ORIGINAL RESEARCH PAPER

An efficient low-complexity block partition scheme for VVC intra coding

Yun Song[1,2] · Biao Zeng[1,2] · Miaohui Wang[3,4] · Zelin Deng[1,2]

Abstract
The newest versatile video coding (VVC) adopts a novel quadtree with a nested multi-type tree (QTMT) partition structure for intra-frame coding and splits the coding unit (CU) into not only square sub-blocks but also rectangular sub-blocks. The more flexible CU sizes improve the encoding efficiency while cause much heavier computational complexity than High-Efficiency Video Coding (HEVC). In this paper, an efficient low-complexity CU partition method based on the texture characteristic is proposed to achieve a good trade-off between the coding efficiency and the complexity reduction. Specifically, the texture complexities of vertical and horizontal directions are quantitatively measured in terms of the sum of the mean absolute deviation (SMAD) of the sub-blocks. Then, the vertical and horizontal texture complexities are compared to eliminate some unlikely partition modes. Moreover, the threshold of directional SMAD ratios for choosing the texture direction is adjusted according to the quantization parameter (QP) to improve the prediction accuracy. Experimental results show that the proposed method achieves the average complexity reduction efficiency by 54.60% than the VVC test model (VTM). The implementation of the proposed method is publicly available at http://sites.google.com/site/wangmiaohui/ and http://github.com/csust-sonie/fastVVC_RTIP_2107.

Keywords Versatile video coding · QTMT · Complexity reduction · Video coding

1 Introduction

In recent years, the rapidly growing multimedia technologies [31] like 4K ultra-high-definition (UHD) video, 360° immersive multimedia and high dynamic range (HDR) video

✉ Miaohui Wang
 wang.miaohui@gmail.com

 Yun Song
 sonie@126.com

[1] School of Computer and Communication Engineering, Changsha University of Science and Technology, Changsha 410114, People's Republic of China

[2] Hunan Provincial Key Laboratory of Intelligent Processing of Big Data on Transportation, Changsha University of Science and Technology, Changsha 410114, People's Republic of China

[3] School of Computer Science and Technology, Hainan University, Haikou 570228, People's Republic of China

[4] Guangdong Key Laboratory of Intelligent Information Processing, Shenzhen University, Shenzhen 518060, People's Republic of China

increase the visual quality, but lead to tremendous workload for the storage and transmission. The previous High-Efficiency Video Coding (HEVC) is difficult to satisfy the compression demand of explosive video data. To overcome the bottleneck of digital media development, the Joint Video Exploration Team (JVET), consisting of ISO Moving Picture Expert Group (MPEG) and ITU-T Video Coding Expert Group (VCEG), started to explore the next-generation video standard beyond HEVC, named Future Video Coding (FVC) in 2015. One of the new techniques of FVC is the quadtree with nested binary tree (QTBT) structure, and it is adopted in the joint exploration model (JEM) [17]. The coding unit (CU) is first partitioned into four equal-sized blocks by the quadtree (QT) structure. Then, the binary tree (BT) partition is performed sequentially to obtain the rectangular CUs. Based on such a partition tool in FVC [7], JEVT further introduces a novel partition technology, named quadtree with nested multi-type tree (QTMT), and develops a new standard for video coding, called Versatile Video Coding (VVC). VVC aims to achieve more than 50% coding efficiency improvement while maintaining the same video quality as the HEVC standard. Except for the QT partition

structure, there are four multi-type tree (MT) partition structures, including vertical binary tree partition (BV), horizontal binary tree partition (BH), vertical ternary tree partition (TV), and horizontal ternary tree partition (TH). As an illustration in Fig. 1, the visual QTMT result of VVC shows a very complicated partition pattern, which indicates that it needs a significant amount of the running time to achieve the optimal CU partition.

Recursive MT partition brings more flexibility to the CU shape, but the related rate-distortion optimization (RDO) process increases the computational complexity. If we can predict the MT partition mode in advance, the encoding time will be significantly saved. It can be seen that in Fig. 1b and c, the dominant direction of the best MT partition mode is closely related to the texture direction. The CUs with vertical textures are more likely to be partitioned by the vertical MT modes, while the CUs with horizontal textures are more likely to be partitioned by the horizontal MT partition modes. Inspired by the observation, we consider determining the partition direction by making use of the texture features. We introduce the concept of the sum of the mean absolute deviation (SMAD) of the sub-blocks to evaluate the complexity of vertical and horizontal directions. Moreover, the ratios of vertical-to-horizontal SMAD are exploited to
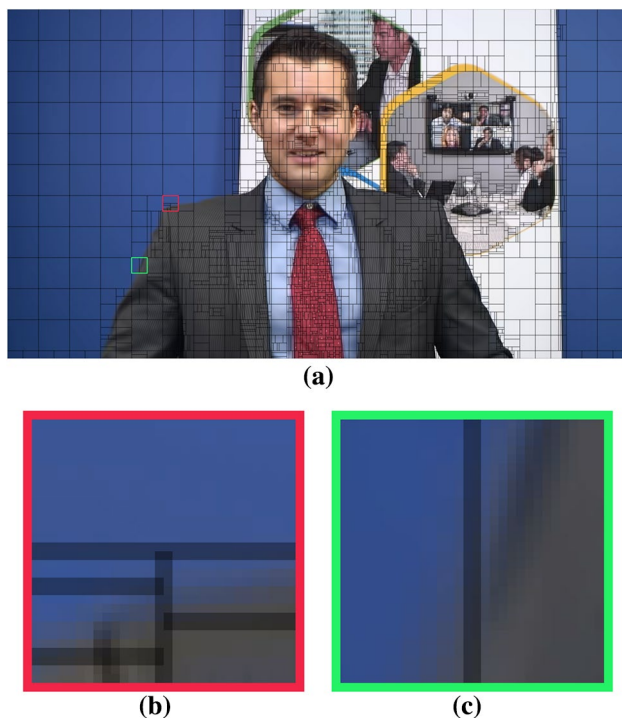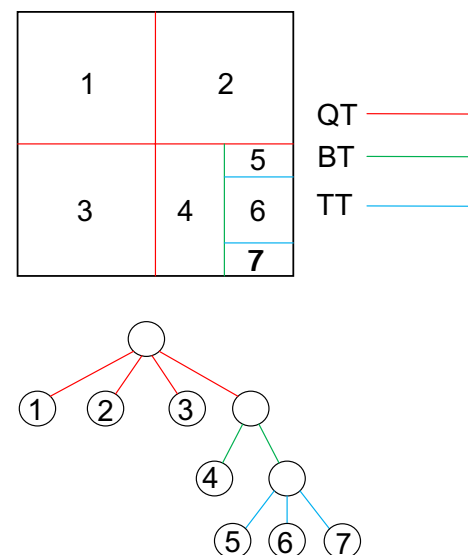
decide the texture direction. Based on the observation and statistical results, we propose an efficient low-complexity CU partition method to reduce the encoding complexity of VVC. We remove the unlikely MT partition patterns by comparing the ratio of vertical SMAD to horizontal SMAD under different quantization parameters (QPs). The experimental results demonstrate that the proposed algorithm reduces the encoding time of the intra-prediction significantly with a negligible coding loss.

The rest of this paper is organized as follows: Sect. 2 briefly introduces some basic background knowledge of the VVC intra coding, and reviews some related fast algorithms. In Sect. 3, the motivation and statistical analysis are presented, and then, the proposed fast CU decision algorithm is described in details. Sect. 4 presents the experimental results and comparisons with other recent methods. The conclusion is provided in Sect. 5.

