Analysis of VVC Intra Prediction Block Partitioning Structure

Mário Saldanha[1], Gustavo Sanchez[2], César Marcon[3], Luciano Agostini[1]
[1]Federal University of Pelotas – Graduate Program in Computing (PPGC) – Pelotas, Brazil
[2]IF Farroupilha – Alegrete, Brazil
[3]Pontificial Catholic University of Rio Grande do Sul – Porto Alegre, Brazil
{mrdfsaldanha, agostini}@inf.ufpel.edu.br, gustavo.sanchez@iffarroupilha.edu.br, cesar.marcon@pucrs.br

2021 International Conference on Visual Communications and Image Processing (VCIP) | 978-1-7281-8551-4/21/$31.00 ©2021 IEEE | DOI: 10.1109/VCIP53242.2021.9675347

***Abstract*—This paper presents an encoding time and encoding efficiency analysis of the Quadtree with nested Multi-type Tree (QTMT) structure in the Versatile Video Coding (VVC) intra-frame prediction. The QTMT structure enables VVC to improve the compression performance compared to its predecessor standard at the cost of a higher encoding complexity. The intra-frame prediction time raised about 26 times compared to the HEVC reference software, and most of this time is related to the new block partitioning structure. Thus, this paper provides a detailed description of the VVC block partitioning structure and an in-depth analysis of the QTMT structure regarding coding time and coding efficiency. Based on the presented analyses, this paper can guide outcoming works focusing on the block partitioning of the VVC intra-frame prediction.**

***Keywords*— Computational complexity analysis, Compression efficiency analysis, Block partitioning, Versatile Video Coding.**

I. INTRODUCTION

Following a long line of success in video coding standardization, a collaboration between ISO Moving Picture Experts Group (MPEG) and ITU-T Video Coding Experts Group (VCEG) established the Joint Video Experts Team (JVET) to develop the Versatile Video Coding (VVC) standard [1]. VVC was designed to obtain a significantly higher coding efficiency than the High-Efficiency Video Coding (HEVC) standard [2] and to have a high versatility for effective use with different types of video content and applications.

VVC improves the compression performance compared to its predecessor video coding standard, inserting many new techniques. This improvement includes new approaches for the block partitioning such as support for larger block sizes, effective block partitioning structure through the Quadtree with nested Multi-type Tree (QTMT) [3] that allows more flexibility for Coding Unit (CU) partition shapes and enables the luminance and chrominance to have a separate coding tree structure through a dual-tree coding structure [4]. The novelties of the intra-frame prediction, which is the focus of this paper, are: (i) increased the directional modes to 65 modes [4], (ii) enabled the use of more reference lines for the prediction through the Multiple Reference Line (MRL) [5], (iii) allowed a higher correlation of reference samples using Intra Subpartition (ISP) [6], and (iv) performed the prediction with the learning-based Matrix weighted Intra Prediction (MIP) tool [7]. Besides, several other modules also were improved, such as the inter-frame prediction with Affine Motion Compensation (AMC) [8], the transform with Multiple Transform Selection (MTS), [9] and the Low-Frequency non-Separable Transform (LFNST) [10].

All these tools contribute to enhance the overall VVC compression efficiency. Bossen et al. [11] reported that these encoding tools implemented in the VVC Test Model (VTM) [12] 10.0 could provide about 25.1% of coding efficiency gain over the latest HEVC Test Model [13] 16.22, considering All-Intra (AI) encoder configuration. However, while these coding tools enable a significant compression enhancement, the required computational effort of the VVC encoder rises about 26 times compared to HM under AI configuration.

This high encoding effort boosted the development of some works focusing on reducing the VVC encoding time. Several of these works [14]-[21] focused on reducing the computational cost of the QTMT partitioning structure, especially for intra-frame prediction. These solutions include fast CU decisions based on statistical analysis [14]-[17] and machine learning techniques [18]-[21]; all these solutions focus only on luminance blocks. Lei et al. [14] developed a fast solution to avoid unnecessary block partition evaluations, where a subset of directional intra-frame prediction modes is evaluated to estimate the horizontal and vertical partitioning cost of the current block. Saldanha et al. [15] proposed a fast block partitioning decision scheme for deciding the direction of binary and ternary partitions based on the intra-frame prediction mode selected for the current block and the variance of subpartitions. Fan et al. [16] developed a solution based on the current block variance, subpartition variance, and Sobel filter. Yang et al. [17] proposed a low complexity QTMT decision based on a machine learning technique; the decision trees are used as classifiers for deciding the block partitioning structure. Fu et al. [18] presented a fast CU partitioning algorithm using a classifier based on the Bayesian decision rule; the information derived from the current CU and horizontal binary splitting is used as a model input feature. The works of Tissier et al. [19], Zhao et al. [20], and Li et al. [21] proposed the use of a Convolutional Neural Network (CNN) to find the best block partitioning and reduce the encoding time.

