



Gradient Based Fast Intra Prediction Algorithm for VVC

Shiyu Wang^(✉) and Qiang Li

School of Communication and Information Engineering,
Chongqing University of Posts and Telecommunications, Chongqing, China
593948357@qq.com

Abstract. Versatile Video Coding (VVC) achieves better performance with about 40% bitrate reduction compared to H.265/HEVC under the same video quality. The improvement of VVC coding performance is at the cost of increased computational complexity of the VVC encoder. To address this issue, a fast algorithm for VVC intra prediction based on gradient information of coding unit (CU) is proposed in this paper. Experimental results illustrate that the proposed algorithm can save 47.78% coding time on average as compared with VTM7.0 with 2.25% increase in BDBR and 0.09 dB loss in PSNR.

Keywords: VVC · Intra prediction · Gradient · CU partition · Mode decision

1 Introduction

With the rapid development of video technology, people have higher requirements for the visual experience brought by video. High Definition (HD) and Ultra-High Definition (UHD) videos have drawn more attention since they can retain more abundant image details and provide more realistic visual effects. However, the network transmission of HD and UHD videos is faced with a great challenge due to the huge data volume brought by HD and UHD videos with the high resolution and extensive range of luminance. Therefore, there is an urgent need for a coding standard with better performance than the previous generation video standard including High Efficiency Video Coding (HEVC) [1] and other currently coding standards. In order to solve this problem, ISO/IEC Moving Picture Experts Group and ITU-T Video Image Experts Group have jointly established the Joint Video Experts Team (JVET) and have launched an emerging standard, namely Versatile Video Coding (VVC). JVET determined the first draft of the standard [2] and Video Test Model 1 (VTM1) [3] at the San Diego conference in the United States on April 10, 2018. The standardization of H.266/VVC is currently in progress, and it is expected to be completed by 2020.

As shown in Fig. 1, VVC still adopts the hybrid coding structure in HEVC, but new coding tools are introduced into each functional module. For the block partition structure, VVC adopts Quadtree with nested Multi-type tree (QTMT) structure and removes the separation of the CU, Prediction Unit (PU), and Transform Unit (TU) concepts [4]. For the intra prediction, VVC adopts the more refined intra prediction mode with 65 intra prediction directions [5] for square CUs and 85 intra

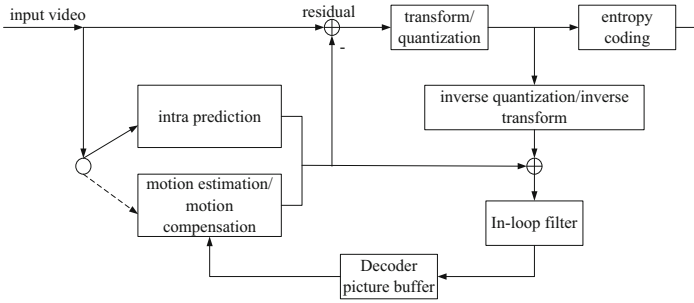


Fig. 1. H.266/VVC coding framework

prediction directions for rectangle CUs [6], to replace the 33 directions in square PUs in HEVC. In order to further improve the compression performance of intra prediction, plenty of advanced coding tools are implemented, e.g. Position-Dependent Intra Prediction Combination (PDPC) [7], Multi-Reference Line (MRL) [8] and Multiple Transform Selection (MTS) [9]. PDPC employs a weighted combination of filtered and unfiltered boundary reference samples for intra prediction. The MRL not only uses the nearest reference line as HEVC intra prediction but also uses two additional lines which near the nearest reference line. Although these advanced coding tools enhance the performance of VVC by increasing the compression ratio of VVC, they increase the coding complexity of the encoder. Under All-Intra (AI) test configuration, the intra coding complexity of VVC Test Model (VTM) is 18 times higher than that of HEVC test Model (HM) [10]. Thus, it is meaningful to committed to the research of the optimization of intra prediction in VVC.