2 Background and related work

2.1 Background of the VVC intra coding

Block partition has taken a crucial role in video coding. VVC integrates a bunch of block partitions for intra-prediction. It splits the frame in the luminance and chrominance components [10], and unifies the concepts of the coding unit (CU), prediction unit (PU), and transform unit (TU). VVC employs the QTMT partition structure to determine the final CU size. Figure 2 demonstrates a schematic example of QTMT structure and the corresponding recursive partition trees. Unlike the QT structure that splits a CU into four

Fig. 1 The visual partition results of the QTMT structure in Versatile Video Coding (VVC). **a** The partition results of the first frame of "Johnny" (1280 × 720) under quantization parameter (QP) = 32. **b** The example of the horizontal partition. **c** The example of the vertical partition

Fig. 2 An exemplary QTMT structure and the corresponding recursive partition tree in VVC

squared sub-CUs, the BT structure divides a CU into two rectangular blocks with a symmetric area, while the ternary tree (TT) structure splits a CU into three rectangular blocks with an area ratio of 1:2:1. The determination process of the VVC intra-prediction and partition process is presented in Fig. 3. The coding tree unit (CTU) is partitioned into four equal-sized sub-CUs by QT first, and the resulted sub-CUs are recursively partitioned into equal-sized CUs by QT or rectangle-sized CUs by MT. The QTMT structure makes the CU more adaptive to the video content, and improves the coding efficiency substantially. However, it also inevitably increases the computational complexity to traverse all the possible partition combinations in a CTU one by one.

To avoid the overlapping partitions, there are some predefined restrictions for the QTMT structure. For instance, the 128×128 CTU is divided into four 64×64 quadtree nodes by default, because the maximum CU size is 64×64, and the minimum CU size is 4×4. The minimum QT partition size is 16×16, while the maximum MT partition size is 32×32. If a CU is split by MT, QT partition will be forbidden for obtaining leaf nodes. If the current CU performs the the TT partition, the inter-sub-blocks are not allowed to be BT partitioned. The redundant partitioning examples are shown in Fig. 4.

As for the intra-prediction in VVC, the candidate modes are extended from 35 to 67, including a DC mode, a planar mode, and 65 directional modes, as shown in Fig. 5. The DC
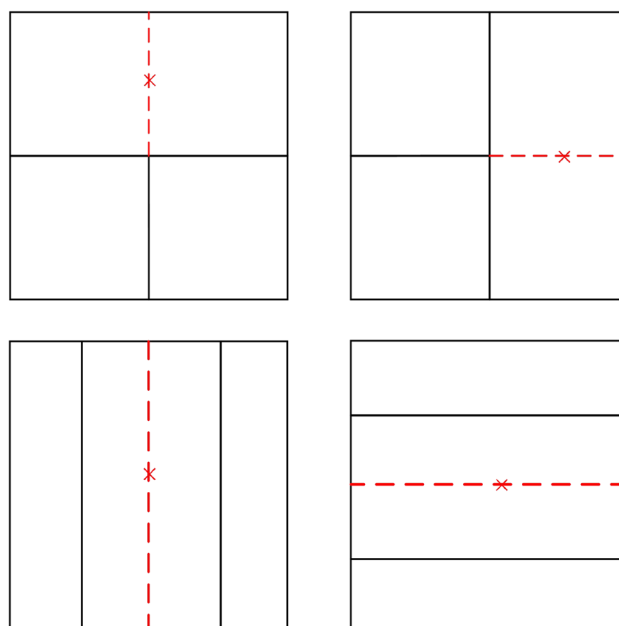

Fig. 4 Illustration of the redundant partitions in VVC

mode is suitable for large flat areas, and the planar mode is for regions where the pixel values are slowly changed. The black lines denote the prediction modes in HEVC, while the red lines denote the additional modes in VVC. The intra-frame coding mode is determined as follows. First, the rough mode decision (RMD) is performed on the traditional 35 modes from HEVC. Then, the first N candidates based on
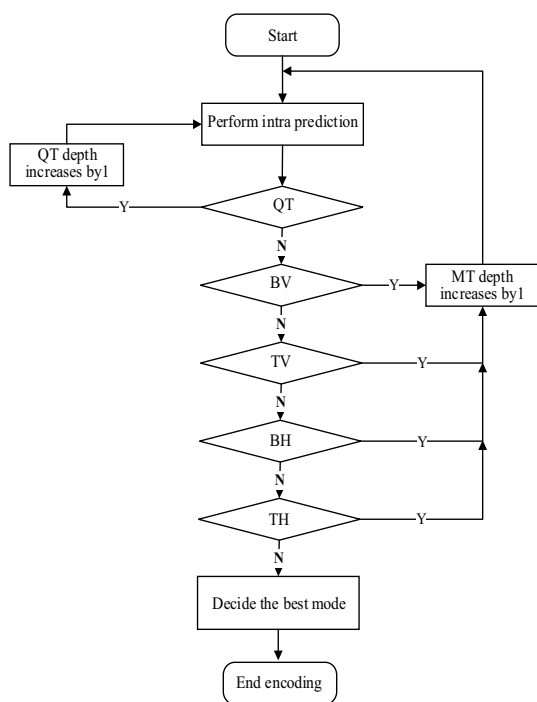

Fig. 3 Pipeline of the VVC intra-prediction and partition determination process
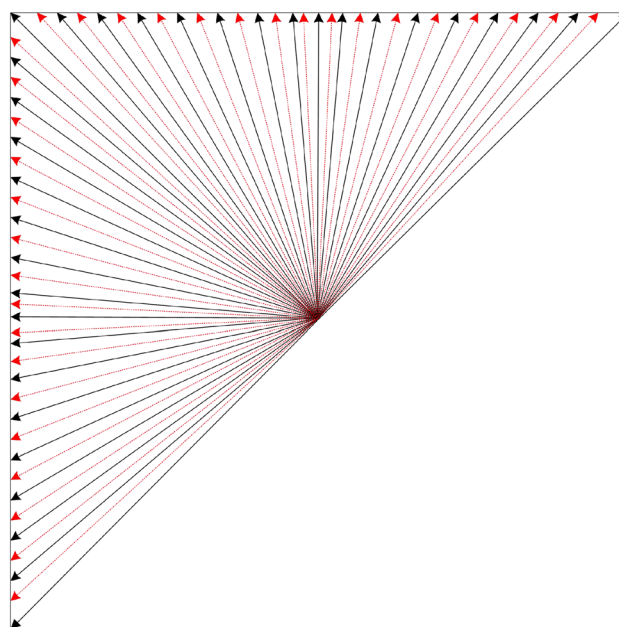

Fig. 5 Illustration of directional intra-prediction modes in VVC

RMD with the smallest sum of absolute Hadamard transformed difference (SATD) are further considered in a fine mode decision process. The direct neighboring modes of the first N modes are also added to the candidates, and the list is updated dynamically in terms of the SATD value. Finally, the most probable modes from the left and above neighboring blocks are merged to perform the full RDO process.

Moreover, VVC adopts many new encoding tools for intra-prediction, such as the position-dependent intra-prediction combination (PDPC) [23], cross-component linear model (CCLM) [18], intra-sub-partition (ISP) [12], and wide angular intra-prediction (WAIP) [37]. These new technologies have been integrated into the VVC test model (VTM). It is worth noting that they benefit the compression efficiency but considerably increase the computational complexity.

2.2 Related work

Many researchers have been devoted to developing fast algorithms for video encoding. For example, the global and local edge complexities of the current CU and its sub-CUs were exploited in [22] to classify a CU to be split, non-split, or undetermined at each depth level. [25] and [24] proposed fast algorithms for HEVC intra coding tasks by employing the texture property and the depth information to make an early CU pruning decision or CU splitting decision. In [33], the content similarity between the current CU and its co-located CU in temporally, and the neighboring CUs in the same frame were referenced to determine the depth early. In [38], the partition modes of different-size CUs were reduced according to the correlation between the quadtree partition and the texture partition features. Wang et al. [30] proposed a fast prediction mode and CU size decision algorithm by grouping 35 intra-prediction modes into 5 sub-lists according to the texture complexity. Hosseini et al. [16] designed a rate-complexity model to allocate the complexity budget and obtain the target complexity by considering the complexity distribution and video content. Varma et al. [29] adopted a fast test zonal search to reduce the encoding complexity, which was used in gird, raster, and refinement search stages. Meanwhile, machine learning solutions were also developed for the fast encoding algorithm. Lim et al. [21] designed a fast CU size decision based on the Bayes decision. The PU skip and split termination were performed by utilizing the ratio of the rate-distortion (RD) cost in RMD from the current PU to that of the neighboring PU. In [35], the CU size decision was modeled as a classification, and the representative features were utilized in support vector machine (SVM) to perform the early CU decision. Grellert et al. [14] trained an automatic learning model by combining the statistically based heuristics algorithm and fast decision method. Bouaafia et al. [6] proposed two fast CU partition algorithms for HEVC inter-mode decision, including the SVM trained

model and the convolutional neural network (CNN) model. Tahir et al. [26] devised a fast algorithm based on a combination of the online and offline random forests (RF) to deal with the CU and transform unit partition. Apart from the above fast methods for HEVC, some fast methods were also investigated for 3D-HEVC, such as correlation-based [20], edge direction-guided [15, 34], and decision tree-driven [3].

