Adaptive Quad-tree Complexity Control for HEVC

Kun Duan[1,2,3,4], Pengyu Liu[1,2,3,4,*], Kebin Jia[1,2,3,4,*] , Zeqi Feng[1,2,3,4] and Zhonghua Sun[1,2,3,4]
[1]Faculty of Information Technology, Beijing University of Technology, Beijing 100124 China
[2]Beijing Laboratory of Advanced Information Networks, Beijing 100124 China
[3]Beijing Advanced Innovation Center for Future Internet Technology, Beijing University of Technology, Beijing 100124 China
[4]Beijing Key Laboratory of Computational Intelligence and Intelligent System, Beijing University of Technology, Beijing 100124 China
[*]E-mail: liupengyu@bjut.edu.cn, kebinj@bjut.edu.cn

***Abstract*— To reduce the computational complexity of High Efficiency Video Coding (HEVC), this paper proposed an adaptive quad-tree complexity control mechanism. The proposed mechanism first defines similar region flag (SRF) to distinguish between similar region and non-similar region. Then, two algorithms, similar region depth range prediction algorithm and non-similar region depth range prediction algorithm, are proposed. The similar region depth range prediction algorithm estimates the feature of similar region based on coding unit (CU) depth of this region. The optimal depth of this region can be predicted. The non-similar region depth range prediction algorithm can skip low probability tree nodes based on depth correlation coefficient (DCC), which is calculated based on scene content change. Experimental results show that the pro-posed mechanism can reduce computational complexity by 28.17% on average with 0.75% Bjøntegaard delta bit rate (BDBR) increase and 0.03dB Bjøntegaard delta peak signal-to-noise rate (BDPSNR) drop under random access (RA) con-figuration. For low delay (LD) configuration, it can reduce complexity by 32.99% with 1.98% BDBR increase and 0.06dB BDPSNR drop penalty. The proposed mechanism is expected to be applied in the real-time environments.**

I. INTRODUCTION

In recent years, the storage and transmission of video data has become increasingly difficult. In pursuit of "lower bit-rate" and "higher quality" the Joint Collaborative Team on Video Coding (JCT-VC) released the next generation High Efficiency Video Coding (HEVC) standard [1] in January 2013. HEVC adopts many advanced coding tools. These tools bring significant improvement in compression efficiency. It achieves more than 50% coding bit-rate reduction with equivalent visual quality to that of the previous Advanced Video Coding (AVC) standard. But the coding complexity is approximately 5 times higher than AVC [2]. High coding complexity has an impact on the application of the video compression. By experiments in HEVC Test Model (HM), the exceedingly high computational complexity is mainly due to the flexible quad-tree structure based coding tree unit (CTU) partition in HEVC [3] instead of macroblock (MB) in AVC. HEVC broke 16×16 MB coding structure in AVC and adopted the quad-tree structured CTU depth traversal strategy, which allowed each CTU recursively splitting into four coding unit (CU) with depths from 0 to 3. Flexible quad-tree coding structure of HEVC is drawn in Fig. 1. Thus, it is important to reduce the complexity of quad-tree structure based CTU partition.

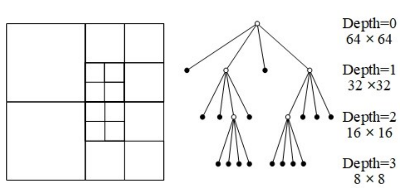

Fig. 1 Flexible quad-tree coding structure of HEVC.

