Fast ISP coding mode optimization algorithm based on CU texture complexity for VVC

Zhi Liu[1]* , Mengjun Dong[1], Xiao Han Guan[1], Mengmeng Zhang[1,2]* and Ruoyu Wang[3]

* Correspondence: lzliu@ncut.edu.
cn; muchmeng@126.com
[1]North China University of
Technology, Beijing 100144, China
Full list of author information is
available at the end of the article

Abstract

In lately published video coding standard Versatile Video Coding (VVC/ H.266), the intra sub-partitions (ISP) coding mode is proposed. It is efficient for frames with rich texture, but less efficient for frames that are very flat or constant. In this paper, by comparing and analyzing the rate distortion cost (RD-cost) of coding unit (CU) with different texture features for using and not using ISP(No-ISP) coding mode, it is observed that CUs with simple texture can skip ISP coding mode. Based on this observation, a fast ISP coding mode optimization algorithm based on CU texture complexity is proposed, which aims to determine whether a CU needs to use ISP coding mode in advance by calculating CU texture complexity, so as to reduce the computation complexity of ISP. The experimental results show that under All Intra (AI) configuration, the coding time can be reduced by 7%, while the BD rate only increase by 0.09%.

Keywords: ISP, VVC, H.266, Intra prediction, Coding unit (CU), Texture complexity

1 Introduction

Intra prediction has always been the main research field in video coding, which can take advantage of the spatial correlation of image to eliminate the spatial information redundancy and realize the compression of video data. In VVC, many new intra prediction techniques have been proposed, including mode-dependent intra smoothing (MDIS), cross-component linear model (CCLM), position-dependent intra prediction combination (PDPC), multiple reference line (MRL) intra prediction, intra sub-partitions (ISP), matrix-weighted intra prediction (MIP), and so on [1]. Intra prediction technology plays an important role in video coding technology and has a great impact on the coding performance. It is a crucial task to optimize the intra prediction technique to improve coding efficiency.

In the process of intra prediction, the reference samples that can be used to create intra prediction signals are located only on the left and above of the current block. As the correlation between samples in the natural image decreases with the increasing of distance, the quality of the predicted samples near the bottom-right corner of the block

will be worse than those close to the top-left boundary of block [2]. In order to solve this problem, an intra sub-partitions coding (ISP) mode has been proposed in VVC, which divides the luminance intra prediction blocks horizontally or vertically into 4 or 2 equal size sub-partitions, with each contains at least 16 samples. The minimum sub-partition size and the maximum block size that can use ISP coding mode are 4 × 8 (or 8 × 4) and 64 × 64, respectively. If the size of a block is greater than 4 × 8 (or 8 × 4), the corresponding block is divided into 4 sub-partitions. If the size of block is equal to 4 × 8 (or 8 × 4), the corresponding block is divided into 2 sub-partitions [3]. Given an input W × H block, the size of the sub-partition will be $W \times (H/K)$ for horizontal split and $(W/K) \times H$ for vertical split, where K is the number of sub-partitions, and W and H represent the width and height of the block, respectively. As shown in Fig. 1, a 32 ×
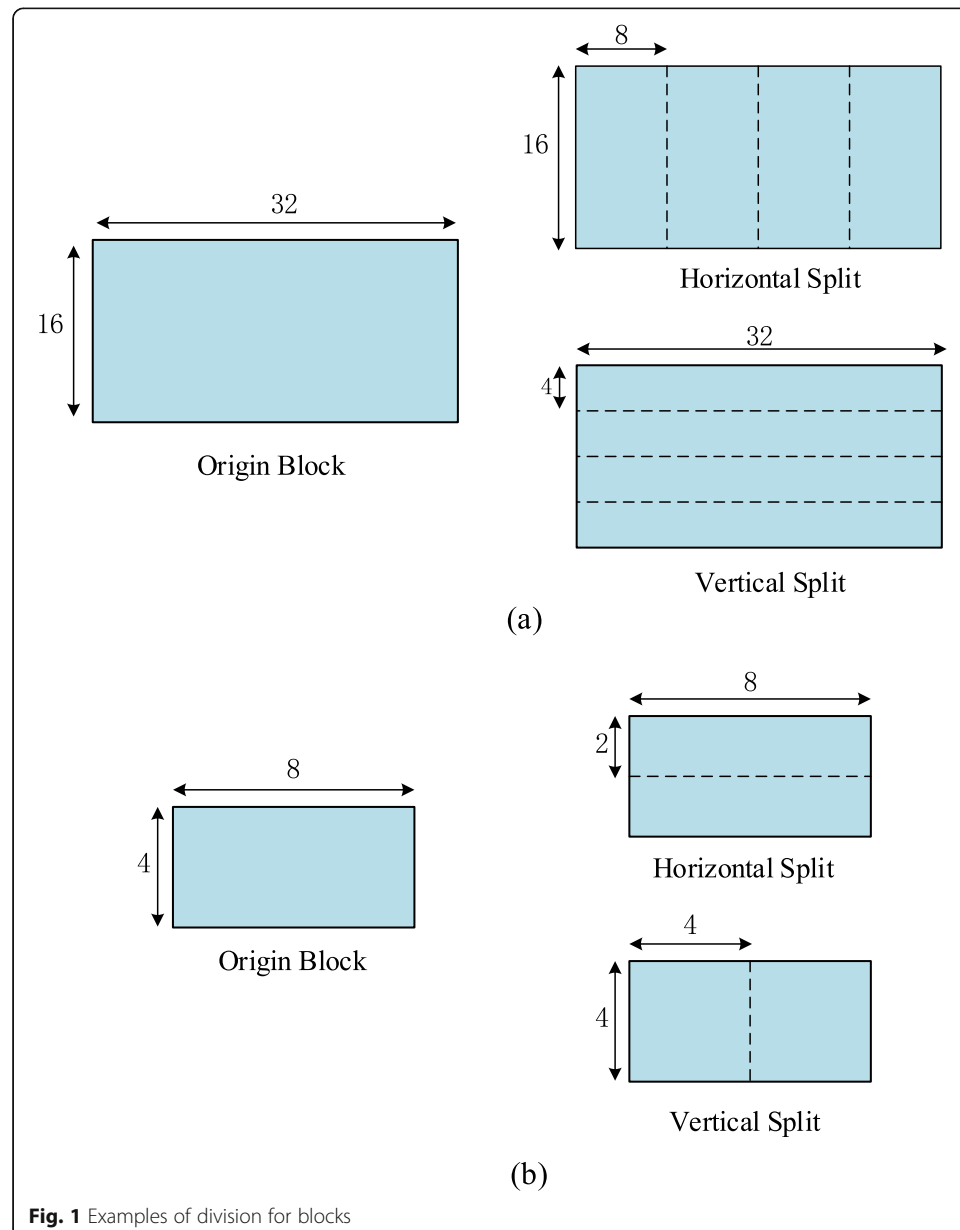
Fig. 1 Examples of division for blocks

16 block can be divided into four 8 × 16 or four 32 × 4 sub-partitions, and a 8 × 4 block can be divided into two 8 × 2 sub-partitions or two 4 × 4 sub-partitions.