The fast encoding schemes mentioned above achieved well-complexity reduction on the encoder side. However, they were all designed for HEVC instead of VVC. Recently, some fast CU algorithms were proposed for VVC. Inspired by the RMD process, [19] adopted the SATD-based mode decision to determine the best intra-prediction mode for sub-block partitions. In [13], the correlations between the parent CU and its following horizontally splitting sub-CUs were exploited to skip the vertical split through the Bayesian rule. And then, the RD cost of the vertical binary split was applied to avoid the horizontal ternary split early. [9] devised a fast algorithm for the new QTMT structure, where directional gradient and the variance of variance of sub-CUs features were exploited for the CU partition decision. An innovative partition determination framework was proposed in [32], in which the QT partition decision was judged before the MT partition decision. The model transformed the multi-classification problem into multiple binary-classification problems at each decision level, which was handled by decision tree classifiers. This work also proposed a fast intra-mode decision with gradient descent search. [28] skipped the vertical or horizontal partition modes, and performed the early termination for intra coding according to the edge features extracted by the Canny operator. Cui et al. [11] calculated the gradient of each CU along vertical, horizontal, and two diagonal directions, and then pre-determined the likelihood of BT or TT partition in horizontal or vertical direction. [2] proposed a lightweight and tunable partitioning method for QTBT and QTMT using the RF classifier to decide the likely partition modes. [8] established the relationships between the extracted features and splitting modes from the online learning frames, and then trained the SVM classifier to predict the direction of the MT structure. [27] performed a pre-decision algorithm for the homogeneous block by the Sobel gradient operator to skip the calculation of sub-CUs, and then implemented a pooling-variable CNN, which trained the partition decision for CUs with various sizes by only one network. [36] developed an adaptive CU partition algorithm based on deep learning and multi-feature fusion to terminate the iteration of the RDO process.

Being different from the previous methods, the article introduced a new fast block partition decision algorithm based on the complexity of the whole CU. Specifically, the vertical and horizontal complexity deviation of sub-blocks was characterized by SMAD. Then, the vertical and horizontal partition is predicted based on the texture direction,

where the unnecessary MT partition patterns are skipped accordingly. Experimental results show that the proposed method obtains a satisfying balance between the prediction accuracy and the complexity reduction.

3 Methodology

VVC adopts a flexible QTMT structure, which improves the coding efficiency at the cost of substantial complexity (i.e., ten times or more than HEVC). To reduce the complexity, we devise a fast CU size decision algorithm based on the texture direction to avoid the redundant MT partitions. In this section, we describe the proposed scheme in details as follows.

3.1 Motivation and statistical analysis

In the VVC reference software, MT partition structures take almost one-third of the total division, and become the most complicated operation, which accounts for more than 90% of the total encoding time on average [8]. The statistical proportion of the MT partition modes (from 32×32 to 8×8) for five sequences is shown in Fig. 6a, and the related running time proportion for the MT modes is shown in Fig. 6b. In the MT partitioning decision process, there is a great flexibility in the partition modes decision, but only one of the vertical and horizontal directions is adopted at the last decision. Therefore, if we can remove the unlikely MT partition modes in advance, the coding complexity from the expensive RDO process will be reduced significantly.

Moreover, it can be observed from Fig. 1a that for the region with smooth textures, like the blue background, VVC tends to choose larger size CUs. While for the area with the complicated textures, like the outline of the characters, VVC is more likely to select smaller size CUs. Moreover, when the region with the directional texture is divided into several sub-CUs along the same direction, which can be categorized into the vertical and horizontal directions roughly. The vertical and horizontal partitions are indicated as the red and green boxes in Fig. 1b and c, respectively.

A toy example is shown in Fig. 7, in which `block1` and `block2` have the same pixel values while `block3` and `block4` have the same pixel values. It can be apparently found that the complexity difference between sub-CUs in the vertical direction is larger than that in the horizontal direction, and the texture direction is considered as horizontal direction. Therefore, it is more preferred to skip the vertical partition mode, and choose the horizontal partition modes in advance.

(a)

(b)
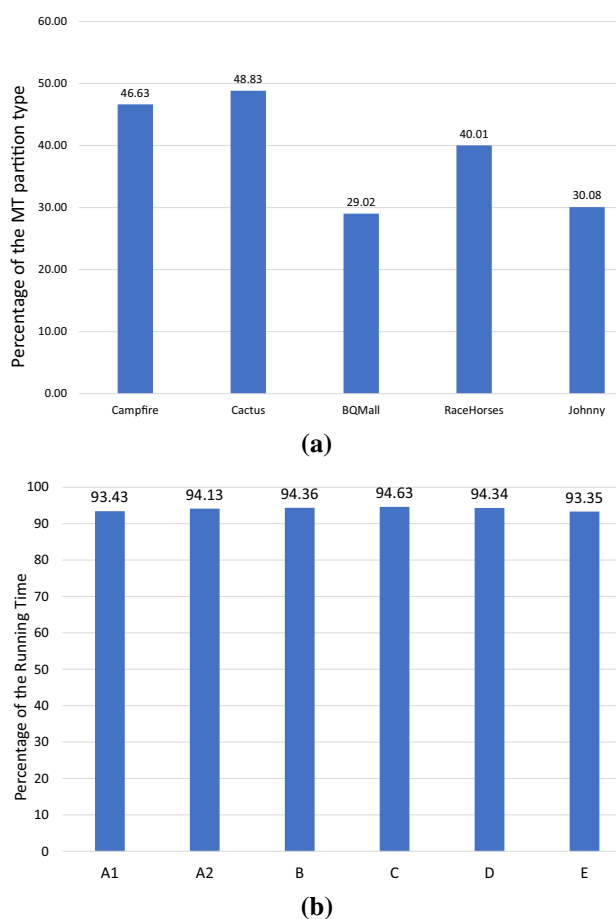
Fig. 6 Statistical data of the MT partition results in the VVC reference software. **a** The proportions of MT partition modes for five video sequences. **b** The average proportions of the encoding time by the MT partition modes for Class A1–E

Fig. 7 An example of the equal-sized sub-CUs
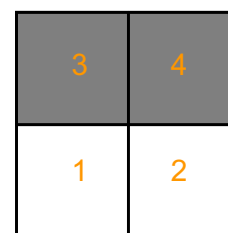
3.2 Establishment of the texture complexity measurement

The median absolute deviation (MAD) represents the mean of the data absolute deviations around its mean, which is computationally simpler than the standard deviation. In this section, we adopt MAD to measure the complexity (or variability) of the image block. The complexity measurement, $\mathcal{C}_{\mathrm{mad}}$, is formulated as

$$C_{\text{mad}} = \frac{1}{W \times H} \sum_{x=1}^{W} \sum_{y=1}^{H} |p(x,y) - m(\mathbf{p})|, \qquad (1)$$

where W and H are the width and height of the current CU, and $p(x,y)$ represents the pixel value at position (x,y). $m(\mathbf{p})$ denotes the mean of the pixel values, which is defined by

$$m(\mathbf{p}) = \frac{1}{W \times H} \sum_{x=1}^{W} \sum_{y=1}^{H} p(x,y). \qquad (2)$$

The straightforward idea of the proposed fast block partition determination scheme is described as follows: if the C_{mad} value is less than a threshold $\mathcal{T}$, it means that there is a little variance between pixels in the CU; it is classified into the homogeneous type, and performs the early determination. Otherwise, the current CU is classified into the complex type. In such a case, the next bypass method is performed.