To overcome this problem, several researches on HEVC fast algorithms have been proposed. Some researchers focus on flags based early termination methods [4] – [6]. Gweon [4] propose an early termination scheme of CU encoding based on coded block flag (CBF). Kim [5] utilize CBF and the differential motion vector (DMV) to do early termination. In [6], a threshold based fast CU termination method is proposed. However, these early termination methods all directly prune the quad-tree, which significantly reduce the coding performance. Furthermore, many works focus on low complexity method based on rate-distortion (RD) cost [7] – [9]. Lee [7] propose a fast CU decision algorithm based on Bayesian decision rule for assumed RD cost distribution; Kim [8] use assumed RD cost distribution modes to reflect spatial and temporal characteristics. Chen [9] propose an adaptive mode selection algorithm by offline statistics to control the coding complexity. However, due to it is difficult to fit the distribution of RD cost accurately, the coding time saving might decrease when the frame contained abundant texture and detail. Finally, some research uses spatial and temporal correlations to predict current CU size or partition depth [10] – [12]. Zhao et al. [10] present a fast mode decision algorithm. The depth situation of the colocated CTU in previous frame is used to predict the current CTU depth range. In [11], CTUs are firstly classified as static and motive and then depth range are determined respectively based on the size of collocated CTU. Zhou [12] proposes a fast coding unit depth algorithm using the spatio-temporal correlation of the depth information. However, most of these algorithms based on spatial and temporal correlations only use the CTU level spatio-temporal correlations to skip unnecessary CU size traversing, the work on spatio-temporal correlations analyzing in sub-CTU level are exceedingly rare.

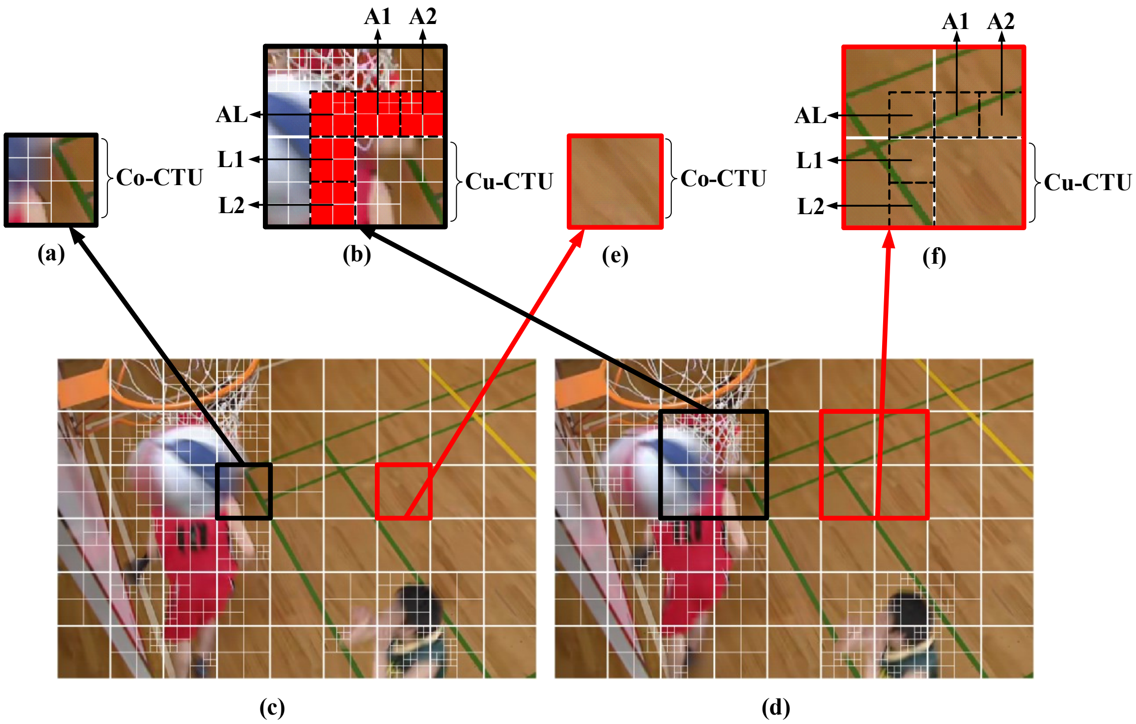

Fig. 2 Example of the spatio-temporal correlation.

To overcome the disadvantages of the above three kinds of fast methods, an adaptive quad-tree complexity control mechanism is proposed to further reduce the process of CTU partition complexity without noticeable coding efficiency loss. The similar region flag (SRF) is defined to distinguish between similar region and non-similar region. For these situations, similar region depth range prediction algorithm and non-similar region depth range prediction algorithm are utilized to predict depth range respectively.

The rest of the paper is organized as follows. Section 2 presents the motivation for the proposed method. Section 3 describes the proposed mechanism in detail. Experiment results are stated and discussed in Section 4. Finally, conclusion is given in section 5.