There have been a couple of researches to reduce the intra prediction computational complexity with more time saving and negligible loss in BDBR and PSNR. The authors in [11] proposed the fast CU partition algorithm for VVC intra coding based on the correlation between the best partition type of the parent CU and that of its sub-CUs. In [12], a fast QTMT partition decision based on the distribution features of CU size and intra mode was proposed. This paper also introduced a fast intra mode decision algorithm based on gradient descent search. Ting-Lan Lin [13] predicted the depth range of the CU and made a fast decision of binary tree division according to the gradient information. In [14], a fast intra coding algorithm based on the spatial correlation of the CU and its neighboring LCUs was proposed. Although the algorithms mentioned above effectively reduce the encoding time for VCC intra prediction, there are still some defects and room for improvement. First, the samples selected by the fast algorithm based on statistics must have universal applicability. Second, when it comes to videos with complex texture, the performance of fast algorithms based on spatial correlation is significantly degraded.

The rest of the paper is organized as follows: Sect. 2 presents an overview of partition structure and intra prediction in VVC. Section 3 gives a detailed description of the fast intra prediction algorithm for VVC, including the fast CU partition algorithm and fast mode decision algorithm. Experimental results and comparisons with VTM7.0 are shown in Sect. 4. Finally, the conclusions are drawn in Sect. 5.

2 Overview of Partition Structure and Intra Prediction in VVC

2.1 Partition Structure

In order to better adapt to the local texture features of the image, VVC employs QTMT as block partition structure. The partition process of Coding Tree Unit (CTU) is as follows. The CTU is partitioned by Quadtree (QT) at first. Then, the leaf nodes in the previous step can be further partitioned by QT or multi-type tree (MT) until the maximum partition depth is reached.

As illustrated by Fig. 2, QTMT includes QT and MT which consist of Vertical Binary Splitting (SPLIT_BT_VER), Horizontal Binary Splitting (SPLIT_BT_HOR), Vertical Ternary Splitting (SPLIT_TT_VER), and Horizontal Ternary Splitting (SPLIT_TT_HOR). Figure 3 is given as an example of block partition in VVC. The QTMT partition structure of a CTU and the corresponding tree representation are shown in Fig. 3(a) and Fig. 3(b), respectively. The QT partition is indicated by solid black lines, and the MT partition is indicated by blue and red dotted lines.

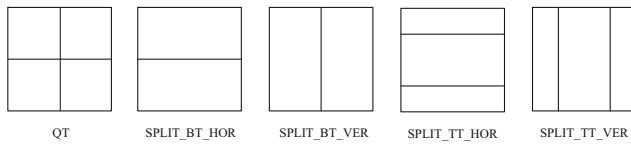
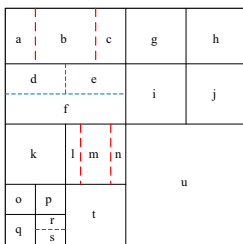
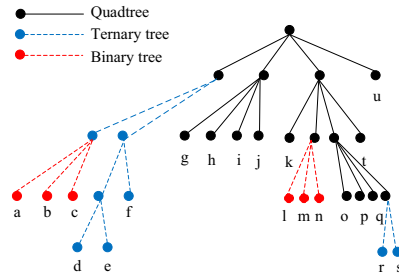


Fig. 2. Five partition types of QTMT structure



(a)



(b)

Fig. 3. An illustration of QTMT structure. (a) Example of QTMT structure, (b) The corresponding tree representation

There are some restrictions to simplify the CTU partition process. For example, QT is forbidden in the leaf nodes of MT and some redundant partition patterns could potentially result in the same coding block structure are disallowed. Besides, the sizes of the quad-tree and the multi-type tree are limited by several encoding parameters.

2.2 Intra Prediction

VVC adopts 67 intra prediction modes in square CU. Specifically, VVC retains the Planar mode and DC mode of HEVC and adds 32 directional modes which are denoted by red dotted arrows in Fig. 4. As shown in Fig. 4, the 65 directional prediction modes of VVC have the same direction range as the 33 directional modes of HEVC, but the granularity is smaller.