The works [22]-[25] analyzed several aspects of VVC, including performance comparisons with prior standards, memory assessment, and encoding and decoding time evaluations. The works [22] and [23] performed objective and subjective evaluations of the VVC performance compared to other video coding standards, such as HEVC and AOMedia Video 1 (AV1). Cerveira et al. [24] presented a memory evaluation of VVC, considering an overall memory profiling and an inter-frame prediction specific analysis. Pakdaman et al. [25] analyzed the VVC encoding and decoding time and memory requirements using a profiling tool. The timing results were obtained for the encoding modules of inter- and intra-frame prediction, transform, quantization, entropy coding, and loop filters. Saldanha et al. [26] presented an analysis of the VVC intra-frame prediction focusing only on encoding time and prediction tools.

Although these works presented several VVC profiling details, none of them presented a detailed analysis of the block partitioning structure to evaluate the possibilities of encoding time reduction and the impact on the coding efficiency, which are crucial to motivate and justify the proposed solutions. To fulfill this gap, this paper details the QTMT partitioning structure and analyzes the encoding time and coding efficiency of each block partition type used in the intra-frame prediction. Besides, this paper supports outcoming works to develop effective solutions targeting different aspects, including complexity reduction, complexity control, and real-time hardware design for VVC intra-frame prediction.

II. VVC Block Partitioning Structure

VVC follows a block-based hybrid video coding approach, a classic concept of all major video coding standards such as HEVC and AV1. This approach splits each video sequence frame into blocks to be processed by intra- and inter-frame prediction, forward/inverse transform and quantization, and entropy coding.

The block partitioning of modern codecs has a crucial role in compression efficiency due to the coding structure. Thus, several block partitioning structures were proposed and enhanced during the VVC standardization, resulting in a structure with larger block sizes than HEVC, which provides an efficient compression rate mainly for ultra-high video resolutions, and with flexible partition types capable of adapting to different texture characteristics.

VVC divides each input frame into Coding Tree Units (CTUs) with up to 128×128 luminance samples and up to 64×64 chrominance samples; each CTU can be recursively partitioned into smaller blocks called CU. To split these CTUs, VVC inherits the concept of Quadtree (QT) employed in HEVC and inserts the Multi-type Tree (MTT) partitioning structure, allowing square- and rectangular-shaped CU sizes through QT, Binary Tree (BT), and Ternary Tree (TT) [4]. These combined structures (QT+MTT) are called QTMT and enable six partition types, shown in Fig. 1. When a CU is defined as no split, the encoding process is performed with the current CU size; otherwise, CU is split using QT, BT, or TT.

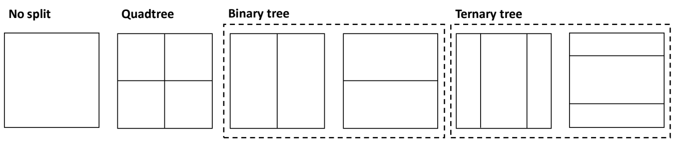


Fig. 1 – Partition types available in the QTMT partitioning structure.

QT splits a CU into four equal-sized square CUs. BT can be defined as Binary Tree Horizontal (BTH) or Binary Tree Vertical (BTV); BTH and BTV split a CU into two symmetric CUs with horizontal and vertical directions, respectively. TT also can be applied in two directions, including Ternary Tree Horizontal (TTH) and Ternary Tree Vertical (TTV). Still, TT divides a CU into three smaller CUs, composed of a central one and two sides of CUs, having 50% and 25% of the original CU size, respectively.

These block partitions can be applied for both luminance and chrominance samples. On the one hand, in P- and B-slices, which allow intra- and inter-frame prediction, the luminance and chrominance CTUs share the same coding structure. On the other hand, for I-slices (allow only intra-frame prediction), the luminance and chrominance can have a separate coding tree, known as Dual-Tree (DT) or Chroma Separate Tree (CST) [4].

The maximum and minimum luminance CU sizes processed in the intra-frame prediction are 64×64 (maximum transform block size) and 4×4 samples, respectively. Regarding chrominance, the maximum and minimum sizes are 32×32 and 16 samples (8×2 or 4×4), respectively.