II. Motivations and Statistical Analysis

Video sequences have strong spatio-temporal correlation. More specifically, the optimal depth level of current CTU is the same or very close to the depth level of its spatially adjacent coded area and its co-located coded area in the previous frame. Thus, some specific depths can be kipped by using this spatio-temporal correlation. Fig. 2 shows two consecutive frames in a coded video sequence. The previous coded frame is represented with Fig. 2(c) and Fig. 2(d) represent current frame. Cu-CTU and Co-CTU represent current CTU and co-located CTU in previous coded frame. The content of the black square in Fig. 2(c) and Fig. 2(d) are shown with Fig. 2(a) and Fig. 2(b) respectively. It can be observed that the partition structure of Cu-CTU is highly similar to the partition structure of the Co-CTU [Fig. 2(a)] in temporal adjacent frame. Besides, the partition structure of Cu-CTU is very similar to the partition structure of spatially adjacent coded area around Cu-CTU [Fig. 2(b)] in current frame, especially in the coverage area with red [Fig. 2(b)] which is the 32×32 sub-CUs of the left, upper and upper left CTU of Cu-CTU. According to Fig. 2(b), A1 and A2 is defined as the third and fourth (Z order) sub-CUs of upper CTU; AL is defined as the fourth sub-CU of upper left CTU; L1 and L2 is defined as the first and third sub-CUs of left CTU. To further analysis the correlation between current CTU, it's Co-located CTU and its neighboring 32×32 sub-CUs, some experiments are executed in some video sequences which have different resolution and characteristics.

Table I shows the experimental results. Obviously, all 6 blocks have strong correlation with Cu-CTU. The correlation between Co-CTU and Cu-CTU is the highest. The second degree is A1, A2, L1 and L2, the correlation between these blocks and Cu-CTU has no significant differences. The last degree is AL, which just achieve 78.74% correlation with Cu-CTU. But the correlation slightly decreases when the coded video sequences have fierce motion, such as "RaceHorses" and "BQMall". Hence, the depth range of current CTU can be early predicted by its temporal and spatial adjacent coded blocks to simplify quad-tree structure based CTU partition in HEVC.

TABLE. I
CORRELATION BETWEEN CURRENT CTU AND ITS SPATIO-TEMPORAL BLOCK.

Class	Sequence	Correlation					
		Co-CTU	AL1	A1	A2	L1	L2
Class A	Traffic	80.73%	77.13%	79.75%	79.34%	80.78%	81.44%
	PeopleOnStreet	92.92%	81.35%	82.86%	83.07%	83.67%	84.31%
Class B	Cactus	85.04%	78.36%	80.56%	80.50%	80.13%	82.07%
	Kimono	81.69%	73.49%	75.71%	75.54%	75.28%	77.78%
	ParkScene	84.80%	76.41%	79.52%	79.40%	78.08%	79.31%
	BQTerrace	90.80%	79.53%	80.20%	80.71%	82.31%	83.97%
Class C	BasketballDril	88.32%	74.51%	80.45%	81.04%	78.45%	76.84%
	BQMall	87.17%	76.95%	79.67%	79.38%	79.44%	78.00%
	PartyScene	93.42%	83.25%	84.83%	84.78%	84.88%	81.11%
	RaceHorses	93.31%	81.06%	83.30%	83.33%	82.03%	78.42%
Class D	BlowingBubbles	96.42%	87.92%	87.29%	87.75%	88.78%	89.53%
	BQSquare	96.89%	86.92%	86.71%	85.71%	87.72%	88.34%
	RaceHorses	98.03%	89.46%	90.48%	91.81%	89.73%	89.18%
	BasketballPass	96.01%	80.34%	84.96%	83.53%	82.18%	83.70%
Class E	FourPeople	86.41%	77.66%	79.48%	79.58%	85.59%	83.01%
	Johnny	85.54%	82.96%	85.05%	84.41%	86.22%	83.71%
	KristenAndSara	85.86%	77.15%	81.11%	81.33%	81.72%	79.87%
Average		89.61%	80.26%	82.47%	82.42%	82.76%	82.39%

TABLE II
THE PERCENTAGE OF CU-CTU DEPTH IS EQUAL TO ITS SPATIO-TEMPORAL ADJACENT CTUS WHEN AL, A1, A2, L1, L2 AND CO-CTU HAVE SAME DEPTH.

