PERCEPTUAL FAST CU SIZE DECISION ALGORITHM FOR AVS2 INTRA CODING

Hao Xie[1], Guoqing Xiang[#2], Dunshan Yu[2], Hong Yu[1], Yuan Li[2], Wei Yan[1]*

[1] School of Software and Microelectronics of Peking University, Beijing 100871, China
[2]School of Electronics Engineering and Computer Science of Peking University, Beijing 100871, China
Email:{1701210416, gqxiang, yuds, yuhong123, yuanli}@pku.edu.cn, yanwei@ss.pku.edu.cn

ABSTRACT

AVS2 is a kind of video coding standard proposed by China, it adopts a flexible partition structure to improve coding performance, while the recursive quad-tree split coding unit (CU) structure brings a significant increase in coding complexity. In order to reduce the computational complexity of intra coding, quite a number of fast CU size decision algorithms have been studied. However, traditional algorithms have focused on reducing encoding time while remaining the objective performance without perceptual consideration. In order to obtain more time savings by taken of the subjective performance, therefore, in this paper, a perceptual CU size decision algorithm based on the spatial-temporal neighboring information and its internal perceptual texture information is proposed. Experimental results show that our approach can achieve 28.06% time complexity reduction on average under all intra testing configuration on RD17.0, while the BD-Rate loss is negligible with SSIM metric.

***Index Terms*—** AVS2, perceptual coding, CU size decision, intra coding

1. INTRODUCTION

[1]AVS2 is a kind of video coding standard independently proposed by the Audio Video Coding Standard Working Group of China [1]. Compared with the previous generation coding standard AVS or AVC/H.264, it can reduce 50% bitrate with barely changing the video quality. It also achieves competitive compression efficiency with the High Efficiency Video Coding (HEVC) [2].

Intra coding is one of the key technologies of AVS2. In analogies to HEVC, AVS2 adopts a flexible CU partitioning method to adapt to different video image contents [2]. The size of the CU is 64×64 at most and 8x8 at least. Firstly, a frame is split to a number of 64x64 CUs. Then the 64x64 CU is recursively split by depth-first traversal of the quad-tree until it reaches the smallest CU. Once the splitting of the CU hierarchical tree is finished, the leaf node CUs can be further split into PUs. Intra PU partitions in AVS2 vary in the set NxN, 2Nx2N, 0.5Nx2N, 2Nx0.5N. It can be seen that AVS2 not only exploits the square PU partitions as HEVC, but also utilizes the non-square PU partitions, which is called short distance intra prediction (SDIP). For edge areas with complex texture, better prediction can be achieved by SDIP with more coding complexity increasing. Besides, AVS2 has 33 prediction modes in the intra coding. With the above coding structures, the performance improvement of the intra coding is obvious but the complexity is increased drastically.

In order to reduce the computational complexity of the intra coding, quite a number of fast algorithms have been studied. Some of them focus on the fast CU size decision. In [3], the authors use the neighboring CU's rate distortion to determine whether to split the current CU. The algorithm can significantly reduce computational complexity while maintaining the performance. In [4], the local and global edge complexities are used to set threshold to assist in the judgment of CU's division, in which the encoding time is reduced by 52%, with 0.8% BD-Rate increasing. A fast CU size decision based on edge detection have been proposed to implement the early termination in [5]. The algorithm significantly reduces computational complexity by 69% on average with 2.99% BD-Rate increase for random access. Besides, machine learning has been applied to the fast CU size decision. In [6], the authors use convolutional neural network (CNN) and long- and short-term memory (LSTM) network to determine whether the current CU is split. However, compared with HEVC, the research of fast CU size decision for AVS2 intra coding is rarely. A fast CU splitting algorithm based on the content flatness is proposed in [7], which can achieve over 26.83% time complexity reduction on average under all intra testing configuration, while the average performance loss is 0.56%.

Almost all of the above methods mainly focus on objective performance degradation without subjective quality consideration. However, the objective quality, which is measured with PSNR, cannot reflect the ability of the human eye to distinguish image information well[8], which means the encoding complexity can be further reduced while remaining the subjective quality.