In VVC, the processing process of each sub-partition is similar to that of intra prediction block. At the encoder side, the residual signal is generated based on the prediction signal, and then the residual signal is transformed, quantized, and entropy encoded and sent to the decoder. At the decoder side, the residual signal is recovered after a series of steps including entropy decoding, inverse quantization, and inverse transformation, and the reconstructed samples can be obtained by adding the residual signal with the prediction signal. Once a sub-partition has been processed, its reconstructed samples can be used to calculate the prediction signal of the next sub-partition, and this step will repeat over and over until all sub-partitions have been encoded [4]. As shown in Fig. 2, the block has been split horizontally into four sub-partitions. The first one is predicted using the neighboring samples of the CU. Afterwards, its reconstructed samples can be used to predict the next sub-partition. The procedure continues until the four sub-partitions have been processed. The advantage of the ISP coding mode is that each sub-partitions can be predicted using neighboring samples located at the shortest possible distance; therefore, the coding efficiency is improved.

Although ISP coding mode can improve coding efficiency, it still has some deficiencies. In this paper, by analyzing the rate-distortion cost of CU with different texture characteristics in ISP coding mode, it is observed that the coding efficiency of CU with simple texture is not evidently improved in ISP coding mode, although extra computation cost has been spent. Based on this observation, a fast decision algorithm for ISP coding mode based on the CU texture complexity is proposed, which aims to determine whether a CU needs to use ISP coding mode in advance by calculating CU texture complexity, so as to evidently reduce the computation complexity of ISP while keeping the coding efficiency nearly no degrade. In addition, a method for calculating the CU texture complexity based on interval sampling is designed, and the decision threshold is optimized by using the distortion cost.

Fig. 2 An example of ISP coding mode
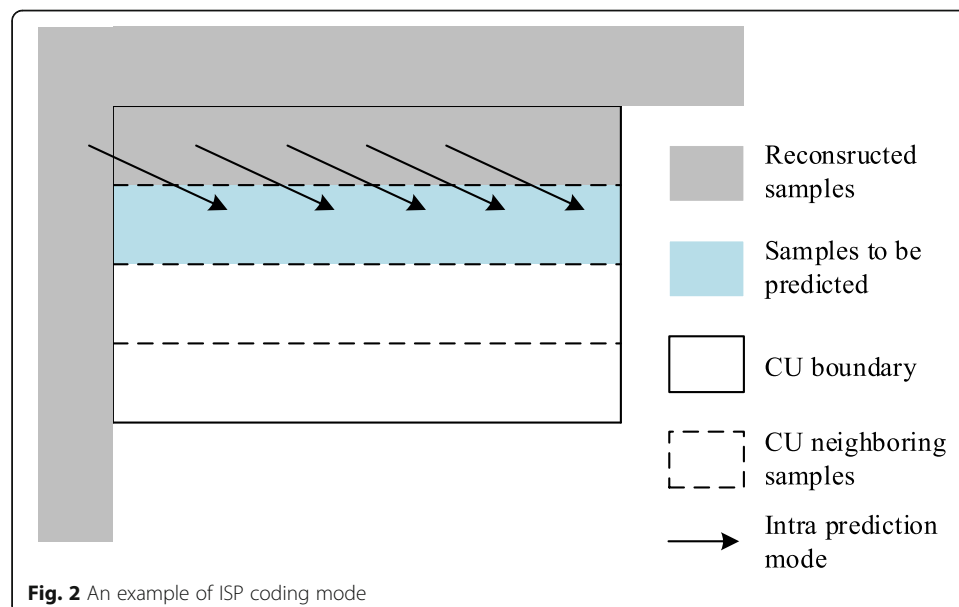
The remainder of this paper is organized as follows. In Section 2, the related works of ISP and the video coding algorithm based on texture feature are presented. In Section 3, the existing problems of ISP coding mode are analyzed and a fast ISP coding mode optimization algorithm based on CU texture complexity is proposed. Experiment results are shown in Section 4. Finally, in Section 5, conclusions are drawn.

2 Related works

ISP coding mode is developed from the line-based intra prediction (LIP) coding mode, which was first presented at the Kth meeting of JVET [5]. The main idea of LIP is to divide the luma intra prediction block into one-dimensional line. Each block needs to be partitioned using the LIP coding mode. In the AI configuration, compared to the standard reference encoder, the BD rate can be reduced by 2.34% on average, while the encoder running time is increased to 293%. A 4×4 block can be divided into 4 (4×1) lines, which can cause throughput problems. All blocks are 1×4 (or 4×1) and this kind of division leads to a bad bit stream. If the number of resulting lines for a block is large (for example, 64 rows), the encoder needs to do a lot of operation and memory access while checking the necessary rate-distortion (RD) tests. Column sub-partition ($1 \times N$) can be more difficult to implement because samples are allocated using raster scans, which makes memory access expensive. To solve the above problems, it was proposed at the Lth meeting of JVET that each block should be set into a certain number of partitions (each partition has at least 16 samples) and the final partition width should be at least 4 samples [6]. In AI configuration, the BD rate can be reduced by 1.01% on average, while the encoder running time becomes 148%, which successfully reduces the complexity of encoder. After this meeting, LIP was officially renamed as ISP. At the Mth meeting of JVET [7], the ISP algorithm was optimized to achieve a better balance between coding gain and encoder running time. The experimental results show that in the AI configuration, the BD rate can be reduced by 0.59% on average. At the same time, the coding running time becomes 112%. By analyzing the experimental results of these proposals, we have found that ISP coding mode can significantly improve the BD rate for coding blocks with rich texture, but it is inefficiency for coding blocks with simple texture, which inspired the ideas of this paper.