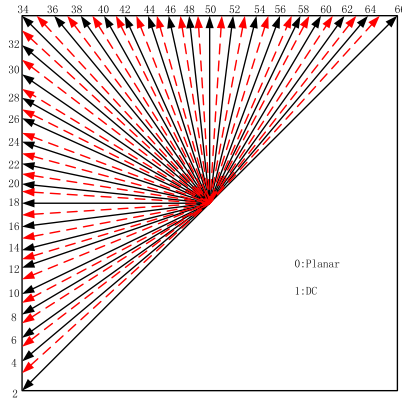


Fig. 4. 67 intra prediction modes

In order to simplify the intra prediction process of VVC, the classical three-stage fast intra mode decision is adopted. Firstly, N modes with least Hadmard cost ($HCost$) is selected through Rough Modes Decision (RMD) to construct the candidate list.

$$HCost = D_{SATD} + \lambda \cdot Bit_{mode} \tag{1}$$

Where, D_{SATD} is the Sum of Absolute Transform Difference (SATD) of residual signal, which represents the distortion of the current prediction mode, λ is the Lagrangian multiplier, and Bit_{mode} represents the number of bits that is required for encoding in the intra prediction mode. Secondly, Most Probable Modes (MPM) are taken into account to update the candidate list. Thirdly, 2 or 3 modes with less cost are involved in the Rate-Distortion Optimization (RDO) process [15] to select the least RDO cost ($RDCost$) mode which is the optimal mode for CU. The $RDCost$ is calculated by (2).

$$RDCost = SSE + \lambda \cdot Bit_{mode} \tag{2}$$

Where, the SSE is the sum of the squared errors between original CU and the reconstructed CU. Although the three-stage fast intra mode decision played a certain role in reducing the complexity of intra prediction, the intra prediction process is still time-consuming. To further reduce the intra prediction coding complexity, VVC adopts two step fast decision in RMD process. First, N modes with least Hadmard cost were

selected from 33 prediction modes with the same direction as HEVC. Then, the neighbor modes of the selected N modes are further checked by Hadmard cost.

3 Proposed Fast Intra Prediction Algorithm

3.1 Fast CU Partition Decision Algorithm

In order to reduce the computational burden of encoder through reducing the number of partition patterns that need to be traversed in CU recursive partition process, we explore the new partition types distribution of VTM7.0. We count the QTMT distribution information of six video sequences with different resolutions, motion characteristics and contents from each class. *FoodMarket* from Class A1 has rich scene changes, *CatRobot* from Class A2 with local motion, *BasketballDrive* from Class B with high motion, *PartyScene* from Class C with rich texture, *BQSquare* from Class D includes natural scenery and building structure and *FourPeople* from ClassE with a lot of static background area. Each sequence is coded with QP of 22, 27, 32, 37 and AI configuration. QT1 means QT depth is 1, QT2 means QT depth is 2, QT3 means QT depth is 3, QT4 means QT depth is 4, noMT means without MT.

Table 1. Partition types distribution for different sequences (%).

Sequence	QT1	QT2	QT2	QT3	QT3	QT4	QT4
		+noMT	+MT	+noMT	+MT	+noMT	+MT
FoodMarket	41.7	22.9	27.7	4.1	2.2	1.1	0.3
CatRobot	27.5	19.7	43.5	2.6	2.9	1.6	2.2
BasketballDrive	15.3	21.3	40.4	5.6	12.2	3	2.2
PartyScene	0.1	1.6	9.9	6.4	46.8	6.9	28.3
BQSquare	0.1	7.6	19.5	5.1	32.1	7.6	28
FourPeople	41.9	16.8	18.5	7.9	5.7	2.5	6.7
Average	21.1	14.98	26.58	5.28	16.98	3.78	11.28

From Table 1, we can observe that the probability that the MT partition is selected is high. In addition, the regions with relatively complex texture are more likely to be partitioned by MT to achieve better prediction results. Therefore, this algorithm makes a fast decision on MT partition to achieve the purpose of saving coding time. Considering that small CUs do not occupy much encoding time, just the CUs of size 16×16 and above are taken into account.

1) Extraction of gradient information of CU

Gradient is often used in edge detection for image processing, which can be applied to CU partition decision [16]. Commonly used gradient operators include Roberts operator, Prewitt operator, and Sobel operator. Roberts operator is the simplest kind of operator, but it will cause the phenomenon of local edge loss, causing the contour edge of the detected object to be discontinuous. Prewitt operator does not have high accuracy

for edge positioning, the detected edge is wider, and there are slightly more discontinuities. Sobel operator that performs weighted averaging on the upper, lower, left, and right neighboring pixels of a pixel can provide accurate edge direction information is adopted in this algorithm. Therefore, Sobel operator is adopted in this section to extract the gradient information of CU. The two convolution masks of Sobel operator, \mathbf{G}_x and \mathbf{G}_y , are described as follows.

