU*

The maximum gradient magnitude (MGM) and variance (VAR) indicate the flatness of current CU, which have high influence on CU partition and mode decision. In this paper, gradient is calculated by Sobel operator. Color number (CN) and zero gradient number (ZGN), which denotes the number of pixel whose gradient equals to zero in a block, have strong correlation with screen content since a block with limited color and large ZGN is more likely to be screen content. Beyond these, we also select the minimum difference of four subCUs DC values (MDSCU) to indicate local smoothness of current block.

3.1.2. *Encoding and Context Information of Current CU*

Our proposed algorithm is implemented after the test of 2Nx2N intra mode, and the coding information in the test of fast IBC mode and 2Nx2N intra mode is helpful to the following classificatory decision. Thus, the RD cost (RDC) and coding bits (BIT) after testing 2Nx2N intra mode are utilized as features. Since coding parameters has strong spatial correlation, two features are extracted from neighbor CUs, including the sum of left PU partition flag and above PU partition flag (SLAPF) and the left CU depth (LCUD). To accelerate encoding process, the fast IBC search has adopted to SCM [16], which is performed before the evaluation of intra prediction mode. According to the experimental observation, if current CU chooses IntraBC after intra 2Nx2N test, it always select IntraBC after all modes checking. Thus, the fast IntraBC flag (FIBCF) is used to indicate whether the best mode is IntraBC after testing 2Nx2N mode.

3.2. Classifier Designing and Training

Aiming at getting more predicted precision, all the classifiers are trained separately for each CU size. Totally, 28160 samples of CU size 64x64, 112640 samples of CU size 32x32, 450560 samples of CU size 16x16 and 1802240 samples of CU 8x8 are obtained from 4 sequences including "Programming", "BasketballScreen", "Map" and "Robot" coding with the QP values of 22, 27, 32, 37, as the dataset. Dataset is equally divided into three parts including training set, validation set and test set.

To train the classification model, two machine learning methods are under consideration including support vector machine (SVM) and decision tree (DT). Since there are sufficient samples and less feature dimensions, both learning methods have similar classification accuracy. Compared with DT, SVM takes extra computation complexity for prediction, and DT is final chosen.

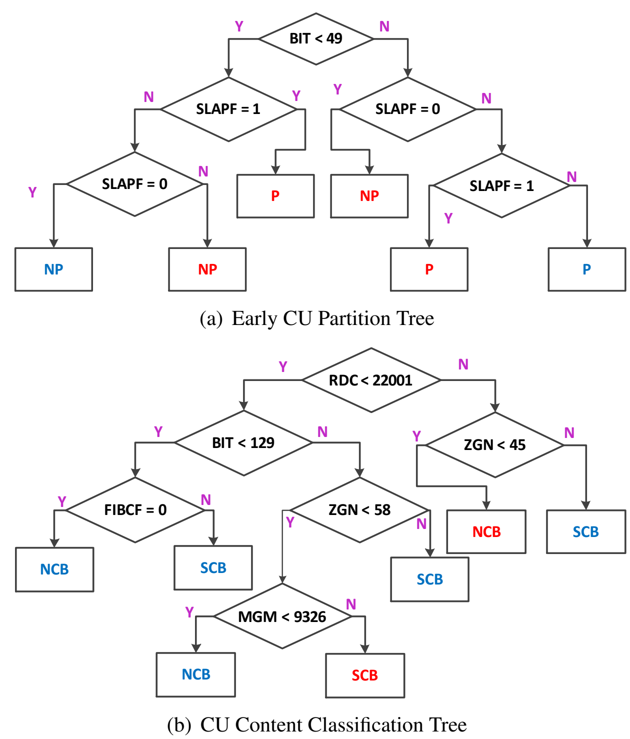

Fig. 3. Decision Tree of CU Size 16x16

The decision tree models are trained in MATLAB with "statistics Toolbox". All features are used for training EPT.

CCT is trained by all features expect SLAPF and LCUD. In order to avoid overfitting and make the model simpler, Post-Pruning strategy is adopted to prune the tree. Owing to space reasons, two trees of CU size 16x16 after pruning are provided in Fig.3, where diamond nodes represent judgment condition and rectangular nodes represent classification result. The split points of judgment condition are extracted by maximum information gain criterion.