Therefore, in this paper, a perceptual fast CU size decision algorithm based on the spatial-temporal neighboring

[#]The co-first author, * the corresponding author

information and internal perceptual texture information is proposed. Firstly, the gradient magnitude similarity deviation (GMSD) model is used to represent the perceptual similarity between the current CU and its neighboring CUs. Based on achieved GMSD and the neighboring CUs' sizes, the current CU size can be estimated. Secondly, a just noticeable difference (JND) model to obtain the perceptual texture consistency of the current CU is utilized to determine whether the current CU should be split further. Experimental results show that our approach can achieve 28.06% time complexity reduction on average under all intra testing configuration on RD17.0 with negligible subjective performance loss.

The rest of the paper is organized as follows. The perceptual fast CU size decision algorithm is introduced in Section 2. Section 3 presents the experimental results of the proposed algorithm. Section 4 concludes the paper.

2. FAST CU SIZE DECISION ALGORITHM

Considering the subjective encoding quality is less sensitive to objective distortion than the objective encoding quality [9], which means the even though a CU with larger size can involve larger objective distortion, it can still remain similar perceptual distortion but a smaller JND value and better subjective quality when it includes a large number of texture contents that produce an obvious visual masking effect. Therefore, an effective perceptual fast CU size decision algorithm will have the opportunity to speed up the encoding process while keeps the subjective quality. In this section, two perceptual fast CU size decision methods are presented.

2.1. GMSD based Fast CU size decision

GMSD is an efficient image quality assessment (IQA) model proposed in [8], which can accurately compare the perceptual difference of two video images. The GMSD model is utilized in the proposed algorithm to calculate the perceptual correlation of the current CU with its spatial-temporal neighboring CUs, which is described as

$$GMSD = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(GMS(i) - GMSM)^2}, \quad (1)$$

where N is the number of pixels in the current CU, i denotes the pixel location and GMS is the gradient magnitude similarity calculated as

$$GMS(i) = \frac{2m_r(i)m_d(i)+c}{m_r^2(i)+m_d^2(i)+c}, \quad (2)$$

where m_r is the gradient of the current CU, m_d is the gradient of the neighboring CU, and c is a positive constant that supplies numerical stability.

The term $GMSM$ denoted gradient magnitude similarity mean is described as

$$GMSM = \frac{1}{N}\sum_{i=1}^{N} GMS(i). \quad (3)$$

More details can be found in [8]. It can be seen that the smaller the GMSD is, the larger perceptual similarity can be achieved for current CU and its neighbors.

As shown in Fig. 1, CU_1, CU_2, CU_3, CU_4 and CU_5 denote the spatially adjacent CUs of the current CU (CU_0), CU_6 denotes the temporally co-located adjacent CU in the previous frame. Considering that CUs in one video possess perceptual temporal and spatial correlations, the current CU is usually perceptually similar to the neighboring CUs. Based on the analyzation above, the current CU's splitting flag can be estimated from the neighboring CU which has the largest perceptual similarity to the current CU. Therefore, in this paper, we propose a threshold of GMSD, *TH*, to realize the fast CU size decision as follows.

(1) Firstly, GMSD of the current CU with the above six CUs are calculated before traversing the current CU splitting. Choose the minimum value of them to denote as $GMSD_{min}$ and compare it with *TH*.

(2) If $GMSD_{min}$ is smaller than *TH*, which means the CU is thought to be perceptually similar to the neighboring CU corresponding to $GMSD_{min}$. The current CU's splitting flag which is named FG, is obtained from the neighboring CU described above.

(3) If the $GMSD_{min}$ is larger than *TH*, the current CU will be analyzed later for further decision.

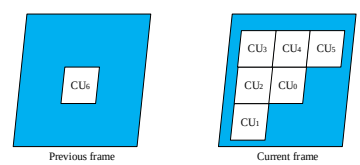


Fig. 1. The current CU and its spatial-temporal neighboring CUs

2.2. JND based CU size decision

The JND model is a measure of the distortion tolerance of the human visual system to the video image signal. The JND models are mainly divided into two categories: JND model in the pixel-domain, and the subband-domain. In this paper, we use an improved pixel-domain JND estimation model [10] that takes into account the effects of regularity of content in visual masking and improves the accuracy as calculated by

$$JND(x,y) = L_A(x,y) + V_M(x,y) - 0.3 \cdot \min\{L_A(x,y), V_M(x,y)\}, \quad (4)$$

where x and y represent the pixel positions, L_A is the luminance adaption factor and V_M is the visual masking. More details can be found in [10].