Class	Sequence	Percentage
Class A	Traffic	97.98%
	PeopleOnStreet	96.70%
Class B	Cactus	94.66%
	Kimono	72.38%
	ParkScene	91.97%
	BQTerrace	93.46%
Class C	BasketballDril	91.13%
	BQMall	91.02%
	PartyScene	88.43%
	RaceHorses	89.03%
Class D	BlowingBubbles	89.55%
	BQSquare	87.67%
	RaceHorses	93.21%
	BasketballPass	95.93%
Class E	FourPeople	96.26%
	Johnny	96.72%
	KristenAndSara	94.59%
Average		91.81%

The content of the red square in Fig. 2(c) and Fig. 2(d) are shown with Fig. 2(e) and Fig. 2(f) respectively. It can be observed that the depth of Cu-CTU is equal to the depth of its spatio-temporal adjacent area when all spatio-temporal blocks have the same depth. To verify the universality of this situation, some statistics have been done. Table II shows the percentage of Cu-CTU depth is equal to its spatio-temporal adjacent blocks when AL, A1, A2, L1, L2 and Co-CTU have same depth. For the video with different feature, the average percentage is higher than 90%. Only the result of "Kimono" is lower than 88%. Therefore, the partition process can be accelerated by take advantage of the situation above.

III. PROPOSED ADAPTIVE QUAD-TREE COMPLEXITY CONTROL MECHANISM FIGURES AND TABLES

A. *References Define SRF to Distinguish Between Similar Region and Non-similar Region*

As stated above, Cu-CTU depth is equal to its spatio-temporal adjacent CTUs when AL, A1, A2, L1, L2 and Co-CTU have same depth. To father utilize this feature, SRF is defined and speed up coding process. Specifically, the SRF is activated when AL, A1, A2, L1, L2 and Co-CTU have same depth. The SRF is used to distinguish between similar region and non-similar region. For SRF is activated, Cu-CTU is located in similar region, the similar region depth range prediction algorithm is utilized to deal with this situation. For SRF is not activated, Cu-CTU is located in non-similar region, the non-similar region depth range prediction algorithm is introduced to deal with this situation. The proposed SRF can detect the feature of the region around Cu-CTU. It should be noted that the prerequisite for this SRF execution is that AL, A1, A2, L1, L2 and Co-CTU all exist.

B. *Similar Region Depth Range Prediction Algorithm*

When Cu-CTU is located in the similar region, the depth range of quad-tree traversal is determined according to the depth of these blocks. For different depth of Cu-CTU spatio-temporal blocks, predicted depth range of Cu-CTU is decided as follows:

(1) When the depth of AL, A1, A2, L1, L2 and Co-CTU all equal to 0, the traversal depth of Cu-CTU is chosen as 0. It means that the current CTU is located in the still or homogeneous motion region.

(2) When the depth of AL, A1, A2, L1, L2 and Co-CTU all equal to 1, the traversal depth of Cu-CTU is chosen as 1. It means the texture and motion in the area of current CTU is simple.