$$\mathbf{G}_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad (3)$$

$$\mathbf{G}_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \quad (4)$$

We add the absolute gradients of the CU in the horizontal and vertical directions to get the absolute total gradient of the CU (G_{total}) as Eq. (5), (6) and (7) are the Calculation formula of the absolute total gradient in horizontal direction (G_{x_total}) and vertical direction (G_{y_total}), respectively.

$$G_{total} = G_{x_total} + G_{y_total} \quad (5)$$

$$G_{x_total} = \sum_{i=0}^{width-1} \sum_{j=0}^{height-1} |G_x(i,j)| \quad (6)$$

$$G_{y_total} = \sum_{i=0}^{width-1} \sum_{j=0}^{height-1} |G_y(i,j)| \quad (7)$$

Where, $G_x(i, j)$ and $G_y(i, j)$ represent the gradients of each pixel in horizontal and vertical direction respectively. $width$ and $height$ represent the numbers of pixels in each row and each column of CU, respectively. $G_x(i, j)$ and $G_y(i, j)$ are calculated as follows.

$$G_x(i,j) = \mathbf{M}_{i,j} \times \mathbf{G}_x \quad (8)$$

$$G_y(i,j) = \mathbf{M}_{i,j} \times \mathbf{G}_y \quad (9)$$

Where, $\mathbf{M}_{i,j}$ represents a 3×3 matrix centered on the pixel with coordinates (i, j) .

2) Fast CU partition decision

In this work, all CUs are classified into three types (horizontal texture CU, vertical texture CU and other CU) according to the gradient information. The partition types of CU are determined as follows.

If the following two conditions are met: ① $G_{total} > TH$; ② $G_{x_total} > G_{y_total}$, then classify the CU to horizontal texture CU and skip SPLIT_BT_VER and SPLIT_TT_VER. Otherwise, if the following two conditions are met: ① $G_{total} > TH$; ② $G_{y_total} > G_{x_total}$, then classify the CU to vertical texture CU and skip SPLIT_B-T_HOR and SPLIT_TT_HOR. Otherwise, classify the CU to other CU and traverse all partition types.

3) Derivation of threshold

The CUs with larger size or richer texture represented by larger G_{total} are highly to be further partitioned. The threshold, namely TH , is used to select this type of CU before deciding the partition types of CU. TH is defined as follows.

$$TH = \alpha > > (10 - bitdepth) \tag{10}$$

Where, $bitdepth$ represents the coding bit depth and α is adjustable parameter. α is derived from three representative test sequences include *FoodMarket*, *BasketballDrive* and *PartyScene*. Figure 5 shows the relationship between α , BDBR and coding time saving (TS). Where, BDBR represents the percentage of the code rate that can be saved by a certain encoding method under the same objective quality. The calculation formula of TS is as follows,

$$TS = \frac{(T_{ref} - T_{pro})}{T_{ref}} \times 100\% \tag{11}$$

Where, T_{ref} and T_{pro} represent the encoding time of VTM7.0 reference software and proposed algorithm, respectively. We can observe from Fig. 5 that the BDBR and TS decrease with α and there exists a corner point for curve of BDBR from which the curve starts to be steady. Therefore, this point can be selected as the value of α . α is set to 700 in this work.

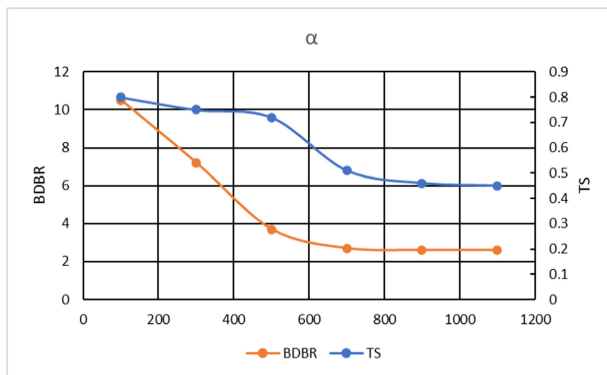


Fig. 5. Relationship between α , BDBR and coding time saving

3.2 Fast Intra Prediction Mode Decision Algorithm

In VVC intra prediction, the most time-consuming part is the calculation of *RDCost*. Although, VVC introduces three-stage fast intra mode decision and two-step fast RMD process, the coding complexity of VVC intra prediction is still very high. A fast intra prediction mode decision algorithm according to gradient information is proposed.