As JND variance of a CU can reflect the perceptual texture consistency [11], smaller JND variance value means that visual characteristics distribution of the CU is more homogeneous, and larger JND variance means that is richer. When the JND variance of four child CUs inside a CU differs a lot, it means the perceptual characteristics in the CU are

different a lot. So we should directly split it to the next coding depth. The CU-level perceptual distortion factor is defined as

$$P_F(d,k,i) = log_2\left(V_J(d,k,i)\right), \quad (5)$$

where d is the depth of the CU, k and i are the indices of the current CU and its child CUs, V_J is its JND variance.

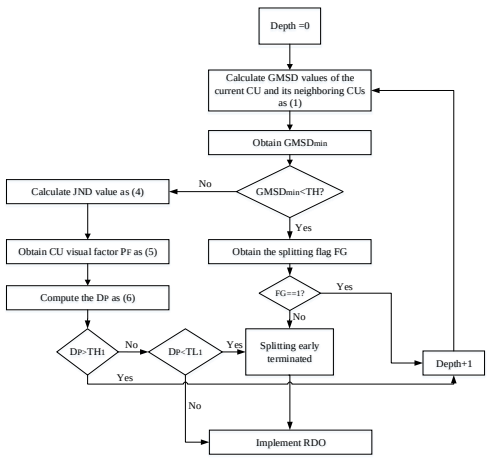


Fig. 2. The whole flow chart of proposed CU size decision algorithm

Table 1. Parameters for different QPs and different CU sizes

CU SIZE	QP	TH	TL_1	TH_1
64x64	27	0.15	0.001	0.001
	32		0.001	0.001
	38		0.001	0.001
	45		0.001	0.001
32x32	27	0.12	0.01	1.10
	32		0.01	1.10
	38		0.01	1.80
	45		0.02	2.00
16x16	27	0.14	0.02	100
	32		0.02	100
	38		0.03	100
	45		0.10	100

The factor P_F can reflect the perceptual characteristics of a CU, and different P_F means that the perceptual distortion of the CU is different. Then the maximum difference of the four child CUs' P_F inside the current CU is defined to show the perceptual difference as

$$D_P(d,k) = \max_{m,n\in\{1,2,3,4\}}\{|P_F(d,k,m) - P_F(d,k,n)|\}, \quad (6)$$

where m and n is the index of the child CUs, respectively.

In this paper, two thresholds of D_P, TH_1 and TL_1 are proposed to realize the further fast CU size decision for the CUs which are not decided by GMSD.

(1) Firstly, D_P is calculated before traversing the CU splitting. Then the CU is classified as perceptually texture homogeneous, rich, or unconfirmed by comparing D_P with TH_1 and TL_1 as follows.

(2) If the D_P of a CU is less than TL_1, which means the CU's perceptual characteristics distribution is thought to be homogeneous. Then the splitting process is terminated early.

(3) If the D_P is beyond TH_1, the CU's perceptual characteristics distribution is classified as rich, and it is directly split into smaller CUs. The coding process of the current CU can be skipped.

(4) Otherwise, the default encoding process is conducted for the current CU without any modification.

With extensive experiments, the TH, TL_1 and TH_1 for different QPs and different CU sizes are listed in Table 1.

Therefore, we can realize the perceptual fast CU size decision algorithm based on the spatial-temporal neighboring information and perceptual texture information. The whole flowchart of our proposed algorithm is showed in Fig. 2.

3. EXPERIMENTAL RESULTS

The test platform of this algorithm is the reference software RD17.0 of AVS2 [12]. The test sequence contains 18 sequences with various resolutions and contents as suggested for AVS2 [13]. For each sequence, 100 frames are encoded with QPs as 27, 32, 38 and 45 under the all intra configuration. The implemented platform is a server with Intel Xeon E5-2620 0 @ 2.00GHz and 32GB RAM. The anchor keeps the same as RD17.0 without any modifications. Since our algorithm is perceptual considering, the BD-Rate with SSIM as the quality metric for each sequence is computed. The coding complexity reduction is calculated as

$$TS = \frac{T_{anchor} - T_{pro}}{T_{anchor}} \times 100\%, \quad (7)$$

where TS denotes coding time reduction. T_{anchor} and T_{pro} denote the encoding time of the anchor and the proposed algorithm, respectively.