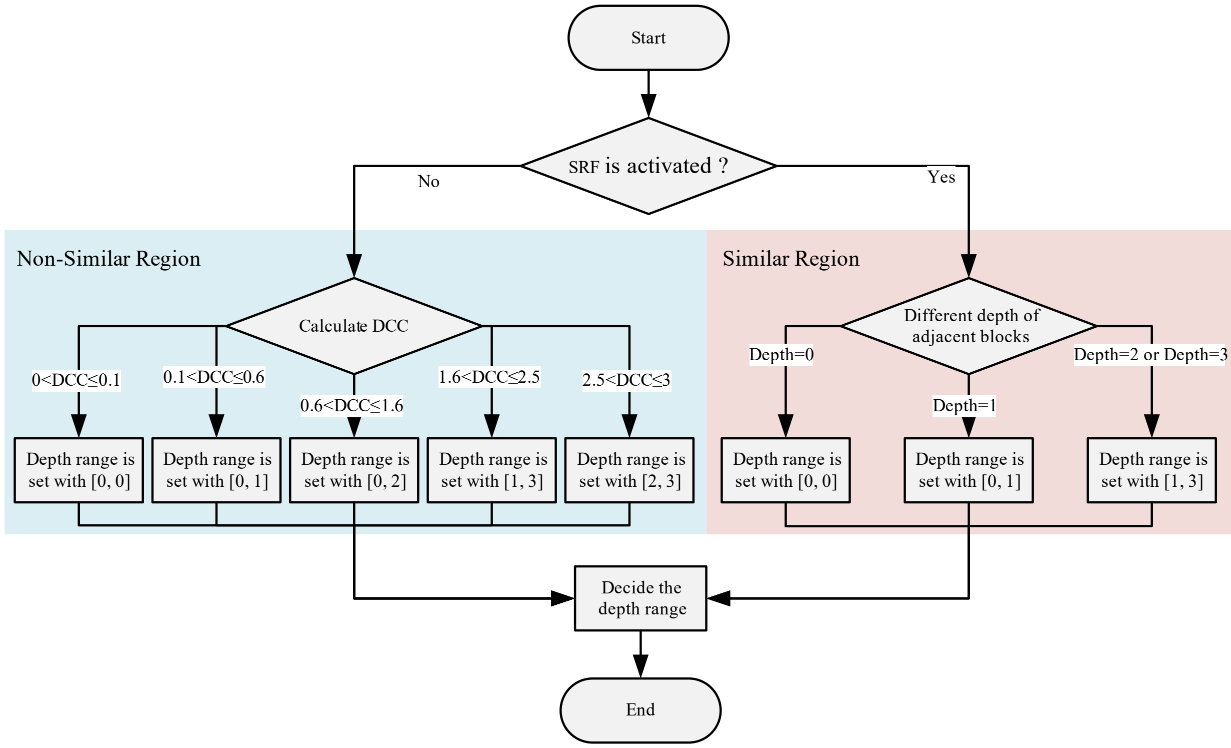

Fig. 3 Flowchart of the adaptive quad-tree complexity control mechanism.

(3) When the depth of AL, A1, A2, L1, L2 and Co-CTU all equal to 2 or 3, the traversal depth range of Cu-CTU is set to [1,3]. Under this situation, the quad-tree structure of current CTU is usually complicated. Thus, single prediction depth is disadvantage to rate distortion performance.

C. Non-similar Region Depth Range Prediction Algorithm

For the non-similar region, the optimal depth of some CTUs can be derived by the depth of its spatial neighboring blocks according to spatial-temporal correlation. Meanwhile, the depth information of co-located CTU significantly affects the depth of current CTU. According to the spatio-temporal correlation, some unnecessary depth of current CTU can be skipped to speed up the coding processing. Specifically, depth correlation coefficient (DCC) is introduced to analyze region properties and predict the optimal depth range of current CTU. DCC is calculated by using spatial neighboring blocks (AL, A1, A2, L1 and L2) and the co-located block (Co-CTU) at the previously coded frame. DCC is defined as follows:

$$DCC = \sum_{i=0}^{N-1} \xi_i \cdot \beta_i \tag{1}$$

where N is the number of spatio-temporal adjacent blocks (here, equal to 6). ξ is the weight factor of each block, which is based on correlations between the current CTU and its spatio-temporal neighboring blocks. The weight factor for Co-CTU is set to 0.3, the weight for A1 and A2 are set to 0.2, and the weight for AL, A2 and AL are set to 0.1. β is the value of depth. According to the value of DCC, the depth range is determined as follows.

(1) When 0<DCC≤0.1, content in current CTU is quite simple. The depth of Cu-CTU is set as 0.
(2) When 0.1<DCC≤0.6, the region of current CTU have slow motion. The depth of Cu-CTU is set to [0, 1].
(3) When 0.6<DCC≤1.6, the depth of Cu-CTU is set to [0, 2].
(4) When 1.6<DCC≤2.5, current CTU is located in the region with considerable motion or texture. The depth of Cu-CTU is set to [1, 3].
(5) When 2.5<DCC≤3, current CTU is located in the region with fast motion or complex texture. The depth of Cu-CTU is set to [2, 3]. (AL, A1, A2, L1 and L2) and the co-located block (Co-CTU) at the previously coded frame. DCC is defined as follows:

D. Overall Algorithm

Based on the analysis above, the flowchart of the adaptive quad-tree complexity control mechanism is shown in Fig. 3.