In recent years, many new tools have been developed and evidently improved the image processing performance [8–11]. However, considering the hardware cost and computation complexity, many of them are not applicable to video coding. Image texture characteristic is a lightweight tool and has been applied by many scholars to optimize video coding. Shen et al. [12] proposed an effective CU size decision algorithm, which utilizes the texture characteristics and coding information of adjacent CUs. Experimental results show that the algorithm can significantly reduce the intra coding time and achieve consistent acceleration of all kinds of video sequences. In order to solve the problem of high computational complexity of QTBT, Peng et al. [13] proposed a multiple classifier-based fast QTBT partitioning algorithm for intra coding. The feature acquisition of classifier is mainly based on texture complexity and direction complexity. The algorithm can reduce the intra coding time by 64.54%, and the reduction of RD performance is negligible. Liu et al. [14] proposed an adaptive fast CU size decision algorithm based on CU complexity classification for HEVC intra prediction by using machine learning technology and some image features are selected for training

classifiers in the paper. Hou et al. [15] proposed a method based on texture complexity for CU partitioning, which uses the complexity of adjacent CU to judge the current CU texture features. On this basis, useless CU size can be filtered out to reduce the encoding time. A fast mode decision algorithm based on texture partition and direction in HEVC was proposed in [16], which includes two sub-algorithms: CTU depth range prediction (CDRP) and internal prediction mode selection (IPMS). Under the AI configuration, the coding time of the proposed overall algorithm is reduced by 60% on average. In [17], the problem of the CU size decision and mode selection in intra-coding of HEVE is solved by measuring the texture complexity and directional energy distribution of each CU, which accelerate the process of RDO. An effective algorithm based on homogeneity for reducing the complexity of HEVC intra coding was proposed in [18]. The scheme aims to terminate the CU partitioning of homogeneous regions in video frames in advance, or to skip the CU partitioning of complex texture regions. The decision result of split/non-split is based on a homogenous classification algorithm that avoids testing all depths to determine the optimal CU size. In [19], a new fast intra frame coding algorithm is presented. Based on the analysis of the mode parameters obtained in the previous frame, a new feature is proposed to measure the complexity of video content and the model is built according to the relationship between the feature and the depth range of the CU. Based on the model, unnecessary operations of CU partitioning are skipped. In [20], a complexity reduction algorithm based on hierarchical classification for HEVC inter coding is proposed. At the beginning of the algorithm, the intra and inter features describing texture and context properties of CUs are obtained from the training set, and then the classification model is generated by selecting features.

To the best of our knowledge, up to now, there is no related works that adopt texture characteristics of CUs to optimize the ISP coding mode in VVC, and the proposed solution in this paper is the first to be applied to optimize the ISP coding mode.

3 Proposed algorithm

The ISP coding mode can improve the intra prediction efficiency of VVC. However, according to our observations and analysis, when the texture of a CU is very flat or constant, the coding efficiency of ISP coding mode will degrade evidently. A fast ISP coding mode optimization scheme based on CU texture complexity is proposed in this section, which can determine whether a CU needs to use ISP coding mode based on the analysis of CU texture complexity in advance, so as to save coding time with negligible gain loss.

3.1 Observation and analysis

The key to improve the encoder performance is to select the best coding parameters. The search for the best coding parameter is traditionally performed in the rate distortion (RD) sense, which can be balanced between the number of bits used to encode an image block and the distortion generated by using these number of bits. In the rate distortion optimization process of intra prediction, RD-cost is an excellent metric to measure prediction performance. By comparing the RD-cost of different partition

modes, the final partition mode will be determined. RD-cost function J is calculated as follow [21]:

$$J = D + \lambda \times R, \tag{1}$$

where D is the distortion, R is the number of bits required to transmit the residual coefficient and signaling parameter information, and λ is the Lagrange multiplier factor.

In this paper, the RD-cost of CU in ISP coding mode and not using ISP (No-ISP) coding mode is calculated respectively. The sequences for statistics are BasketballDrill, RaceHorses, BasketballPass, BlowingBubbles, FourPeople, and Kimono. The RD-cost of CU in ISP coding mode is denoted as J_{ISP}, and the RD-cost of CU in No-ISP coding mode is denoted as J_{No-ISP}. As shown in Fig. 3, the percentage of CU where J_{ISP} greater than J_{No-ISP} is only 40%, which indicates that not all CUs are suitable for ISP coding mode. It is found in this paper that the CUs whose RD-cost in ISP coding mode is higher than that in No-ISP coding mode have similar characteristics, which is that most of them have smooth texture. Therefore, for CUs with simple texture, fairly good prediction performance can be obtained in No-ISP coding mode. Figure 4 shows the first frame of FourPeople sequence. It can be found that block A and block B have rich texture, while the textures of block C and D are very smooth. The comparison of RD-cost in ISP and No-ISP coding mode for these four blocks is shown in Fig. 5. It is obvious that the RD-cost in ISP coding mode for block A and block B is significantly lower than that in No-ISP coding mode. However, the RD-cost in ISP coding mode for block C and block D is slightly higher than that in No-ISP coding mode.

In summary, for some CUs with simple texture, better prediction results will be obtained in No-ISP coding mode and in ISP coding mode. Therefore, for this kind of CUs, we can skip the ISP coding mode in advance, to save coding time. In the next subsection, a method to measure the texture complexity of CU is presented to determine whether CU needs to use ISP coding mode or not.
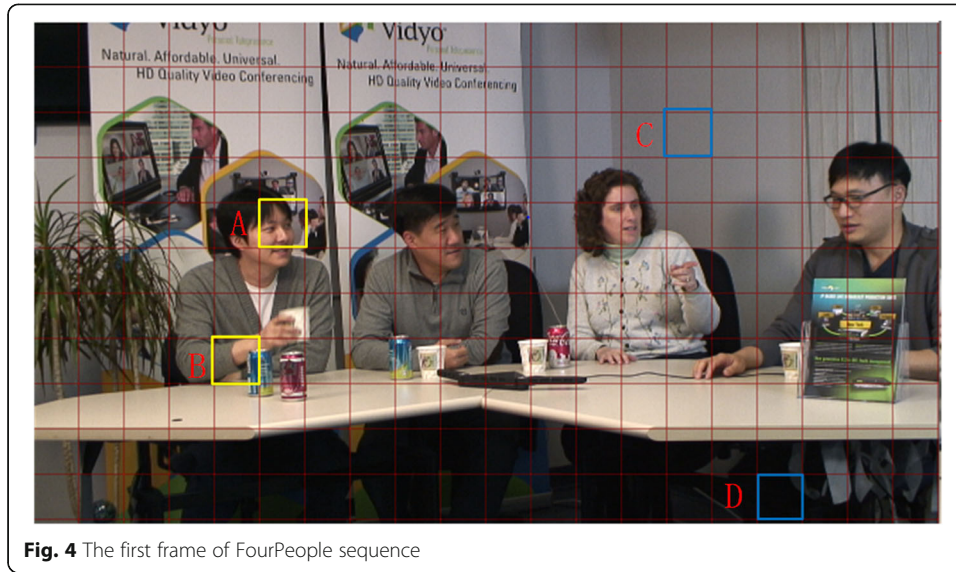