In order to analyse the correlation between the best intra prediction mode and gradient information of CU, six sequences mentioned above are tested (Table 2).

Table 2. Best intra mode distribution for different sequences (%).

Sequence	Sit1	Sit2	Sit3	Sit4	Sit5
FoodMarket	27.55	26.34	3.57	5.25	37.29
CatRobot	28.32	27.15	2.06	1.94	40.53
BasketballDrive	21.54	20.71	3.84	4.61	49.30
PartyScene	26.94	24.60	5.08	5.46	37.92
BQSquare	22.58	26.05	3.95	4.26	43.16
FourPeople	33.57	33.47	4.09	2.46	26.41
Average	26.75	26.39	3.77	3.99	39.10

Sit1 (situation1) means that the CU is classified to horizontal texture CU and the distribution of the best intra prediction mode is between 2 to 33; Sit2 (situation2) means that the CU is classified to vertical texture CU and the distribution of the best intra prediction mode is between 34 to 65; Sit3 (situation3) means that the CU is classified to horizontal texture CU and the distribution of the best intra prediction mode is between 34 to 65; Sit4 (situation4) means that the CU is classified to vertical texture CU and the distribution of the best intra prediction mode is between 2 to 33; Sit5 (situation5) means the best prediction mode is Planar mode or DC mode.

From Table 1, we can find that DC mode, Planar mode and the mode conformed to the major edge of image would probably be the best prediction mode. However, some unexpected situations should not be ignored.

Therefore, a method of reducing the intra prediction modes need to be traversed in RMD according to the gradient information of the CU is proposed. At the first stage of RMD, skip the mode 36, 40, 44, 48, 52, 56, 60, 64, if the CU is horizontal texture CU. Otherwise, skip the mode 4, 8, 12, 16, 20, 24, 28, 32, if the CU is vertical texture CU (Fig. 6).