IV. EXPERIMENTAL RESULTS

To evaluate the performance of the proposed adaptive quad-tree complexity control mechanism, HM16.9 is used as encoder with two coding configurations random access (RA) case and low delay (LD) case. For both RA and LD, max CU size is set to 64 × 64 pixels; min CU size is set to 8×8 pixels; the initial motion search range is 64 in both horizontal and

TABLE IV
TEST SEQUENCES.

Class	Resolution	Sequence	Frame Rate
Class A	2560 × 1600	Traffic, PeopleOnStreet	30fps, 30fps
Class B	1920 × 1080	Cactus, Kimono, ParkScene, BQTerrace	50fps, 24fps, 24fps, 60fps
Class C	832 × 480	BasketballDril, BQMall, PartyScene, RaceHorses	50fps, 60fps, 50fps, 30fps
Class D	416 × 240	BlowingBubbles, BQSquare, RaceHorses, BasketballPass	50fps, 60fps, 30fps, 50fps
Class E	1280 × 720	FourPeople, Johnny	60fps, 60fps

TABLE III
EXPERIMENTAL CONDITION.

PC Configuration	
CPU	Intel® Core™ i7-3720QM 3.40GHz
RAM	8GB DDR3 1600 (Dual channel)
Software Environment	
Compiler	Visual Studio 2015
HM	HM-16.9

vertical directions, Fast motion Search and RD optimization are enabled; QP is chosen with 22, 27, 32 and 37. Table III shows the experimental condition. In addition, the test video sequences of various resolutions from 2560 × 1600 to 416 × 240, which recommended by JCT-VC, as shown in Table IV.

In this paper, the bjøntegaard delta peak signal-to-noise ratio (BDPSNR) and bjøntegaard delta bit rate [13] (BDBR) were used to evaluate the overall RD performance with four QPs, in addition, the ΔT is used to measure the time saving, which is calculated as follows:

$$\Delta T = \frac{Time_{HM16.9} - Time_{proposed}}{Time_{HM16.9}} \times 100\% \qquad (2)$$

where $Time_{HM16.9}$ represents the coding time of HM16.9 original method, $Time_{proposed}$ represents the coding time of the proposed fast method in this paper.

Table V shows the comparison results between the proposed algorithm and a state-of-the-art fast algorithm (Zhou's method [12]) under RA configuration. The average time saving of proposed algorithm achieve 28.17%, the BDBR only increased by 0.75% and the BDPSNR slightly decreased by 0.03dB. For different sequences, the proposed algorithm can greatly reduce the coding time with similar RD performance, with a maximum of 40.87% in "Traffic" and a minimum of 15.95% in "BlowingBubbles". Furthermore, the proposed algorithm performs better on high resolution, homogeneous or low motion sequences, such as "Traffic", "Kimono", "BQTerrace" and "BQMall". On the other hand, Zhao's method just achieves 19.89% time saving with 0.43% BDBR increased and 0.02dB BDPSNR reduced. Compared with Zhou's method, the proposed method has lower computational complexity with similar RD performance.

Table VI compares the proposed adaptive quad-tree complexity control mechanism with Zhou's method under LD configuration. It can be seen from Table VI that the proposed algorithm can greatly reduce the encoding time with similar encoding efficiency for all sequences. Specifically, average 32.99% coding time has been reduced in LD with maximum of 62.31% in "Johnny" and a minimum of 16.03% in "RaceHorses". Meanwhile, the RD performance loss is negligible, where the average increase of BDBR is 1.98% and the average drop of BDPSNR is 0.06dB. Similar with RA, the proposed algorithm has better performance with high resolution, homogeneous or low motion sequences in LD case. Especially for the sequence with nearly still background (sequences in "Class E"), the time saving is up to 50%. Compared with [12], the time saving of proposed algorithm is more than about 13% with negligible change of coding efficiency, especially in "Class B" and "Class E". The results of Table V and Table VI both indicate that the proposed adaptive quad-tree complexity control mechanism can fully utilize the coding information of spatio-temporal correlation between blocks and skip unnecessary depth traversal to efficiently reduce computational complexity in HEVC

TABLE. V
RESULT OF THE PROPOSED METHOD COMPARED TO [12] UNDER RA CONFIGURATION.