Fig. 3 CU distribution
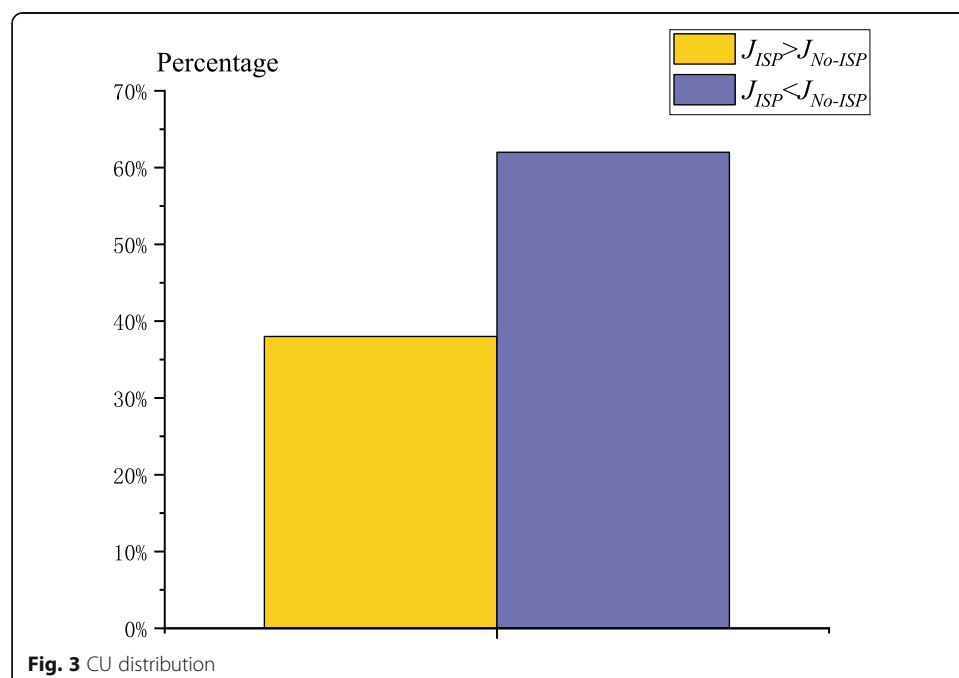
Fig. 4 The first frame of FourPeople sequence

3.2 Metric for measuring CU texture complexity

In order to determine a CU to be with simple texture or complex texture, a suitable metric is needed. MAD (mean absolute deviation) is the most representative metric, which is the average of the absolute deviation of a set of data. It can be used as an indicator to measure the texture complexity of CU. MAD is calculated as follows:

$$MAD = \frac{1}{\text{height} \times \text{width}} \sum_{j=1}^{height} \sum_{i=1}^{width} \mid P(x, y) - mean \mid, \qquad (2)$$

where *width* and *height* are the width and height of CU, respectively, and *P(i,j)* is the luminance value for pixel at *(i,j)*.

To further reduce the computational complexity, the calculation of MAD is simplified in this paper. Instead of calculating all pixel values, the interval sampling method is used to obtain pixel values. The sampling method is shown in Fig. 6. Odd points are sampled from odd rows of pixel points, and even points are sampled from even rows of pixel points. This sampling method can not only reduce the computation by half, but also accurately measure the texture complexity. The formula for calculation texture complexity (*TC*) used in this paper is shown in Formula (3).

$$TC = \frac{2}{\text{height} * \text{width}} \left(\sum_{j=2n-1}^{height} \sum_{i=2n-1}^{width} |P(i, j) - mean| + \sum_{j=2n}^{height} \sum_{i=2n}^{width} |P(i, j) - mean| \right) \text{ n} = 1, 2 \ldots \ldots, \quad (3)$$
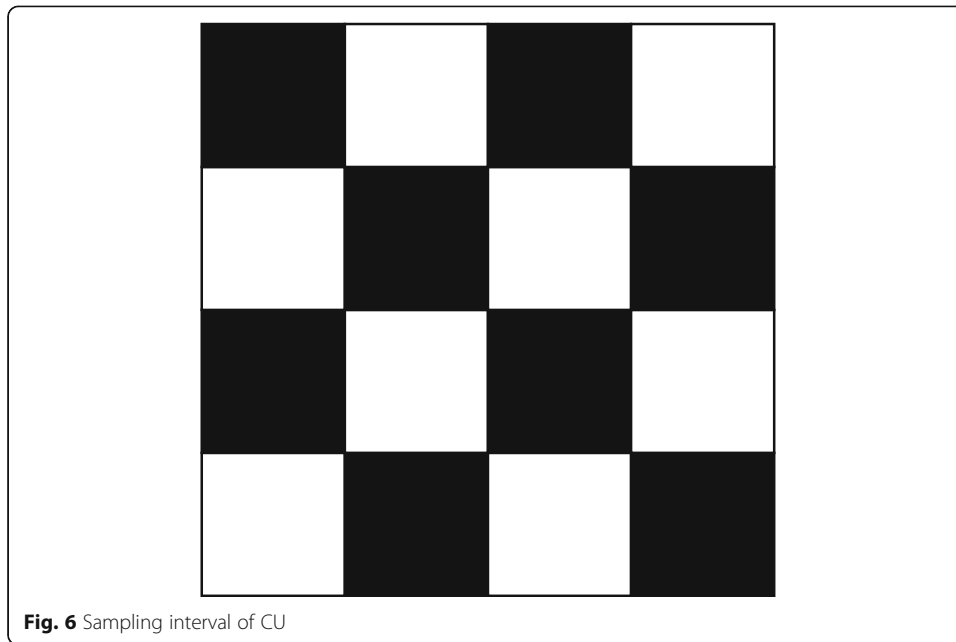
where *width* and *height* are the width and height of CU, respectively, and *P(i,j)* is the luminance value of pixel at *(i,j)*, and *mean* is the average of all sampled pixels.

According to the value of *TC*, CUs will be classified into two categories: simple texture and complex texture. When *TC* is less than the given threshold ε, the current CU is classified as simple texture. When *TC* is greater than ε, the CU is classified as complex texture. ε is the threshold to judge the texture complexity of CUs.