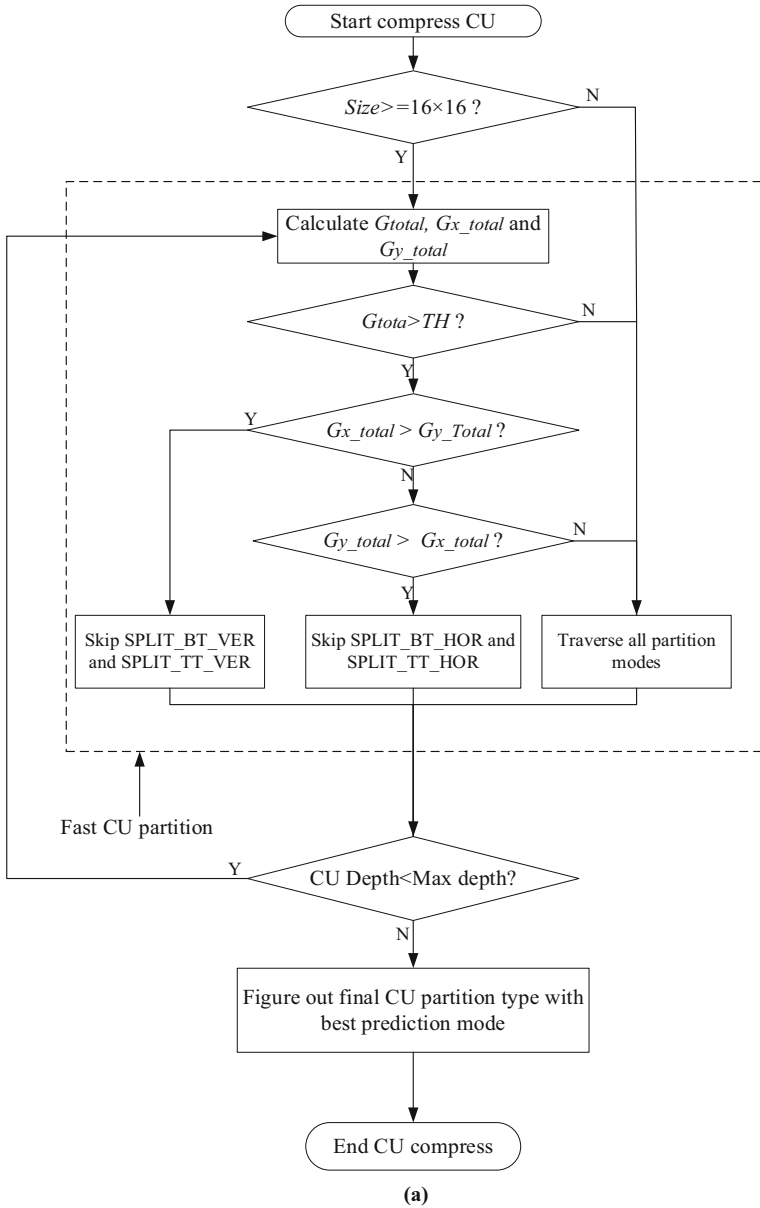


Fig. 6. Flow chart of fast CU partition and mode decision (a) Flow chart of fast CU partition (b) Flow chart of fast mode decision

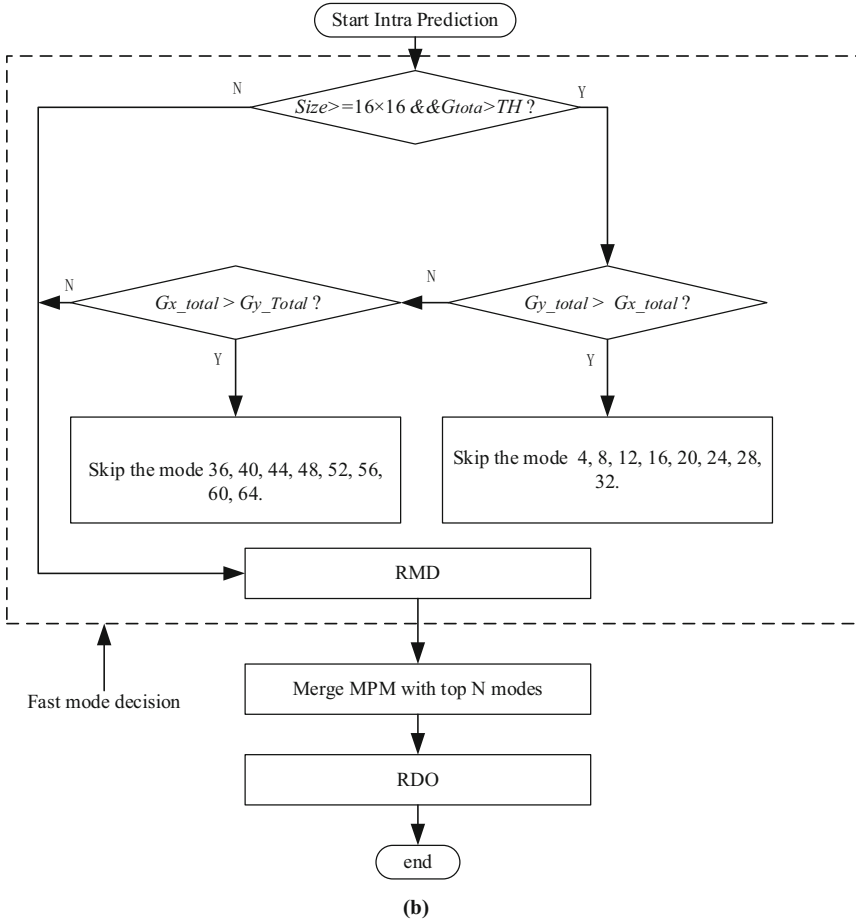


Fig. 6. (continued)

4 Overall Algorithm

According to the previous analysis, our fast intra prediction algorithm is described as follows.

- 1) Start compress CU.
- 2) Extract gradient information, including absolute total gradient and that in horizontal direction and vertical direction of CU.
- 3) Fast CU partition decision: if $G_{total} > TH$ and $G_{x_total} > G_{y_total}$, then classify the CU to horizontal texture CU and skip SPLIT_BT_VER and SPLIT_TT_VER. Otherwise, if $G_{total} > TH$ and $G_{y_total} > G_{x_total}$, then classify the CU to vertical texture CU and skip SPLIT_BT_HOR and SPLIT_TT_HOR. Otherwise, classify the CU to other CU and traverse all partition types.