Based on the above analysis, we propose to divide the current CU into four equal-sized blocks, and compute the complexity of each sub-CU in terms of C_{mad}. Considering the new characteristic of the VVC codec, we first calculate the mean complexity deviation between sub-CUs in both the vertical and horizontal directions. Then, we compare the complexity difference from such two directions to estimate the texture direction of the whole CU.

The C_{mad}^{bl}, C_{mad}^{br}, C_{mad}^{tl}, and C_{mad}^{tr} values are used to represent the texture complication of the bottom-left, bottom-right, top-left, and top-right sub-CUs, respectively, which are calculated by Eqs. (3a)–(3d)

$$C_{\text{mad}}^{bl} = \frac{1}{\left(\frac{W}{2}\right) \times \left(\frac{H}{2}\right)} \sum_{x=1}^{W/2} \sum_{y=1}^{H/2} |p(x,y) - m(\mathbf{p}_1)|, \qquad (3a)$$

$$C_{\text{mad}}^{br} = \frac{1}{\left(\frac{W}{2}\right) \times \left(\frac{H}{2}\right)} \sum_{x=\frac{W}{2}+1}^{W} \sum_{y=1}^{H/2} |p(x,y) - m(\mathbf{p}_2)|, \qquad (3b)$$

$$C_{\text{mad}}^{tl} = \frac{1}{\left(\frac{W}{2}\right) \times \left(\frac{H}{2}\right)} \sum_{x=1}^{W/2} \sum_{y=\frac{H}{2}+1}^{H} |p(x,y) - m(\mathbf{p}_3)|, \qquad (3c)$$

$$C_{\text{mad}}^{tr} = \frac{1}{\left(\frac{W}{2}\right) \times \left(\frac{H}{2}\right)} \sum_{x=\frac{W}{2}+1}^{W} \sum_{y=\frac{H}{2}+1}^{H} |p(x,y) - m(\mathbf{p}_4)|. \qquad (3d)$$

We estimate the texture complexity according to the SMAD value of C_{mad} for each CU, including the vertical and horizontal complexities. The vertical SMAD, S_{ver}, is defined by the sum of the difference of the left sub-CUs and that of the right sub-CUs. Similarly, the horizontal SMAD, S_{hor}, is defined by the sum of the difference of the bottom sub-CUs and that of the top sub-CUs. Furthermore, we consider utilizing the ratio of S_{ver} and S_{hor} to represent the possibility of the partition mode, which is formulated as

$$\mathcal{R}_{\text{ver/hor}} = \frac{S_{\text{ver}}}{S_{\text{hor}}} = \frac{|C_{\text{mad}}^{bl} - C_{\text{mad}}^{tl}| + |C_{\text{mad}}^{br} - C_{\text{mad}}^{tr}|}{|C_{\text{mad}}^{bl} - C_{\text{mad}}^{br}| + |C_{\text{mad}}^{tl} - C_{\text{mad}}^{tr}|}. \qquad (4)$$

Experiments on the VVC reference software [1] were conducted to investigate the relationship between the final partition mode and the SMAD feature. Six video sequences from the JVET common test condition (CTC) [5] were used in the experiments, including "FoodMarket4" (3840×2160), "DaylightRoad2" (3840×2160), "Cactus" (1920×1080), "BQMall" (832×480), "BasketballPass" (416×240), and "FourPeople" (1280×720). The first five frames of each sequence were encoded in all-intra (AI) configurations under QP = 27. We collect the statistical results of $\mathcal{R}_{\text{ver/hor}}$ and the corresponding vertical and horizontal MT partition modes, and the overall results are presented in Table 1.

As we can see from Table 1, the percentages of the vertical and horizontal partition modes vary with different sequences. However, when the $\mathcal{R}_{\text{ver/hor}}$ values are less than 1, the percentages of vertical partition modes are 74%, 61%, 66%, 75%, 71%, and 78%, respectively. In such a case, the average possibility of being encoded by the vertical partition is more than twice that of the horizontal partition. Meanwhile, when the $\mathcal{R}_{\text{ver/hor}}$ values are greater than 1, we can find that the average possibility of being encoded by the horizontal partition is more than twice that of the vertical partition.

Table 1 The statistical data of the vertical and horizontal MT modes over 32×32 and 16×16 for $\mathcal{R}_{\text{ver/hor}} < 1$ or $\mathcal{R}_{\text{ver/hor}} > 1$

	Vertical partition modes (%)	Horizontal partition modes (%)
(a) FoodMarket4 (3840×2160)		
$\mathcal{R}_{\text{ver/hor}} < 1$	74	26
$\mathcal{R}_{\text{ver/hor}} > 1$	27	73
(b) DaylightRoad2 (3840×2160)		
$\mathcal{R}_{\text{ver/hor}} < 1$	61	39
$\mathcal{R}_{\text{ver/hor}} > 1$	27	73
(c) Cactus (1920×1080)		
$\mathcal{R}_{\text{ver/hor}} < 1$	66	34
$\mathcal{R}_{\text{ver/hor}} > 1$	30	70
(d) BQMall (832×480)		
$\mathcal{R}_{\text{ver/hor}} < 1$	75	25
$\mathcal{R}_{\text{ver/hor}} > 1$	36	64
(e) BasketballPass (416×240)		
$\mathcal{R}_{\text{ver/hor}} < 1$	71	29
$\mathcal{R}_{\text{ver/hor}} > 1$	24	76
(f) FourPeople (1280×720)		
$\mathcal{R}_{\text{ver/hor}} < 1$	78	22
$\mathcal{R}_{\text{ver/hor}} > 1$	39	61

Based on the statistical data in Table 1, the proposed fast CU decision scheme is designed based on the following rule. When the $\mathcal{R}_{\text{ver/hor}}$ value is greater than the higher threshold $\mathcal{T}_h$, it means that the complexity of the horizontal direction is greater than that of the vertical direction, and the current CU is more likely to have the horizontal texture. In this case, we abandon the vertical MT partition modes in advance. In contrast, when the $\mathcal{R}_{\text{ver/hor}}$ value is less than a lower threshold $\mathcal{T}_l$, it represents that the current CU is more likely to have the vertical texture, so we skip the horizontal partitioning mode. When $\mathcal{R}_{\text{ver/hor}}$ is between $\mathcal{T}_h$ and $\mathcal{T}_l$, it can be considered that there is a little difference in the texture complexity between the vertical and horizontal directions. In such a case, every CU partition almost has the same probability. Thus, choosing the original VVC scheme can ensure the prediction accuracy. Although we separate the CU into the equal-size blocks, the proposed method is also applied for the TT partition. Considering that the maximum MT size is 32×32, we only implement our method for the luminance blocks with 32×32 and 16×16 to keep the trade-off between the complexity reduction and the encoding efficiency.

3.3 Determination of the partition condition

To verify the assumptions, we statistically count the cumulative distribution of the splitting modes and the $\mathcal{R}_{\text{ver/hor}}$ values. The mistaken cases are defined as follows: (1) when $\mathcal{R}_{\text{ver/hor}} > \mathcal{T}_h$, the best splitting mode should be the horizontal splitting mode theoretically, but the actual splitting mode is vertical; (2) when $\mathcal{R}_{\text{ver/hor}} < \mathcal{T}_l$, the best splitting mode should be the vertical splitting mode theoretically, but the best splitting mode is horizontal instead. The error ratio is formulated as Eq. (5)

$$\mathcal{R}_{\text{err}} = \frac{N_{err}}{N_{all}} \times 100\%, \tag{5}$$

where N_{err} represents the number of the misclassification of the vertical and horizontal partitions, and N_{all} represents the total number of all the testing CUs. Accordingly, the correct ratio, $\mathcal{R}_{\text{acc}}$, is calculated by $\mathcal{R}_{\text{acc}} = 1 - \mathcal{R}_{\text{err}}$. Consequently, we can use these two indicators to estimate the similarity between our method and the original scheme.