Threshold ε plays a key role in the proposed algorithm. Coding efficiency can be improved by selecting the appropriate threshold. According to Section 3.1, for some CUs

**Fig. 5** The comparison of RD-cost for ISP and No-ISP coding mode
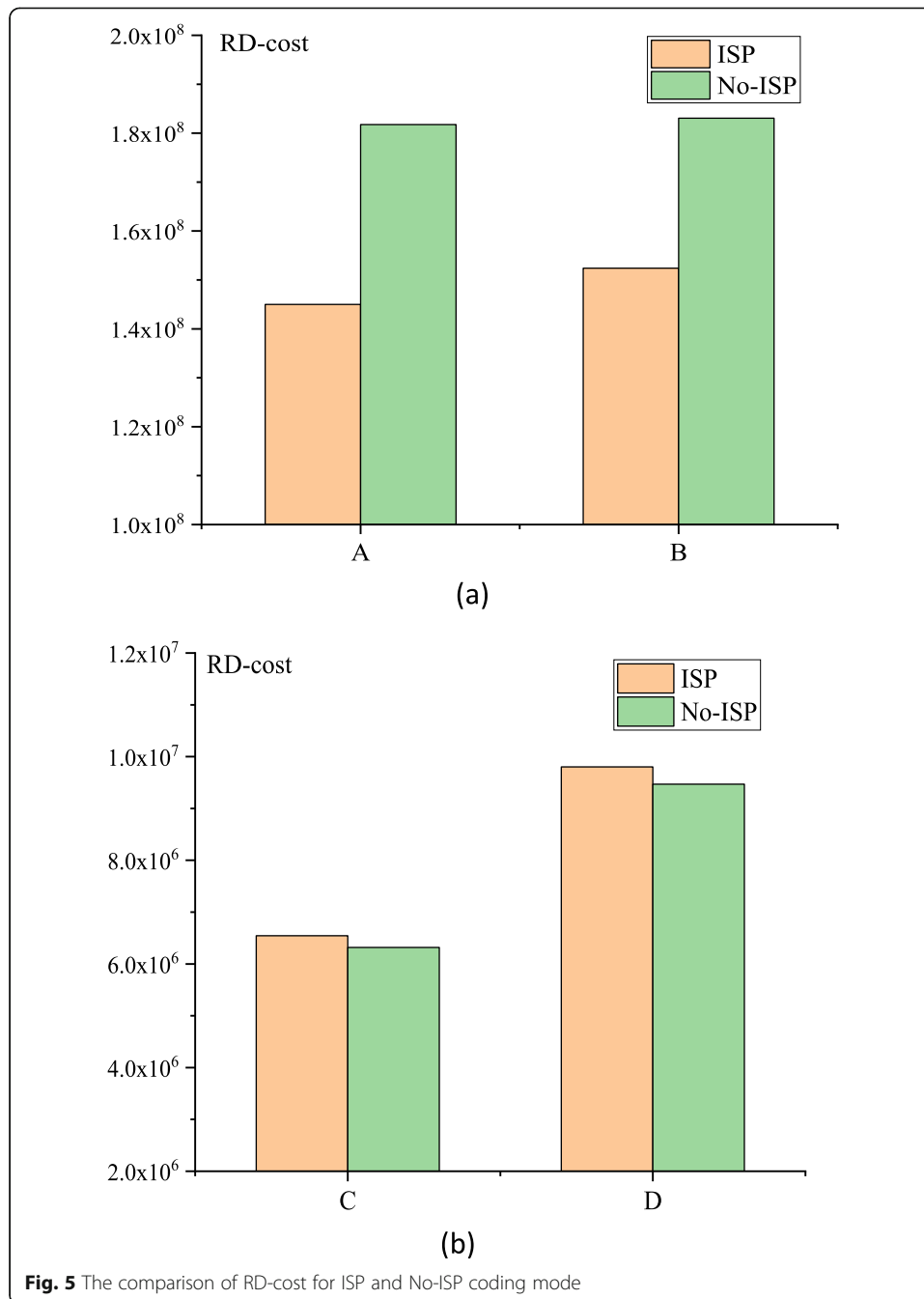
with simple texture, whose RD-cost in ISP coding mode is usually higher than that in No-ISP coding mode, are not suitable for using ISP coding mode. For some CUs with complex texture, whose RD-cost in ISP coding mode is usually lower than that in No-ISP coding mode, are suitable for using ISP coding mode. We take a statistical analysis for the CU luminance samples in the test video sequences, and TC of each CU is calculated to indicate the texture complexity. In the sequences used for statistics, the TC value of CU is divided into two groups. The first group is the CUs that are not suitable for using ISP coding mode. The second group is the CUs that are suitable for using ISP coding mode. The threshold value of the CU texture complexity is determined by the

Fig. 6 Sampling interval of CU

probability density of *TC*, which is shown in Fig. 7. As can be found from the figure, when the *TC* value is less than 20, the probability density of *TC* in the first group is always greater than that of in the second group. When *TC* is greater than 20, the probability density of *TC* in the first group is always smaller than that of in the second group. Therefore, in this paper, ε is set to 20. When the *TC* value of a CU is less than 20, it will be judged to skip ISP coding mode in advance during coding.

In this paper, the texture complexity distribution of CUs whose RD-cost in ISP coding mode is greater than that of in No-ISP coding mode is calculated. As shown in Fig. 8, the