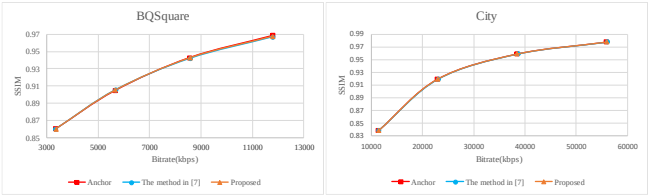
Besides, an efficiency factor [14] is introduced as equation (8). The proposed E reflects the complexity reduction per BD-Rate loss. The larger factor means that the coding efficiency of the proposed scheme is higher.

$$E = \frac{TS}{BD-Rate}. \quad (8)$$

Considering our goal is to reduce the intra coding complexity of AVS2, and to our best knowledge, there are few studies on this, here the method in [7] for AVS2 is selected for comparison.The experimental results as shown in Table 2 indicate the proposed algorithm of CU size decision is faster and more efficient than the anchor. It can be seen that the encoding time of the proposed algorithm has significantly been reduced with negligible BD-Rate loss. The performance comparison with the method in [7] is also given. Compared with it, the proposed algorithm's BD-Rate is 0.92%, which is less than that of the method in [7] while their time reductions are almost the same. What's more, the proposed algorithm presents a larger E, which is 31.11, which means the coding efficiency of the proposed algorithm is higher. Typical subjective performance comparisons are shown as Fig. 3 with the sequence BQSquare and City as the examples. It shows

Table 2. Time savings and BD-Rate with SSIM comparison

Resolution	Sequence	[7]		Proposed	
		BD-Rate[%]	TS[%]	BD-Rate[%]	TS[%]
1080P	Sunflower	1.86	23.78	0.33	17.44
	Pedestrian_area	2.05	29.71	1.62	26.69
	Kimono1	3.81	20.16	0.83	13.50
	Cactus	1.65	26.86	0.53	24.75
	BasketballDrive	0.66	30.15	0.44	28.25
WVGA	BasketballDrill	0.18	20.51	0.77	25.77
	BQMall	0.97	27.48	0.65	26.35
	PartyScene	0.08	23.13	0.59	33.00
	RaceHorses	0.04	21.60	0.04	12.33
WQVGA	BasketballPass	0.19	24.99	1.27	37.06
	BQSquare	0.38	37.24	0.20	37.48
	BlowingBubbles	0.14	21.03	0.66	24.69
	RaceHorses	0.01	21.22	0.00	9.96
720P	City	0.90	14.98	0.38	18.45
	Crew	3.34	43.39	0.79	34.94
	Vidyo1	3.58	39.41	3.03	46.23
	Vidyo3	4.65	42.27	2.12	42.98
	Vidyo4	0.74	39.55	1.81	45.15
Average value		1.40	28.19	0.92	28.06
E		20		31.11	



(a) (b)

Fig. 3. SSIM vs Rate comparison for different algorithms

Fig. 4. The representative region of RaceHorses @832×480

(a) Anchor (b) The method in [7] (c) Proposed algorithm

Fig. 5. CU partition structure comparisons on the representative region of sequence RaceHorses @832×480 with QP=38

that the proposed algorithm's sujective performance is very close to the anchor and the method in [7]. Besides, the objective performance comparisons based on PSNR are also conducted. The objective performance is almost not different. However, the efficiency factor E of the method in [7] is 98 and 16.64 for BQSquare and City respectively, while they are 187.4 and 48.55 of our proposed algorithm, respectively. It is shown that the proposed scheme can achieve more efficiencies than the method in [7].