Testing results of DT show that several leave nodes (shown in red) don't have sufficient classification accuracy. It means that some CUs may be categorized into a incorrect category thereby missing the optimal mode, which causes RD performance degeneration. Therefore, we determine that if a CU is classified into these uncertain leaves, full RDO tests are applied to it. When the classification accuracy of the leaves is lower than 85%, it will be marked as indeterminate block in implementation. It should be noted that the threshold is adaptive according to user's requirement. The lower of the threshold denotes more complexity reducing with more RD performance losing. It is the tradeoff between the complexity saving and RD performance.

4. EXPERIMENTAL RESULTS

In order to evaluate the performance of the proposed algorithm, we implemented it on the recent SCC test software SCM-5.0, and the experiment runs on Intel^R Core^{TM} i5-4590 CPU@3.30GHz platform.

The experiment is conducted under the SCC common test conditions [17] with All-Intra main configuration. The coding performance is measured by Bjonteggard Delta Bit Rate (BD-BR), Peak-Signal-Noise Ratio (BDPSNR) [18] and encoding time saving (TS), as defined in (1),

$$TS = \frac{T_s - T_p}{T_s} \times 100\%. \tag{1}$$

where T$_s$ and T$_p$ are encoding time of SCM-5.0 and the proposed algorithm, respectively. One hundred frames of eleven test sequences were employed in the coding experiment, respectively. All these frames were encoded with four QPs, which are 22, 27, 32 and 37. The TS for each sequence is calculated the average TS for four different QPs.

The coding performance comparisons between the proposed algorithm and the original SCM are shown in TABLE I. The sequences with " * " denote the training sequences. It can be seen that the proposed algorithm shows a consistent gain in encoding time saving for all sequences with the least gain of 41.27% in *FlyingGraphics* and the most gain of 70.65% in *SlideShow*. For all sequences, the proposed algorithm can save 48.89% encoding time with 2.72% BDBR loss on average.

To further evaluate the performance of the proposed method, we also compare our method with three state-of-the-art fast intra coding algorithms for screen content coding.

Table 1. Complexity and RD Performance Comparison with SCM-5.0

Sequence	Resolution	BDBR(%)	TS(%)
Robot*	1280 × 720	1.11	50.00
Programming*	1280 × 720	3.80	44.43
WebBrowsing	1280 × 720	2.49	47.62
SlideShow	1280 × 720	3.28	70.65
Map*	1280 × 720	2.93	43.83
Desktop	1920 × 1080	4.53	50.08
MissionControlClip3	1920 × 1080	1.79	45.04
FlyingGraphics	1920 × 1080	4.85	41.27
Kimono	1920 × 1080	0.40	52.40
BasketballScreen*	2560 × 1440	2.98	43.85
MissionControlClip2	2560 × 1440	1.80	48.66
Average	**–**	**2.72**	**48.89**

Table 2. The Comparisons between Our Proposed Method and Other Methods

	BDBR(%)	TS(%)
[12]	3.05	36.70
[19]	2.56	37.20
[20]	0.93	7.04
Proposed	2.72	48.89

The performance comparisons between the proposed method and others algorithms are shown in Table II. Compared with [12], the proposed method performs better in terms of BD-BR and TS. In [19], the BDBR increase is 2.56% which is little less than that of our method, but our method achieve additional 11.7% encoding time reduction. In [20], it has the smallest BDBR increase, but only 7.04% time saving is achieved, which is far below time saving of our method.

The results comparisons and analyses of the proposed algorithm demonstrates that the selected features and the two trained classifiers are effective on partition decision and content selection. The proposed fast intra coding method outperforms the state-of-the-art algorithms in terms of the coding efficiency or encoding complexity.

5. CONCLUSION

In this paper, we propose an efficient fast intra coding algorithm using machine learning for SCC. The proposed two classifiers are used to determine the CU partition decision and CU content category. Therefore, the skipped RDO on unnecessary CU depth levels and prediction modes brings 48.89% encoding time saving with 2.72% BDBR loss on average.

ACKNOWLEDGMENT

This work is supported by the National Natural Science Foundation of China under grant No. 61671282, 61422111 and U1301257, and sponsored by Shanghai Pujiang Program (15pjd015).