Fig. 7 Probability density of TC

Fig. 8 CU texture complexity distribution
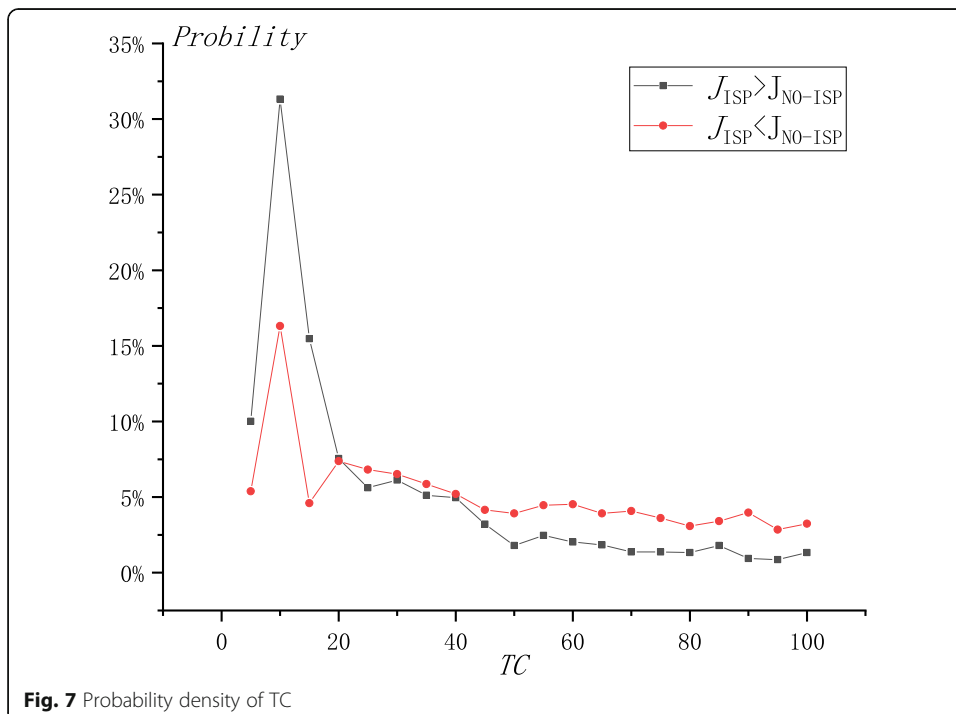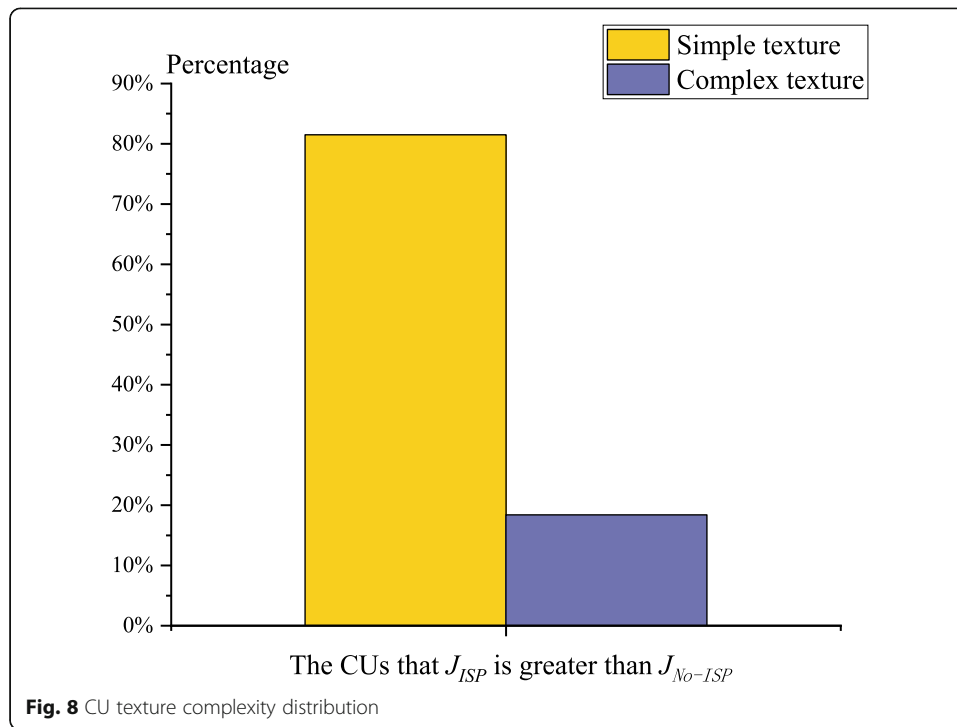
proportion of CU with simple texture is 80%, which further proves the sensible of the proposed algorithm.

3.3 Flow of the proposed algorithm

In Section 3.1, we have drawn the conclusion that the performance of ISP coding mode is closely related to CU texture complexity and a CU with simple texture can obtain fairly good coding performance in No-ISP coding mode. In Section 3.2, a metric named *TC* is designed to measure the CU complexity texture, and the decision threshold is set to be 20. According to the *TC* values of all CUs, they are divided into two groups: simple texture and complex texture. If the current CU is classified to have simple texture, the ISP coding mode will not be tested. Otherwise, if the current CU is classified to have complex texture, ISP coding mode will be tested on it, using the original procedure.

In short, a decision step is attached to the original ISP algorithm. Before testing ISP, the *TC* value of current CU is calculated. If *TC* < 20, then the ISP flag is set to 0; no ISP coding mode is tested for this CU. If *TC* > 20, the algorithm will continue to use the original ISP coding mode.

4 Experimental results and discussion

In order to verify the performance of the proposed algorithm, the algorithm is implemented on the VVC reference software (VTM8.0). Since the proposed algorithm is used to accelerate intra coding, All Intra (AI) configuration is adopted and the QP value is set to be 22, 27, 32, and 37, respectively [22]. In our work, standard test sequences of HEVC are selected, which involve different scenes and different resolutions.

Bjøntegaard Delta rate (BD-rate) with piece-wise cubic interpolation method is used to assess the coding performance of the proposed algorithm. Time saving (*TS*) is used to measure the reduction of computational complexity and is defined as:

$$TS = \frac{1}{4} \sum_{QP_i \in \{22,27,32,37\}} \frac{Time_{VTM}(QP_i) - Time_{Pro}(QP_i)}{Time_{VTM}(QP_i)} \tag{4}$$

The performance comparison between the VTM8.0 and the proposed algorithm are shown in Table 1. It presents the average BD-rate of five classes of video sequences. In the AI configuration, the proposed algorithm can achieve about 7% coding time saving with negligible loss of coding efficiency. It can be also observed that a consistent gain is obtained over all sequences. The largest gain comes from the sequence "Basketball-Drill", with up to 11% time saving. Since "BasketballDrill" contains a large number of flat or constant blocks that are considered as simple texture, it can be compressed more efficiently by the proposed algorithm in this paper. Sequences with similar characteristics such as "BasketballDrive" and "Johnny" all show remarkable time saving. The proposed method does not perform well for sequences containing rich texture, such as "PartyScene" and "BQTerrace".