Fig. 2 exemplifies the QTMT partitioning applied for a 128×128 luminance CTU split into several CU sizes with different QT and MTT levels. Each colored line represents a block partition type; black lines denote the QT splitting, green and orange lines indicate the BTH and BTV splitting, respectively, and blue and red lines represent TTH and TTV splitting, respectively.

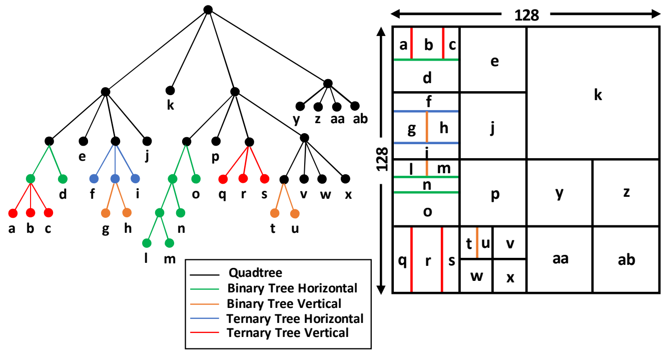


Fig. 2 – Illustration of the QTMT block partitioning structure.

Since the intra-frame prediction is performed only for 64×64 or smaller blocks, the 128×128 luminance CTU is partitioned by QT splitting, resulting in four 64×64 CUs. Then, the QTMT depth starts from level one for the intra-frame prediction. From that, each CU can be recursively partitioned using the QT structure. The QT splitting follows the same process used in HEVC. Additionally, each QT leaf node can be further partitioned in an MTT structure, which also can divide the CU recursively through BT/TT partitions. It is important to mention that the MTT structure can be applied only to QT leaf nodes, i.e., once a CU is split using MTT, no further QT splitting is allowed.

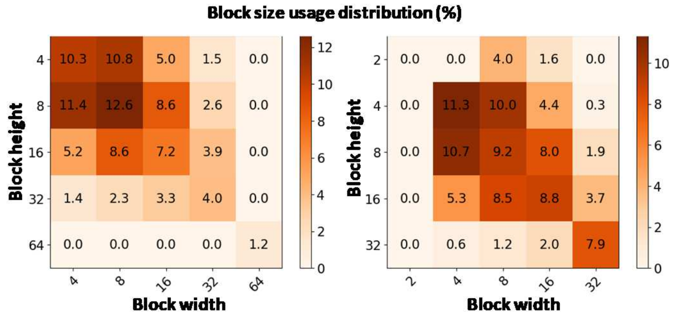
III. Performance Analysis of the QTMT Structure

As previously discussed, the high computational cost of the QTMT structure motivated several works to develop solutions for VVC block partitioning time reduction. To guide future works, this section aims to provide a detailed analysis of the new QTMT structure in the VVC intra-frame prediction. This analysis includes block size usage and coding time distribution, encoding time distribution between luminance and chrominance, and performance evaluation regarding both coding efficiency and encoding time reduction. The following analyses are performed with VTM 10.0 according to JVET Common Test Conditions (CTC) [27] under All-Intra configuration. The CTC specifies six classes of video sequences, where classes A1 and A2 are Ultra-High Definition (UHD) video sequences (3840×2160), and class B encloses Full-High Definition (FHD) video sequences (1920×1080). Classes C and D include lower video resolutions with 832×480 and 416×240 pixels, respectively. Finally, Class E covers 1280×720 video sequences. Each video sequence is encoded with four Quantization Parameter (QP) values: 22, 27, 32, and 37. These experiments were performed into a server with the Ubuntu 20.04 operating system, AMD Opteron™ Processor 6376, and 128 GB DDR3 memory.

Fig. 3 (a) and (b) demonstrates the block size distribution for luminance and chrominance, respectively, through a heat map, indicating the usage percentage of each available block size. The results refer to the average among all video sequences and the four QP values specified in CTC.

Since the MTT structure is not applied for 64×64 luminance CUs, 64×N and N×64 blocks have no occurrences. For chrominance blocks, VVC inserts some restrictions for blocks with width or height equals to 2. In this case, VVC allows only 8×2, 16×2, and 32×2 block sizes; however, the usage of 32×2 is smaller than 0.015%.