- 4) Fast mode decision: skip some modes at the first stage of RMD, according to the gradient information of CU. If the CU is horizontal texture CU, then skip the mode 36, 40, 44, 48, 52, 56, 60, 64. Otherwise, if the CU is vertical texture CU, then skip the mode 4, 8, 12, 16, 20, 24, 28, 32.
- 5) The CU partition types and best intra prediction mode are decided.

5 Experimental Result

This fast intra prediction algorithm was implemented in VTM7.0 reference software. On the condition of AI, 30 frames of each sequence of all classes are tested with QP value of 22, 27, 32 and 37, afterwards calculate the mean result of them as the final result of each sequence. The test platform is Intel(R) Xeon(R) Gold 6128 CPU @ 3.40 GHz, 64 GB RAM and CentOS7 system.

Coding efficiency is measured with BDBR, BD-PSNR and *TS*. Where, BD-PSNR represents the difference between the PSNR-*Y* of the two methods at the given equivalent code rate.

Table 3. Simulation results of the proposed algorithm.

Sequence	BDBR (%)	BD-PSNR (dB)	<i>TS</i> (%)
Tango	1.52	-0.02	45.52
FoodMarket	1.26	-0.02	33.07
Campfire	2.65	-0.08	49.70
CatRobot	1.93	-0.03	34.30
DaylightRoad	1.34	-0.03	40.20
ParkRunning	0.62	-0.04	45.24
MarketPlace	1.78	-0.05	53.16
RitualDance	3.86	-0.19	53.85
Cactus	2.61	-0.09	51.70
BasketballDrive	2.76	-0.06	51.39
BQTerrace	1.90	-0.11	47.94
BasketballDrill	4.14	-0.13	50.51
BQMall	2.47	-0.15	50.47
PartyScene	1.47	-0.12	42.06
RaceHorses	1.67	-0.11	50.17
BasketballPass	3.28	-0.09	50.63
BQSquare	2.43	-0.21	43.62
BlowingBubbles	1.67	-0.10	56.20
RaceHorses	2.04	-0.15	43.27
FourPeople	2.91	-0.16	53.68
Johnny	2.85	-0.11	51.58
KristenAndSara	2.45	-0.12	52.81
Average	2.25	-0.09	47.78

Table 3 presents the results of the proposed algorithm compared with VTM7.0. From this table, we conclude that the proposed algorithm, compared with VTM7.0, can save 47.78% of the encoding time with 2.25% BDBR increase and 0.09 dB BD-PSNR decrease.



(a)



(b)

Fig. 7. CU partition results of different algorithms (a) partition result of VTM7.0 (b) partition result of the proposed algorithm

Figure 7 displays the partition of the first frame from BasketballDrive by using the default algorithm in VTM7.0 and the proposed algorithm separately. For some areas with flat texture, the partition results of the two algorithms are the same. But for areas with more complex textures, the partition results of the two algorithms are different. Although the proposed algorithm skips some partitions in areas with complex textures,



(a)



(b)

Fig. 8. Decoded frames of different algorithms (a) A decoded frame of DaylightRoad by VTM7.0 (b) A decoded frame of DaylightRoad by the proposed algorithm

resulting in a difference from the partition result of the default algorithm in VTM7.0, the performance degradation caused by this difference is almost negligible.

Figure 8 shows the decoded frames from DaylightRoad by using different algorithms. The naked eyes can hardly identify the difference between the two frames in Fig. 8. There is almost no difference between these buildings and vehicles.

6 Conclusions

This paper proposed a fast intra prediction algorithm for VCC includes fast CU partition and fast mode decision by making use of the gradient information of CU to skip some partition modes and intra prediction modes. Experimental results show that the proposed algorithm achieves a significant coding time saving with negligible effect in BDBR and BD-PSNR, compared to VTM7.0. The algorithm proposed in this paper can be applied to scenarios that require high real-time video transmission, such as live broadcast and video communication.

Acknowledgment. This work is supported by the National Natural Science Foundation of China (61571102) and Chongqing Basic and Frontier Research Project (cstc2017jcyjXBX0037).