As one of the main optimization goals of VVC is to improve the coding efficiency of ultra-high definition/high definition (UHD/HD) videos, it can be noted that the
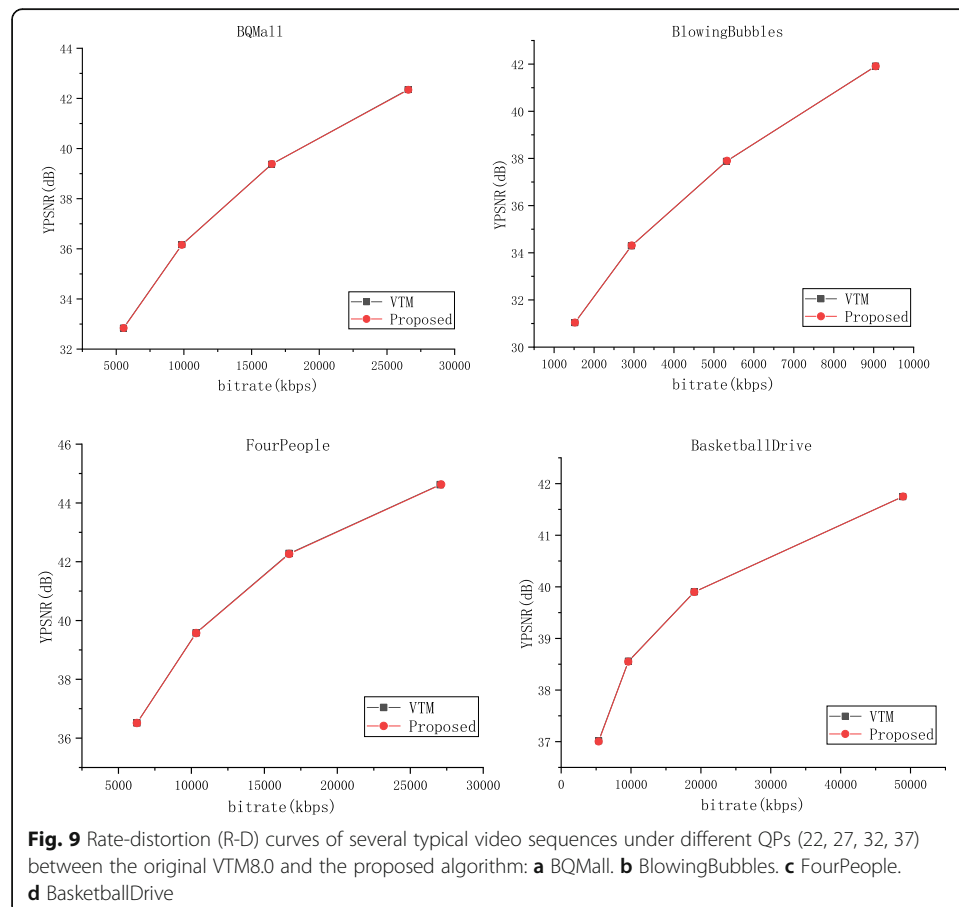
Table 1 Performance comparison between the original VTM8.0 and the proposed algorithms

Class	Sequence	BD-rate			TS
		Y	U	V	
Class A	Traffic	0.12%	− 0.24%	0.04%	7%
	PeopleOnStreet	0.04%	0.03%	− 0.11%	7%
	Nebuta	− 0.01%	− 0.04%	− 0.03%	6%
	SteamLocomotive	0.04%	− 0.25%	− 0.23%	6%
Class B	Kimono	0.05%	− 0.10%	− 0.07%	6%
	ParkScene	0.06%	− 0.05%	0.35%	6%
	Cactus	0.14%	0.07%	− 0.08%	6%
	BasketballDrive	0.24%	− 0.41%	1.10%	9%
	BQTerrace	0.01%	− 0.33%	− 0.43%	4%
Class C	BasketballDrill	0.30%	0.45%	0.47%	11%
	BQMall	0.10%	− 0.24%	0.37%	6%
	PartyScene	0.01%	0.46%	− 0.33%	4%
	RaceHorsesC	− 0.07%	− 0.02%	0.09%	6%
Class D	BasketballPass	0.04%	1.47%	0.65%	8%
	BQSquare	0.18%	0.27%	0.61%	8%
	BlowingBubbles	0.00%	0.43%	0.65%	6%
	RaceHorses	0.12%	− 0.12%	− 0.53%	5%
Class E	FourPeople	0.17%	− 0.51%	− 0.71%	7%
	Johnny	0.22%	− 0.33%	− 0.26%	8%
	KristenAndSara	0.10%	0.22%	− 0.40%	8%
	Average	0.09%	0.04%	0.06%	7%

proposed algorithm generally performs better on UHD/HD sequences than other sequences. This is due to that the proposed fast ISP coding mode optimization algorithm based on texture complexity has better performance on simple texture blocks and simple texture blocks often take a larger portion in UHD/HD sequences. It is a prominent advantage to favor high resolution sequences in future applications. Figure 9 illustrates the rate-distortion (RD) curves comparison results of our proposed algorithm compared with VTM8.0 for the BQMall (832 × 480), BlowingBubbles (832 × 480), FourPeople (1280 × 720), and BasketballDrive (1920 × 1080) sequences respectively, which includes the best case (BlowingBubbles) and the worst case (BasketballDrill) in terms with the RD performance. The results show that the proposed algorithm is superior to VTM8.0 for most sequences, either in low bitrate or in high bitrate configuration. Even in the worst case, the proposed algorithm and the original VTM reference encoder can obtain very similar image quality under different QPs.

5 Conclusion

A fast ISP coding mode optimization algorithm based on CU texture complexity is proposed in this paper to reduce the computation complexity of ISP. Firstly, it is observed in this paper that the CUs with simple texture are not suitable for ISP coding mode by comparing the RD-cost of CU with different texture complexity in ISP coding mode

Fig. 9 Rate-distortion (R-D) curves of several typical video sequences under different QPs (22, 27, 32, 37) between the original VTM8.0 and the proposed algorithm: **a** BQMall. **b** BlowingBubbles. **c** FourPeople. **d** BasketballDrive

and No-ISP coding mode. Secondly, a metric to measure the CU texture complexity is proposed, which divide CUs into simple texture and complex texture. Finally, the ISP coding mode will be skipped for CUs with simple texture. The proposed method is tested on VVC reference software VTM8.0; the experimental results show that proposed algorithm can achieve about 7% coding time saving with negligible loss of coding efficiency.

Abbreviations
VVC: Versatile Video Coding; HEVC: High Efficiency Video Coding; CU: Coding unit; VTM: VVC test model; RD-cost: Rate distortion cost; No-ISP: Not using ISP coding mode; AI: All Intra; MDIS: Mode-dependent intra smoothing; CCLM: Cross-component linear model; PDPC: Position-dependent intra prediction combination; MRL: Multiple reference line intra prediction; MIP: Matrix-weighted intra prediction; LIP: Line-based intra prediction; MAD: Mean absolute deviation; TC: Texture complexity; TS: Time saving

Authors' contributions
ZL proposed the framework of this work, and MD carried out the whole experiments and drafted the manuscript. MZ, HZ, and RW offered useful suggestions and helped to modify the manuscript. All authors read and approved the final manuscript.