The threshold plays a crucial role in the proposed scheme. If the $\mathcal{T}_h$ value is too small, the algorithm will obtain better complexity reduction, because more partition modes are determined in advance. If the $\mathcal{T}_h$ value is too large, the coding performance will be better, but the computational complexity will increase. The encoding results of 32×32 and 16×16 are collected from "BasketballDrive", "BQMall", and "BasketballPass" under QP=27 and QP=37 to test the prediction accuracy. The relationship between the threshold and $\mathcal{R}_{\text{acc}}$ is demonstrated in Fig. 8.
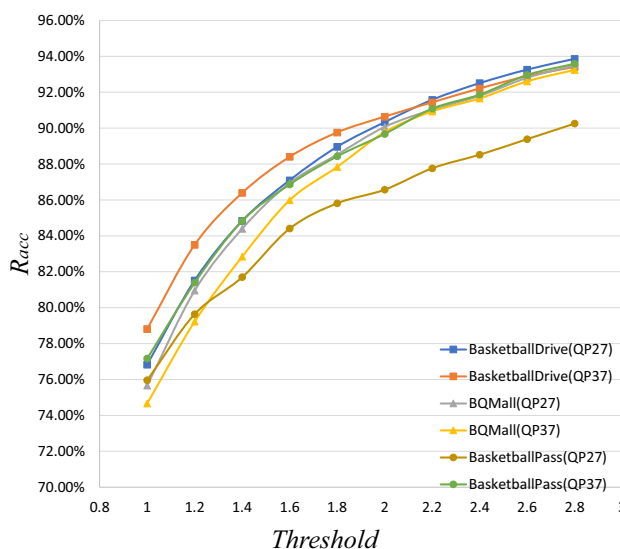

Fig. 8 Ablation results of the prediction accuracy under various thresholding values

Moreover, we also carry out experiments to reveal the partitioning results under different QPs. As shown in Fig. 9, we can intuitively observe that the number of CUs encoded by 32×32 increases with a higher QP. Meanwhile, the number of CUs encoded by 16×16 decreases with a higher QP. It indicates that we can update the threshold value along with the QP increasing. In view of the RD performance, since the mistaken classification of a larger CU will generate more BD rate than a smaller CU, we take a tight judgement condition for high QPs. In our method, when the QP value is greater than 28, we set the threshold $\mathcal{T}_h$ as 1.7; otherwise, we set the threshold $\mathcal{T}_h$ as 1.2. As presented in Fig. 8, the overall accuracy of the proposed method ranges from 80% to 90%. To simplify the modeling process, the $\mathcal{T}_l$ is defined as the reciprocal of $\mathcal{T}_h$, because the number of vertical and horizontal partition modes is at a similar numeral level generally for a specific sequence.

3.4 The overall algorithm

The pseudo-code of the overall algorithm can be summarized as Algorithm 1. The main idea is described as follows. If the size of the current CU is 32×32 or 16×16, the fast algorithm is performed for intra-prediction. We calculate the C_{mad} values for all sub-blocks in the vertical and horizontal directions, which is used to measure the complexity. And the ratio $\mathcal{R}_{\text{ver/hor}}$ of the vertical-to-horizontal complexity is calculated to compare the possibility of the texture direction. If the $\mathcal{R}_{\text{ver/hor}}$ value is larger than the higher threshold $\mathcal{T}_h$, the vertical MT partitioning mode is skipped, including BV and TV. If the $\mathcal{R}_{\text{ver/hor}}$ value is less than the lower threshold $\mathcal{T}_l$, the horizontal MT partition is skipped, including BH and

(a) BasketballDrive (1920 × 1080)

(b) BQMall (832 × 480)

(c) BasketballPass (416 × 240)
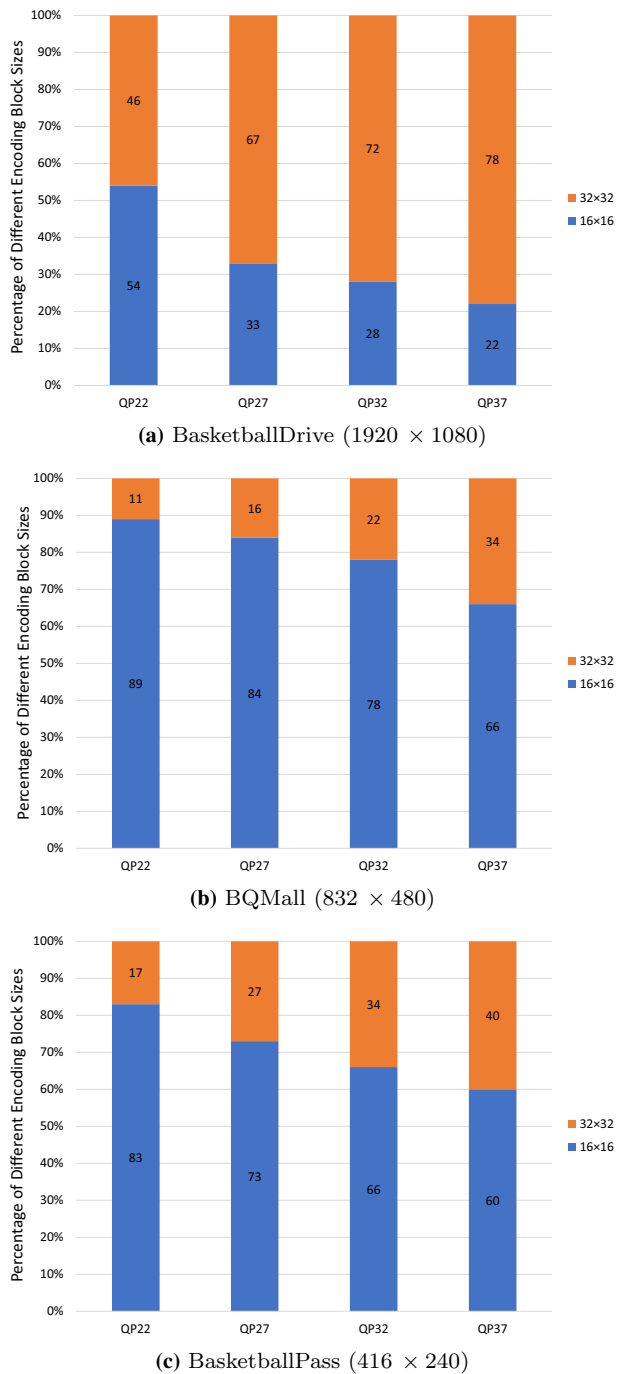
Fig. 9 Illustrations of the relationship between the block partition and quantization parameters. Two different CU sizes (16 × 16 and 32 × 32) and four different QPs (22, 27, 32, 37) are evaluated in the experiments