References

1. Sullivan, G.J., Ohm, J., Han, W., Wiegand, T.: overview of the high efficiency video coding (HEVC) standard. *IEEE Trans. Circ. Syst. Video Technol.* **22**(12), 1649–1668 (2012)
2. Bross, B.: Versatile video coding (Draft 1). Joint Video Experts Team (JVET) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11 10th Meeting. San Diego, US: Joint Video Exploration Team, Doc. JVET-J1001 (2018)
3. Chen, J., Alshina, E.: Algorithm description for versatile video coding and test model 1 (VTM 1). Joint Video Experts Team (JVET) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11 10th Meeting. San Diego, US: Joint Video Exploration Team, Doc. JVET-J1002 (2018)
4. Ma, J., et al.: Quadtree plus binary tree with shifting. Joint Video Experts Team (JVET) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11 10th Meeting. San Diego, US: Joint Video Exploration Team, Doc. JVET-J0035 (2018)
5. Auwera, G.V., Seregin, V., Ramasubramonian, A.K., Karczewicz, M.: CE3: variable number of directional intra modes (Tests 1.1.1 and 1.1.2). Joint Video Experts Team (JVET) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11 11th Meeting. Ljubljana, SI: Joint Video Exploration Team, Doc. JVET-K0060 (2018)
6. Racapé, F., Rath, G., Urban, F., Zhao, L., Liu, S., Zhao, X., et al.: CE3-related: Wide-angle intra prediction for non-square blocks. Joint Video Experts Team (JVET) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11 11th Meeting. Ljubljana, SI: Joint Video Exploration Team, Doc. JVET-K0500 (2018)
7. Auwera, G.V., Seregin, V., Said, A., Ramasubramonian, A.K., Karczewicz, M.: CE3: simplified PDPC (Test 2.4.1). Joint Video Experts Team (JVET) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11 11th Meeting. Ljubljana, SI: Joint Video Exploration Team, Doc. JVET-K0063 (2018)

8. Bross, B., et al.: CE3: multiple reference line intra prediction (Test 1.1.1, 1.1.2, 1.1.3 and 1.1.4). Joint Video Experts Team (JVET) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11 12th Meeting. Macao, CN: Joint Video Exploration Team, Doc. JVET-L0283 (2018)
9. Zhao, X., Chen, J., Karczewicz, M., Zhang, L., Li, X., Chien, W.: Enhanced multiple transform for video coding. In: Data Compression Conference (DCC). Snowbird, United States, pp. 73–82. IEEE (2016)
10. Li, X., Suehring, K.: AHG report: JEM software development (AHG3). Joint Video Experts Team (JVET) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11 10th Meeting. San Diego, US: Joint Video Exploration Team, Doc. JVET-J1001 (2018)
11. Fu, T., Zhang, H., Mu, F., Chen, H.: Fast CU partitioning algorithm for H.266/VVC intra-frame coding. In: IEEE International Conference on Multimedia and Expo (ICME). Shanghai, China, pp. 55–60. IEEE (2019)
12. Yang, H., Shen, L., Dong, X., Ding, Q., An, P., Jiang, G.: Low complexity CTU partition structure decision and fast intra mode decision for versatile video coding. *IEEE Trans. Circ. Syst. Video Technol.* p. 1 (2019)
13. Lin, T., Jiang, H., Huang, J., Chang, P.: Fast binary tree partition decision in H.266/FVC intra coding. In: IEEE International Conference on Consumer Electronics-Taiwan (ICCE-TW). Taichung, China, pp. 1–2. IEEE (2018)
14. Chen, J., Chiu, Y., Lee, C., Tsai, Y.: Utilize neighboring LCU depth information to speedup FVC/H.266 intra coding. In: International Conference on System Science and Engineering (ICSSE). Dong Hoi, Vietnam, pp. 308–312. IEEE (2019)
15. Mir, J., Talagala, D.S., Fernando, A.: Optimization of HEVC λ -domain rate control algorithm for HDR video. In: IEEE International Conference on Consumer Electronics (ICCE). Las Vegas, Vietnam, pp. 1–4. IEEE (2018)
16. Jiang, W., Ma, H., Chen, Y.: Gradient based fast mode decision algorithm for intra prediction in HEVC. In: 2nd International Conference on Consumer Electronics, Communications and Networks (CECNet). Yichang, China, pp. 1836–1840. IEEE (2012)