Authors' information
ZL: Doctor of Engineering, master instructor. He received the B.S. degree in electronic information technology and the Ph.D. in signal and information processing from Beijing Jiaotong University, China, in 2001 and 2011 respectively. Currently, he is an associate professor in North China University of Technology. His major research interests include the video codec, pattern recognition, and self-organizing network.
MJD: Studying master of North China University of Technology. Her major research is VVC.
XHG: She received the Ph.D. degree from Tianjin University, Tianjin, China, in 1999. She is now an associate professor with department of electronic information engineering, North China University of Technology, Beijing, China.
MMZ: Doctor of Engineering, professor, master instructor, master of Communication and Information Systems. His major research interests include the video codec, embedded systems, image processing, and pattern recognition. He has authored or co-authored more than 40 refereed technical papers in international journals and conferences in the field of video coding, image processing, and pattern recognition. He holds 21 national patents and 2 monographs in the areas of image/video coding and communications.
RYW: Doctor candidate of Harbin Institute of Technology (Weihai), Weihai, China.

Availability of data and materials
The conclusion and comparison data of this article are included within the article.

Declarations

Competing interests
The authors declare that they have no competing interests.

Author details
[1]North China University of Technology, Beijing 100144, China. [2]Beijing Polytechnic College, Beijing 100144, China. [3]Harbin Institute of Technology (Weihai), Weihai 264200, China.

References
1. J. Chen, Y. Ye, S. Kim, *Algorithm description for Versatile Video Coding and Test Model 8 (VTM 8), Document JVET-Q1002 (JVET (Joint Video Experts Team), Brussels, 2020)*
2. S. De-Luxán-Hernández, H. Schwarz, D. Marpe, T. Wiegand, in *IEEE International Symposium on Multimedia (ISM)*. Fast line-based intra prediction for video coding (2018), pp. 135–138
3. S. De-Luxán-Hernández, V. George, J. Ma, T. Nguyen, H. Schwarz, D. Marpe, T. Wiegand, in *2019 IEEE International Conference on Image Processing (ICIP)*. An intra subpartition coding mode for VVC (2019)
4. S. De-Luxán-Hernández, H. Schwarz, D. Marpe, T. Wiegand, in *25th IEEE International Conference on Image Processing (ICIP)*. Line-based intra prediction for next-generation video coding (2018), pp. 221–225
5. S. De Luxán Hernández, H. Schwarz, D. Marpe, T. Wiegand, in *Document JVET-K0049 of JVET, Ljubljana*. CE3: line-based intra coding mode (Tests 1.4.1, 1.4.2 and 1.4.3) (2018)

6. S. De Luxán Hernández, H. Schwarz, D. Marpe, T. Wiegand, in *Document JVET-L0076 of JVET, Macao*. CE3: Line-based intra coding mode (Tests 2.1.1 and 2.1.2) (2018)
7. S. De-Luxán-Hernández, V. George, J. Ma, T. Nguyen, H. Schwarz, D. Marpe, T. Wiegand, in *Document JVET-M0102 of JVET, Marrakech, MA*. CE3: intra sub-partitions coding mode (Tests 1.1.1 and 1.1.2) (2019)
8. C. Yan, B. Gong, Y. Wei, Y. Gao, in *IEEE Transactions on Pattern Analysis and Machine Intelligence*. Deep multi-view enhancement hashing for image retrieval (2020)
9. C. Yan, B. Shao, H. Zhao, R. Ning, Y. Zhang, F. Xu, 3D room layout estimation from a single RGB image. IEEE Trans. Multimed. **22**(11), 3014–3024 (2020)
10. C. Yan, Z. Li, Y. Zhang, Y. Liu, X. Ji, Y. Zhang, Depth image denoising using nuclear norm and learning graph model. ACM Trans. Multim. Comput. Commun. Appl. **16**(4), 1–7 (2020)
11. C. Yan, Y. Hao, L. Liang, J. Yin, A. Liu, Z. Mao, Z. Chen, X. Gao, Task-Adaptive Attention for Image Captioning. IEEE Trans. Circuits Syst. VideoTechnol. (2021) https://doi.org/10.1109/TCSVT.2021.3067449.
12. L. Shen, Z. Zhang, Z. Liu, Effective CU size decision for HEVC intracoding. IEEE Trans. Image Process. **23**(10), 4232–4241 (2014)
13. Z. Peng, C. Huang, F. Chen, G. Jiang, X. Cui, M. Yu, Multiple classifier-based fast coding unit partition for intra coding in future video coding. Signal Process. Image Commun. **78**, 171–179 (2019)
14. X. Liu, Y. Li, D. Liu, et al., An adaptive CU size decision algorithm for HEVC intra prediction based on complexity classification using machine learning. IEEE Trans. Circuits Syst. Video Technol. **29**(1), 144–155 (2019)
15. J. Hou, D. Li, Z. Li, X. Jiang, in *2013 International Conference on Mechatronic Sciences, Electric Engineering and Computer (MEC)*. Fast CU size decision based on texture complexity for HEVC intra coding (2013)
16. W. Zhu, Y. Yi, H. Zhang, Fast mode decision algorithm for HEVC intra coding based on texture partition and direction. J. Real Time Image Process. **17**(2), 275–292 (2020)
17. X. Sun, X. Chen, Y. Xu, Fast CU size and prediction mode decision algorithm for HEVC based on direction variance. J. Real Time Image Process. **16**(5), 1731–1744 (2019)
18. M. Mohamed, B. Noureddine, B. Nejmeddine, Homogeneity-based fast CU partitioning algorithm for HEVC intra coding. Eng. Sci. Technol. **22**(3), 706–714 (2019)
19. F. Chen, D. Jin, Z. Peng, Fast intra coding algorithm for HEVC based on depth range prediction and mode reduction. Multimed. Tools Appl. **77**(8), 28375–28394 (2018)
20. Y. Lu, X. Huang, H. Liu, Hierarchical classification for complexity reduction in HEVC inter coding. IEEE Access **8**, 41690–41704 (2020)
21. T. Zhang, M. Sun, D. Zhao, W. Gao, Fast intra-mode and CU size decision for HEVC. IEEE Trans. Circuits Syst. Video Technol. **27**(8), 1714–1726 (2017)
22. F. Bossen, J. Boyce, X. Li, V. Seregin, K. Sühring (editors). JVET common test conditions and software reference configurations for SDR video, Document JVET-M1010, Marrakech, MA, 2019.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.