6. REFERENCES

[1] G. J. Sullivan, J.-R. Ohm, W.-J. Han, and T. Wiegand, Overview of the high efficiency video coding (HEVC) standard, IEEE Trans. Circuits Syst. Video Technol., vol. 22, no. 12, pp. 1648-1667, Dec. 2012.

[2] G. J. Sullivan et al., Standardized extensions of high efficiency video coding (HEVC), IEEE J. Sel. Topics Signal Process., vol. 7, no. 6, pp. 1001-1016, Dec. 2013.

[3] J. Chen et al., Description of Screen Content Coding Technology Proposal by Qualcomm, document JCTVC-Q0031, Apr. 2014.

[4] L. Guo, M. Karczewicz, and J. Sole, RCE3: Results of Test 3.1 on Palette Mode for Screen Content Coding, document JCTVC-N0247, Jul. 2013.

[5] L. Zhao, X. Fan, S. Ma, D. Zhao, Fast intra-encoding algorithm for high efficiency video coding, Signal Process.: Image Commun. , vol. 29, no. 10, pp. 935-944, 2014.

[6] Y. Zhang, Z. Li, and B. Li, Gradient-based fast decision for intra prediction in HEVC, in Proc. IEEE Vis. Commun. Image Process. (VCIP), San Diego, CA, USA, pp. 1-6, 2012.

[7] Tao Zhang, Ming-Ting Sun, Fast Intra Mode and CU Size Decision for HEVC, IEEE Trans. Circuits and Systems for Video Technology. , vol. pp., no.99, Apr. 2016.

[8] 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.

[9] L. Shen, Z. Zhang, and Z. Liu, Effective CU size decision for HEVC intracoding, IEEE Trans. Image Process., vol. 23, no. 10, pp. 4232-4241, Oct. 2014.

[10] D. Ruiz-Coll, V. Adzic, G. Fernandez-Escribano, H. Kalva, J.L. Martinez, and P. Cuenca, Fast partitioning algorithm for hevc intra frame coding using machine learning, IEEE International Conference on Image Processing (ICIP), pp. 4112-4116, Oct. 2014.

[11] Y. Zhang, S. Kwong, X. Wang, H. Yuan, Z. Pan, and L. Xu, Machine learning-based coding unit depth decisions for flexible complexity allocation in high efficiency video coding, IEEE Transactions on Image Processing, vol. 24, no. 7, pp. 2225-2238, July 2015.

[12] F. Duanmu, Z. Ma, and Y. Wang, Fast CU partition decision using machine learning for screen content compression, in Proc. IEEE Int. Conf. Image Process. (ICIP), Sep. 2015, pp. 4972-4976.

[13] B. Li and J. Xu, Hash-Based Motion Search, document JCTVC-Q0245, Mar. 2014.

[14] D.-K. Kwon and M. Budagavi, Fast intra block copy (IntraBC) search for HEVC screen content coding, in Proc. IEEE Int. Symp. Circuits Syst., pp. 9-12, Jun. 2014.

[15] L. Zhang, J. Chen, M. Karczewicz, B. Li, and J. Xu, Unification of Colour Transforms in ACT, document JCTVC-S0254, Strasbourg, France, Oct. 2014.

[16] R. Joshi, J. Xu, R. Cohen, S. Liu, and Y. Ye, Screen Content Coding Test Model 6 Encoder Description (SCM 6), document JCTVC-V1014, Oct. 2015.

[17] H. Yu, R. Cohen, K. Rapaka, and J. Xu, Common Test Conditions for Screen Content Coding, document JCTVC-X1015, Jun. 2016.

[18] G. Bjontegaard, Calculation of Average PSNR Differences Between RD Curves (VCEG-M33), document ITU-T SG16 Q.6, VCEG Meeting, Austin, TX, USA, Apr. 2001.

[19] Y. Kawakami, G. Chen, T. Ikenaga, Content based mode and depth skipping with Sharp and Directional Edges for intra prediction in Screen Content Coding, in IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp.46-49, Mar. 2016.

[20] S.-H. Tsang, Y.-L. Chan, and W.-C. Siu, Fast and efficient intra coding techniques for smooth regions in screen content coding based on boundary prediction samples, in IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 1409-1413, Apr. 2015.