TH. Otherwise, we consider that the texture direction is not apparent, and choose the default partition modes of the VVC reference software. The flowchart of the proposed algorithm is shown in Fig. 10.
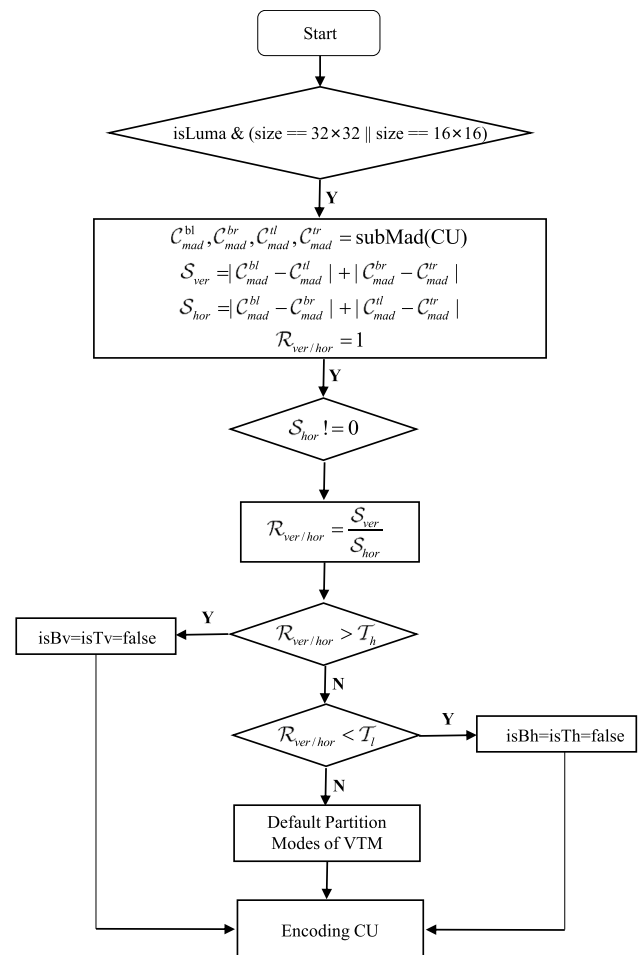
Fig. 10 Flowchart of the proposed algorithm

Algorithm 1: The pseudo-code of the proposed fast block decision algorithm.

if isLuma **and** size $==32{\times}32$ **or** size $==16{\times}16$ **then**
$\quad \mathcal{C}_{mad}^{bl}, \mathcal{C}_{mad}^{bl}, \mathcal{C}_{mad}^{tl}, \mathcal{C}_{mad}^{tr} = subMad(CU)$
$\quad \mathcal{S}_{ver} = |\mathcal{C}_{mad}^{bl} - \mathcal{C}_{mad}^{tl}| + |\mathcal{C}_{mad}^{br} - \mathcal{C}_{mad}^{tr}|$
$\quad \mathcal{S}_{hor} = |\mathcal{C}_{mad}^{bl} - \mathcal{C}_{mad}^{br}| + |\mathcal{C}_{mad}^{tl} - \mathcal{C}_{mad}^{tr}|$
$\quad$**if** $\mathcal{S}_{hor} \neq 0$ **then**
$\quad\quad \mathcal{R}_{ver/hor} = \mathcal{S}_{ver}/\mathcal{S}_{hor}$
$\quad$**end if**
$\quad$**if** $\mathcal{R}_{ver/hor} > \mathcal{T}_h$ **then**
$\quad\quad isBv = false$ **and** $isTv = false$
$\quad$**end if**
$\quad$**if** $\mathcal{R}_{ver/hor} < \mathcal{T}_l$ **then**
$\quad\quad isBh = false$ **and** $isTh = false$
$\quad$**end if**
end if

---

4 Experimental results

To evaluate the performance of the proposed fast method, we conduct the experiments on the VVC reference software [1]. A total of 22 common test sequences belong to

six categories with various resolutions: A1 (3840×2160), A2 (3840×2160), B (1920×1080), C (832×480), D (416×240), and E (1280×720). The encoding parameters take the recommended setting by the JVET common test condition (CTC) [5] configurations under QP = 22, 27, 32, and 37, respectively.

We carry out the experiments based on the computing platform with Intel(R) Core (TM) i5-6200U CPU @ 2.30 GHz, 8 GB RAM with Microsoft Visual Studio C++ 2017 compiler, and we use the open-source computer vision (OpenCV) 3.4.1 library. The encoding efficiency is measured by Bjøntegaard delta peak signal-to-noise ratio (BDPSNR) and Bjøntegaard delta bit rate (BDBR) [4]. The time saving, T_s, is measured by

$$T_s = \frac{T_o - T_p}{T_o} \times 100\%, \tag{6}$$

where T_o denotes the total encoding time of the original VVC, and T_p denotes the total encoding time of the proposed method.

4.1 Overall performance

Table 2 provides the overall encoding results of the proposed method. The proposed texture-based fast decision strategy achieves the time saving from 26.55 to 35.63% for various video sequences. The average encoding time saving is 30.33% compared with the anchor. More importantly, the complexity reduction has a negligible effect on the encoding efficiency; for example, 0.57% BDBR increase or 0.03 dB BDPSNR decrease on average.

4.2 Rate-distortion comparison

Figure 11 illustrates the overall rate-distortion curves of six typical sequences from Classes A1 to E: "Campfire" (3840×2160), "ParkRunning3" (3840×2160), "BasketballDrive" (1920×1080), "BQMall" (832×480), "BasketballPass" (416×240), and "Johnny" (1280×720) with QP= 22, 27, 32, and 37, respectively. It can be seen that the RD curves of the original VVC anchor and the proposed method are approximately identical for all video sequences. The results verify the effectiveness of the proposed scheme in reducing the computational complexity but maintaining the coding efficiency.

4.3 Complexity reduction efficiency comparison

Complexity reduction efficiency is an important indicator to measure the complexity reduction versus the compress bit-rate increment. It is defined by
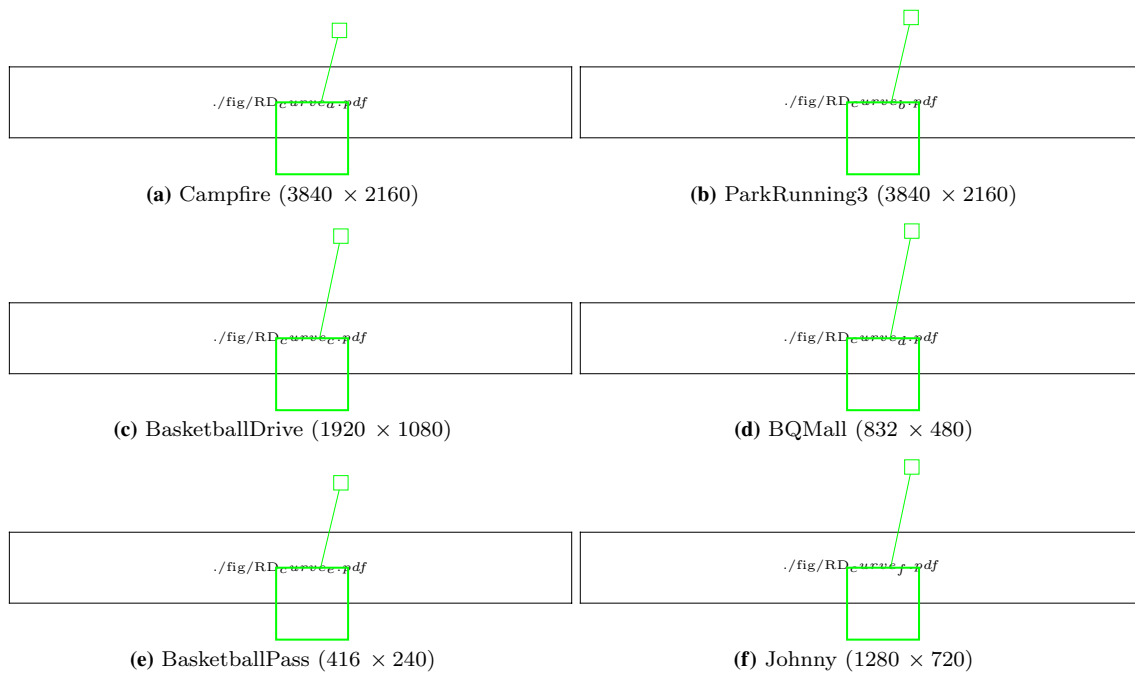
Table 2 Experimental results of the proposed method under the all-intra configuration