Class	Sequence	Proposed			Zhou's method [12]		
		BDBR (%)	BDPSNR (dB)	ΔT (%)	BDBR (%)	BDPSNR (dB)	ΔT (%)
Class A	Traffic	1.11	-0.03	40.87%	0.33	-0.01	23.61%
	PeopleOnStreet	0.85	-0.03	29.97%	0.82	-0.04	27.28%
Class B	Cactus	0.72	-0.01	36.32%	0.24	-0.01	23.22%
	Kimono	0.49	-0.01	35.80%	0.01	0.00	20.17%
	ParkScene	1.03	-0.03	35.11%	0.19	-0.01	21.81%
	BQTerrace	0.68	-0.01	41.46%	0.16	-0.01	21.97%
Class C	BasketballDril	1.47	-0.06	25.38%	0.64	-0.03	19.84%
	BQMall	1.00	-0.04	28.56%	0.78	-0.03	20.30%
	PartyScene	0.46	-0.01	25.81%	0.33	-0.01	21.11%
	RaceHorses	0.92	-0.03	23.76%	0.83	-0.03	19.72%
Class D	BlowingBubbles	0.35	-0.02	15.95%	0.03	0.00	14.53%
	BQSquare	0.18	-0.01	20.24%	0.14	-0.01	15.04%
	RaceHorses	0.55	-0.03	17.96%	0.90	-0.04	16.18%
	BasketballPass	0.62	-0.03	17.16%	0.59	-0.03	13.70%
Average		0.75	-0.03	28.17%	0.43	-0.02	19.89%

TABLE VI
RESULT OF THE PROPOSED METHOD COMPARED TO [12] UNDER LD CONFIGURATION.

Class	Sequence	Proposed			Zhou's method [12]		
		BDBR (%)	BDPSNR (dB)	ΔT (%)	BDBR (%)	BDPSNR (dB)	ΔT (%)
Class B	Cactus	2.71	-0.06	41.47%	0.60	-0.01	22.62%
	Kimono	0.76	-0.03	39.00%	0.07	-0.01	21.57%
	ParkScene	2.43	-0.08	39.00%	0.52	-0.02	21.01%
	BQTerrace	2.33	-0.05	46.21%	0.34	-0.01	22.97%
Class C	BasketballDril	3.13	-0.12	31.43%	1.21	-0.04	20.84%
	BQMall	2.32	-0.09	29.97%	1.01	-0.04	21.30%
	PartyScene	0.57	-0.02	24.52%	0.78	-0.03	21.91%
	RaceHorses	0.93	-0.03	23.10%	1.16	-0.04	22.12%
Class D	BlowingBubbles	0.23	-0.01	16.65%	0.91	-0.04	17.53%
	BQSquare	0.5	-0.02	18.06%	0.66	-0.02	16.34%
	RaceHorses	0.54	-0.03	16.03%	2.03	-0.09	17.98%
	BasketballPass	0.88	-0.04	17.09%	1.84	-0.09	15.70%
Class E	FourPeople	6.86	-0.14	56.96%	0.34	-0.01	23.54%
	Johnny	3.52	-0.08	62.31%	0.37	-0.01	23.95%
Average		1.98	-0.06	32.99%	0.85	-0.03	20.67%

encoder. Meanwhile, the proposed algorithm keeps better RD performance for all sequences.