Finally, in order to analyze the behavior of the present algorithm more comprehensive, the sequence RaceHorses with resolution 832×480 is selected as the representative as shown in Fig. 4. One typical 64x64 CU region marked as "A" in red are selected. For the region A, it includes a large number of texture contents that produce an obvious visual masking effect, resulting in a large JND value. This means that the human eye can't distinguish the details in the CU very well, so the CU can be compressed with larger size.

The impact of different algorithms on the CU partition structure are shown from Fig. 5(a) to Fig. 5(c). It can be seen that the anchor and the method in [7] generate refined CUs in region A. While as shown in Fig. 5(c), our proposed algorithm adopts coarse CUs in region A compared with the above two algorithms. Therefore, the algorithm proposed in this paper can have the opportunity to obtain more compression time reduction while remain the subjective quality for the human eyes.

4. CONCLUSION

The paper proposes a perceptual fast CU size decision for the intra coding of AVS2. It involves two methods. Firstly, the perceptual correlation of the current CU and its spatial-temporal neighboring CUs, which is characterized by GMSD, is exploited to estimate its size. Secondly, the JND model to obtain the perceptual texture consistency of the current CU is utilized to determine whether to split the current CU. Experiments are implemented in RD17.0 reference software of AVS2 and the results show that the proposed fast algorithm achieves 28.06% time reduction, while BD-Rate is 0.92% with SSIM. Compared with the method in [7], their encoding time reduction is similar, but the efficiency factor E of our proposed algorithm is improved significantly. The algorithm can also be compatible with other video coding standards such as HEVC/H.265 etc. In the future, we will explore more efficient fast algorithms.

ACKNOWLEDGMENT

This work is partially supported by the National Key Research and Development Program of China under contract No. 2016YFB0401904 the Guangdong science and technology project: AVS2 ultrahigh-definition TV front-end codec equipment development and application and the China Postdoctoral Science Foundation funded project (2018M641090).

5. REFERENCES

[1] Gao W , Ma S . An Overview of AVS2 Standard[M]// Advanced Video Coding Systems. Springer International Publishing, 2014: 35–49.

[2] 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 and Systems for Video Technology, 2012, 22(12):1649-1668.

[3] Shen L , Liu Z , Zhang X , et al. An Effective CU Size Decision Method for HEVC Encoders[J]. IEEE Transactions on Multimedia, 2013, 15(2):465-470.

[4] Min B , Cheung R C C . A Fast CU Size Decision Algorithm for the HEVC Intra Encoder[J]. IEEE Transactions on Circuits and Systems for Video Technology, 2015, 25(5):892-896.

[5] Belghith F , Kibeya H , Ben Ayed M A , et al. Fast coding unit partitioning method based on edge detection for HEVC intra-coding[J]. Signal, Image and Video Processing, 2016, 10(5):811-818.

[6] Mai X , Tianyi L , Zulin W , et al. Reducing Complexity of HEVC: A Deep Learning Approach[J]. IEEE Transactions on Image Processing, 2018:1-1.

[7] Li J , Luo F , Zhou Y , et al. Content Based Fast Intra Coding for AVS2[C]// 2017 IEEE Third International Conference on Multimedia Big Data (BigMM). IEEE Computer Society, 2017.

[8] Xue W , Zhang L , Mou X , et al. Gradient Magnitude Similarity Deviation: A Highly Efficient Perceptual Image Quality Index[J]. IEEE TRANSACTIONS ON IMAGE PROCESSING, 2014, 23(2):684-695.

[9] Xiang G , Jia H , Yang M , et al. A novel adaptive quantization method for video coding[J]. Multimedia Tools and Applications, 2018, 77(12):14817-14840.

[10] Wu J , Shi G , Lin W , et al. Enhanced just noticeable difference model with visual regularity consideration[C]// 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2016.

[11] Xiang G , Jia H , Yang M , et al. An Improved Adaptive Quantization Method Based on Perceptual CU Early Splitting for HEVC[J]. IEEE International Conference on Consumer Electronics (ICCE), 2017.

[12] http://www.avs.org.cn/

[13] X. Zheng, “Avs2-p2 common test conditions,” 48th AVS Meeting, AVS-N2047, Beijing, May, 2014.

[14] Wang S , Ma S , Jiang X , et al. A fast intra optimization algorithm for HEVC[C]// Visual Communications & Image Processing Conference. IEEE, 2015.