Class	Sequence	BDPSNR(dB)	BDBR(%)	T_s(%)
A1	Tango2	− 0.01	0.60	29.44
	FoodMarket4	− 0.02	0.52	29.37
	Campfire	− 0.02	0.59	29.28
A2	CatRobot1	− 0.02	0.74	28.85
	DaylightRoad2	− 0.02	0.92	35.03
	ParkRunning3	− 0.02	0.27	28.34
B	MarketPlace	− 0.02	0.42	29.43
	RitualDance	− 0.03	0.64	28.39
	Cactus	− 0.02	0.61	30.73
	BasketballDrive	− 0.02	0.74	34.48
	BQTerrace	− 0.03	0.62	28.85
C	RaceHorses	− 0.03	0.46	27.83
	BQMall	− 0.03	0.65	33.79
	PartyScene	− 0.03	0.42	31.62
	BasketballDrill	− 0.02	0.40	26.55
D	RaceHorses	− 0.03	0.48	27.84
	BQSquare	− 0.02	0.29	29.97
	BlowingBubbles	− 0.03	0.43	29.34
	BasketballPass	− 0.04	0.70	30.53
E	FourPeople	− 0.04	0.78	35.63
	Johnny	− 0.03	0.69	30.65
	KristenAndSara	− 0.03	0.59	31.38
Average		− 0.03	0.57	30.33

$$E_c = \frac{T_s}{\text{BDBR}} \times 100\%. \tag{7}$$

The performance comparison results of the complexity reduction efficiency are summarized in Table 3. Three recent advances are employed to demonstrate the comparison results, including FU2019 [13], CHEN2019 [9], and YANG2019 [32]. Sine CHEN2019 [9] only performed the experiments on the sequences of 8-bit depth, we provided the results on class B–E for fair comparison. FU2019 reduces 42.36% of the encoding time at the cost of the BDBR increase by 0.93%, CHEN2019 saves the encoding time by 52.48% at the cost of the BDBR increase by 1.59%, and YANG2019 achieves 52.98% of the complexity reduction at the cost of the BDBR increase by 1.86%.

In terms of the complexity reduction efficiency, our method outperforms three methods with the average performance $E_c = 54.60\%$. The other three methods have an explicitly low efficiency. For instance, the average complexity reduction efficiency results of FU2019 [13], CHEN2019 [9], and YANG2019 [32] are 45.44%, 33.08%, and 28.47%, respectively.

(a) Campfire (3840 × 2160)

(b) ParkRunning3 (3840 × 2160)

(c) BasketballDrive (1920 × 1080)

(d) BQMall (832 × 480)

(e) BasketballPass (416 × 240)

(f) Johnny (1280 × 720)

Fig. 11 Rate-distortion curve comparison between the proposed method and the VVC anchor reference software

Table 3 Comparison results of the proposed algorithm and thee recent advances on four QPs (22, 27, 32, and 37)

Class	Sequence	FU2019 [13]			CHEN2019 [9]			YANG2019 [32]			Proposed		
		BDBR (%)	T_s (%)	E_c (%)	BDBR (%)	T_s (%)	E_c (%)	BDBR (%)	T_s (%)	E_c (%)	BDBR (%)	T_s (%)	E_c (%)
B	Cactus	0.96	44.00	45.83	1.74	55.82	32.08	1.95	56.07	28.75	0.61	30.73	50.38
	BasketballDrive	0.95	49.00	51.58	3.09	65.26	21.12	2.25	64.01	28.45	0.74	34.48	46.59
	BQTerrace	0.79	48.00	60.76	1.16	49.44	42.62	2.07	56.07	27.09	0.62	28.85	46.53
C	RaceHorses	0.71	42.00	59.15	0.84	52.07	61.99	1.16	48.39	41.72	0.46	27.83	60.50
	BQMall	0.87	39.00	44.83	1.79	56.49	31.56	2.15	55.23	25.69	0.65	33.79	51.98
	PartyScene	0.46	42.00	91.30	0.28	41.71	148.96	0.60	45.73	76.22	0.42	31.62	75.29
	BasketballDrill	1.89	46.00	24.34	1.91	53.05	27.77	2.01	48.19	23.98	0.40	26.55	66.38
D	RaceHorses	0.75	39.00	52.00	0.54	44.93	83.20	0.86	43.17	50.20	0.48	27.84	58.00
	BQSquare	0.47	44.00	93.62	0.17	32.34	190.24	0.81	46.06	56.86	0.29	29.97	103.34
	BlowingBubbles	0.56	37.00	66.07	0.49	43.9	89.59	1.31	57.16	43.63	0.43	29.34	68.23
	BasketballPass	0.70	43.00	61.43	2.02	54.16	26.81	2.33	45.85	19.68	0.70	30.53	43.61
E	FourPeople	1.37	41.00	29.93	2.55	62.18	24.38	2.75	57.64	20.96	0.78	35.63	45.68
	Johnny	1.33	39.00	29.32	3.07	62.55	20.37	3.29	58.98	17.93	0.69	30.65	44.42
	KristenAndSara	1.24	40.00	32.26	2.56	60.82	23.76	2.51	59.19	23.58	0.59	31.38	53.19
Average		0.93	42.36	45.44	1.59	52.48	33.08	1.86	52.98	28.47	0.56	30.66	54.60

5 Conclusion

In this paper, we propose a high-efficient low-complexity block partition scheme for VVC intra coding. We evaluate the block texture complexity in the vertical and horizontal directions by the sum of mean absolute deviation between sub-blocks. The relationship between the vertical and horizontal texture complexities is statistically counted for 32×32 and 16×16 blocks, which verifies the relevance between the directional texture complexity feature and the best partition mode. Experimental results demonstrate that the proposed method obtains the average complexity

reduction efficiency by up to 103.34% under the common test condition.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

References

1. JVET Group: VVC Reference Software. [online] https://vcgit.hhi.fraunhofer.de/jvet/ VVCSoftware_VTM/-/tags/VTM-6.3 (2019)
2. Amestoy, T., Mercat, A., Hamidouche, W., Menard, D., Bergeron, C.: Tunable VVC frame partitioning based on lightweight machine learning. IEEE Trans. Image Process. **29**, 1313–1328 (2019)
3. Bakkouri, S., Elyousfi, A.: Machine learning-based fast cu size decision algorithm for 3D-hevc inter-coding. Springer J. Real-Time Image Process. **18**(3), 983–995 (2021)
4. Bjøntegaard, G.: Calculation of average PSNR differences between RD-curves. In: Proceedings of the VCEG-M33, p 1–4 (2001)
5. Bossen, F., Boyce, J., Suehring, K., Li, X., Seregin, V.: JVET common test conditions and software reference configurations. In: Joint Video Experts Team (JVET) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11 11th Meeting, p 1–6 (2018)
6. Bouaafia, S., Khemiri, R., Sayadi, F.E., Atri, M.: Fast cu partition-based machine learning approach for reducing hevc complexity. Springer J. Real-Time Image Process. **17**(1), 185–196 (2020)
7. Bross, B.: Versatile video coding (draft 1). In: Joint Video Experts Team (JVET) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11 10th Meeting (2018)
8. Chen, F., Ren, Y., Peng, Z., Jiang, G., Cui, X.: A fast CU size decision algorithm for VVC intra prediction based on support vector machine. Multimed. Tools Appl. **79**(37), 27923–27939 (2020)
9. Chen, J., Sun, H., Katto, J., Zeng, X., Fan, Y.: Fast QTMT partition decision algorithm in VVC intra coding based on variance and gradient. In: IEEE Visual Communications and Image Processing (VCIP), pp. 1–4 (2019)
10. Chuang, T.D., Chen, C.Y., Huang, Y.W., Lei, S.M.: CE1-related: Separate tree partitioning at 64x64-luma/32x32-chroma unit level. In: Joint Video Experts Team (JVET) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11 11th Meeting, p. 1–3 (2018)
11. Cui, J., Zhang, T., Gu, C., Zhang, X., Ma, S.: Gradient-based early termination of CU partition in VVC intra coding. In: IEEE Data Compression Conference (DCC), pp. 103–112 (2020)