V. CONCLUSIONS

In this paper, an adaptive quad-tree complexity control mechanism is proposed to reduce the computational complexity of HEVC encoder. In proposed mechanism, the SRF is defined to distinguish between similar region and non-similar region. Finally, similar region depth range prediction algorithm and non-similar region depth range prediction algorithm are introduced to speed up coding process. The proposed mechanism is implemented on the recent HEVC reference software HM16.9. Experimental results show that the proposed mechanism can significantly reduce the computational complexity of HM while maintaining almost the same RD performances as the original encoder. For RA configuration, the proposed mechanism can save 28.17% coding time with 0.75% BDBR increase and 0.03dB BDPSNR drop. For LD configuration, the proposed mechanism can save 32.99% coding time with 1.98% BDBR increase and 0.06dB BDPSNR drop.

ACKNOWLEDGMENT

This paper is supported by the Project for the National Natural Science Foundation of China under Grant No. 61672064, the Beijing Natural Science Foundation under Grant No. 4172001 and KZ201610005007, the China Postdoctoral Science Foundation under Grant Nos. 2016T90022 and 2015M580029, and the Beijing Postdoctoral Research Foundation under Grant No. 2015ZZ-23, Beijing Municipal Communications Commission Science and Technology Project under Grants No.2017058.

REFERENCES

[1] G. J. Sullivan *et al.*, "Overview of the High Efficiency Video Coding (HEVC) Standard," *IEEE Transactions on Circuits & Systems for Video Technology,* vol. 22, no. 12, pp. 1649-1668, Dec. 2012.

[2] G. Correa *et al.*, "Performance and Computational Complexity Assessment of High-Efficiency Video Encoders," *IEEE Transactions on Circuits & Systems for Video Technology,* vol. 22, no. 12, pp. 1899-1909, Dec. 2012.

[3] I. K. Kim *et al*., "Block Partitioning Structure in the HEVC Standard," *IEEE Transactions on Circuits & Systems for Video Technology,* vol. 22, no. 12, pp. 1697-1706, Dec. 2013.

[4] R. H. Gweon and Y. L. Lee, "Early Termination of CU Encoding to Reduce HEVC Complexity," Ieice Transactions on Fundamentals of Electronics Communications & Computer Sciences, vol. 95, No. 7, pp. 1215-1218, Jul. 2012.

[5] J. Kim *et al.*, "Early determination of mode decision for HEVC," in *Proc. Picture Coding Symposium (PCS2012),* Kraków, Poland, 2012, pp. 449–452.

[6] L. T. Hui, C. K. Chi and S. Rahardja, "Fast Coding Quad-Tree Decisions Using Prediction Residuals Statistics for High Efficiency Video Coding (HEVC)," *IEEE Transactions on Broadcasting,* vol. 62, no. 1, pp. 128-133, Mar. 2016.

[7] J. Lee *et al.*, "A fast cu size decision algorithm for HEVC," *Circuits & Systems for Video Technology IEEE Transactions on*, vol. 25, no. 3, pp. 411-421, Mar. 2015.

[8] H. S. Kim and R. H. Park, "Fast CU Partitioning Algorithm for HEVC Using an Online-Learning-Based Bayesian Decision Rule," *IEEE Transactions on Circuits & Systems for Video Technology,* vol. 26, no. 1, pp. 130-138, Jan. 2016.

[9] G. Chen, et al. "Complexity control algorithm based on adaptive mode selection for interframe coding in high efficiency video coding." *Journal of Electronic Imaging,* vol. 26, no. 4, p.043001, Jul. 2017.

[10] W. Zhao, T. Onoye and T. Song, "Hierarchical Structure-Based Fast Mode Decision for H.265/HEVC," *IEEE Transactions on Circuits & Systems for Video Technology,* vol. 25, no. 10, pp. 1651-1664, Oct. 2015.

[11] F. Chen et al, "A fast inter coding algorithm for HEVC based on texture and motion quad-tree models," *Signal Processing: Image Communication,* vol. 47, pp. 271-279, Sep. 2016.

[12] C. Zhou, "Spatio-temporal correlation-based fast coding unit depth decision for high efficiency video coding," *Journal of Electronic Imaging,* vol. 22, no. 4, p. 3001, Oct. 2013.

[13] G. Bjøntegaard, "Calculation of average PSNR differences between RD curves," in *Document VCEG-M33, 13th ITU-T SG16/Q6 VCEG meeting,* Austin, Texas, USA, 2001.