12. De-Luxán-Hernández, S., George, V., Ma, J., Nguyen, T., Schwarz, H., Marpe, D., Wiegand, T.: An intra subpartition coding mode for VVC. In: IEEE International Conference on Image Processing (ICIP), pp. 1203–1207 (2019)
13. 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), pp. 55–60 (2019)
14. Grellert, M., da Silva Cruz, L.A., Zatt, B., Bampi, S.: Coding mode decision algorithm for fast hevc transrating using heuristics and machine learning. Springer J. Real-Time Image Process., 1–16 (2021)
15. Hamout, H., Elyousfi, A.: An efficient edge detection algorithm for fast intra-coding for 3d video extension of hevc. Springer J. Real-Time Image Process. **16**(6), 2093–2105 (2019)
16. Hosseini, E., Pakdaman, F., Hashemi, M.R., Ghanbari, M.: Fine-grain complexity control of hevc intra prediction in battery-powered video codecs. Springer J. Real-Time Image Process. **18**(3), 603–618 (2021)
17. Huang, H., Zhang, K., Huang, Y.W., Lei, S.: EE2.1: Quadtree plus binary tree structure integration with JEM tools. In: Joint Video Exploration Team (JVET) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11 3rd Meeting (2016)
18. Laroche, G., Taquet, J., Gisquet, C., Onno, P.: CE3-5.1: On cross-component linear model simplification. In: Joint Video Experts Team (JVET) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11 12th Meeting, p. 1–3 (2018)
19. Lei, M., Luo, F., Zhang, X., Wang, S., Ma, S.: Look-ahead prediction based coding unit size pruning for VVC intra coding. In: IEEE International Conference on Image Processing (ICIP), pp. 4120–4124 (2019)
20. Li, Y., Yang, G., Zhu, Y., Ding, X., Song, Y., Zhang, D.: Hybrid stopping model-based fast pu and cu decision for 3d-hevc texture coding. Springer J. Real-Time Image Process. **17**(5), 1227–1238 (2020)
21. Lim, K., Lee, J., Kim, S., Lee, S.: Fast PU skip and split termination algorithm for HEVC intra prediction. IEEE Trans. Circuits Syst. Video Technol. **25**(8), 1335–1346 (2014)
22. Min, B., Cheung, R.C.: A fast CU size decision algorithm for the HEVC intra encoder. IEEE Trans. Circuits Syst. Video Technol. **25**(5), 892–896 (2014)
23. Said, A., Zhao, X., Karczewicz, M., Chen, J., Zou, F.: Position dependent prediction combination for intra-frame video coding. In: IEEE International Conference on Image Processing (ICIP), pp. 534–538 (2016)
24. Shang, X., Wang, G., Fan, T., Li, Y.: Fast CU size decision and PU mode decision algorithm in HEVC intra coding. In: IEEE International Conference on Image Processing (ICIP), pp. 1593–1597 (2015)
25. Shen, L., Zhang, Z., Liu, Z.: Effective CU size decision for HEVC intracoding. IEEE Trans. Image Process. **23**(10), 4232–4241 (2014)
26. Tahir, M., Taj, I.A., Assuncao, P.A., Asif, M.: Fast video encoding based on random forests. Springer J. Real-Time Image Process., 1–21 (2019)
27. Tang, G., Jing, M., Zeng, X., Fan, Y.: Adaptive CU split decision with pooling-variable CNN for VVC intra encoding. In: IEEE Visual Communications and Image Processing (VCIP), pp. 1–4 (2019)
28. Tang, N., Cao, J., Liang, F., Wang, J., Liu, H., Wang, X., Du, X.: Fast CTU partition decision algorithm for VVC intra and inter coding. In: IEEE Asia Pacific Conference on Circuits and Systems (APCCAS), pp. 361–364 (2019)
29. Varma, K.R.C., Mahapatra, S.: Complexity reduction of test zonal search for fast motion estimation in uni-prediction of high efficiency video coding. Springer J. Real-Time Image Process. **18**, 511–524 (2021)
30. Wang, J., Ji, B., Wang, H., Cheng, L.: Prediction mode grouping and coding bits grouping based on texture complexity for fast

hevc intra-coding. Springer J. Real-Time Image Process. **18**(3), 839–856 (2021)

31. Wang, M., Xie, W., Meng, X., Zeng, H., Ngan, K.N.: Uhd video coding: a light-weight learning-based fast super-block approach. IEEE Trans. Circuits Syst. Video Technol. **29**(10), 3083–3094 (2018)

32. 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. Circuits Syst. Video Technol. **30**(6), 1668–1682 (2019)

33. Yang, Z., Shao, Q., Guo, S.: Fast coding algorithm for HEVC based on video contents. IET Image Proc. **11**(6), 343–351 (2017)

34. Zhang, R., Jia, K., Liu, P., Sun, Z.: Fast intra-mode decision for depth map coding in 3d-hevc. Springer J. Real-Time Image Process. **17**(5), 1637–1646 (2020)

35. Zhang, Y., Pan, Z., Li, N., Wang, X., Jiang, G., Kwong, S.: Effective data driven coding unit size decision approaches for HEVC INTRA coding. IEEE Trans. Circuits Syst. Video Technol. **28**(11), 3208–3222 (2017)

36. Zhao, J., Wang, Y., Zhang, Q.: Adaptive CU split decision based on deep learning and multifeature fusion for H.266/VVC. Sci. Program. **2020** (2020)

37. Zhao, L., Zhao, X., Liu, S., Li, X., Lainema, J., Rath, G., Urban, F., Racapé, F.: Wide angular intra prediction for versatile video coding. In: IEEE Data Compression Conference (DCC), pp. 53–62 (2019)

38. Zhu, W., Yi, Y., Zhang, H., Chen, P., Zhang, H.: Fast mode decision algorithm for HEVC intra coding based on texture partition and direction. J. Real-Time Image Proc. **17**(2), 275–292 (2020)

Yun Song received his M.S. and Ph.D. degrees in computer science from Changsha University of Science and Technology and Hunan University, China in 2008 and 2016, respectively. He is currently is an associate professor in the School of Computer and Communication Engineering of Changsha University of Science and Technology, China. His major research interests include video coding, multimedia communication and security, compressive sensing, video analysis, and image processing.

Biao Zeng received the B.E. degree from Changsha University of Science and Technology, P. R. China. He is currently pursuing the M.E. degree at School of Computer and Communication Engineering, Changsha University of Science and Technology, Changsha, P. R. China. His main research interests are image processing and video coding.
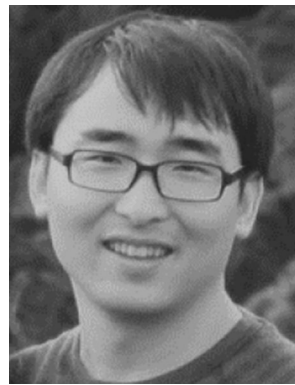
Miaohui Wang received the Ph.D. degree from Department of Electronic Engineering, The Chinese University of Hong Kong (CUHK), Hong Kong, P. R. China. From 2014 to 2015, he was a researcher working on the standardization of video coding at the Innovation Laboratory, InterDigital Inc., San Diego, CA, USA. From 2015 to 2017, He was a senior research staff working on computer vision and machine learning at The Creative Life (TCL) Research Institute of Hong Kong, Hong Kong, P. R.

China. Dr. Wang was the recipient of the Best Thesis Awards from the Ministry of Education of Shanghai City and Fudan University (FDU), respectively. He received the Best Paper Award in *International Conference on Advanced Hybrid Information Processing* (2018), and received the second runner-up of Grand Challenge on "Learning-based Image Inpainting" in *International Conference on Multimedia and Expo* (ICME 2019). He was the recipient of the outstanding reviewer for ICME 2021. He has authored or co-authored over 50 peer-reviewed papers in international journals and conferences. His current research interests cover a wide range of topics related with high-dimension visual data compression and processing, medical image analysis, computer vision and machine learning.

Zelin Deng received his M.S. and Ph.D. degrees in computer science from Huazhong University of Science and Technology and Central South University, China in 2003 and 2013, respectively. He is an associate professor at the School of Computer and Communication Engineering of Changsha University of Science and Technology. His major research interests include pattern recognition, image processing, and evolutionary computation.