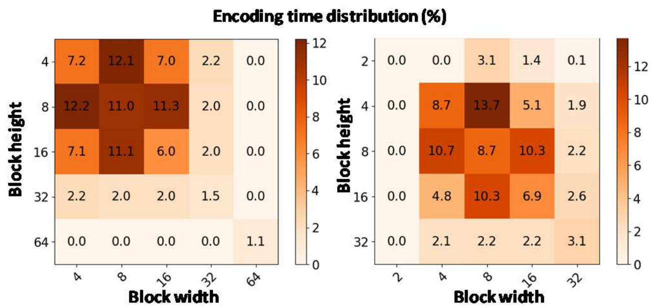
This analysis shows that the QT splitting results square CU sizes with high usage, representing 35.3% and 37.2% of the total usage distribution for luminance and chrominance, respectively. However, the rectangular-shaped CU sizes represent more than 50% of the total usage distribution for both luminance and chrominance, demonstrating that the MTT structure also increases the coding efficiency significantly.



(a) (b)

Fig. 3 – Block size usage distribution for (a) luminance and (b) chrominance.

Fig. 4 displays the encoding time distribution for luminance (Fig. 4(a)) and chrominance (Fig. 4 (b)) block sizes. One can notice that most of the encoding time is spent on smaller blocks, where about 80% of the encoding time is concentrated in 16×N, N×16, or smaller blocks, for both luminance and chrominance. Besides, the rectangular CU sizes (obtained with MTT splitting) represent about 74% and 68% of the total encoding time for luminance and chrominance, respectively.



(a) (b)

Fig. 4 – Block size time distribution for (a) luminance and (b) chrominance.

Fig. 5 presents the CU size distribution (luminance) for the first frame of the BasketballPass video sequence encoded with VTM 10.0 using All-Intra configuration and QP 37, allowing the usage case analysis of these partition types. One can notice that the block partition structure is strongly correlated with the image details. Fig. 5(a) and (b) illustrate this effect detaching a smooth region encoded with larger block sizes and a detailed region encoded with smaller block sizes, respectively. In the detailed region, several MTT structure levels and different directions of BT and TT partitions are employed according to the texture characteristics. In contrast, few QT and MTT splitting levels are required to provide effective compression in the smooth region.

Compared to the block partitioning structure of HEVC, QTMT enables high flexibility to represent the block sizes and shapes. Thus, these block partition types can adapt to a wide variety of video characteristics, resulting in better coding efficiency. However, this high flexibility also results in more block sizes, increasing the encoding time significantly.

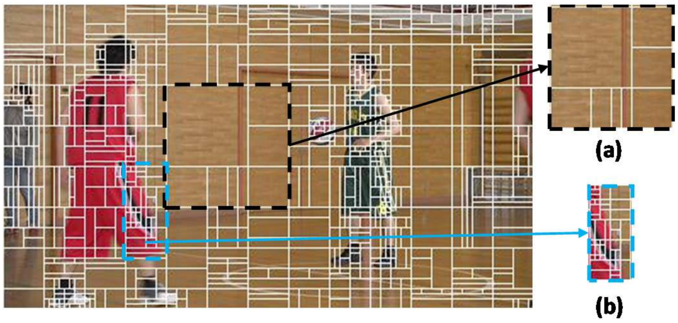


Fig. 5 – CU size distribution for the BasketballPass video sequence highlighting (a) smooth and (b) detailed regions.

Since VVC inserts the DT (or CST) encoding tool, it is interesting to analyze the encoding time distribution and understand the time reduction opportunities for each color component. Fig. 6 presents the encoding time distribution between luminance and chrominance components according to each QP.

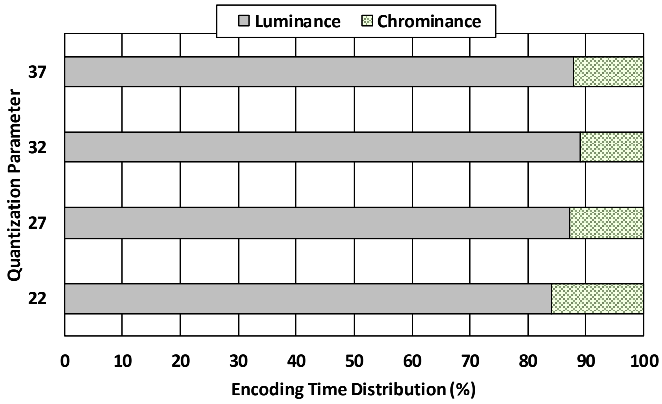


Fig. 6 – Encoding time distribution between luminance and chrominance components according to the QP.

One can notice that slight variations in the encoding time distribution occur when QP varies. The luminance encoding represents the highest encoding time for all evaluated cases, obtaining up to 89.04% for QP 32. The highest chrominance encoding time is noticed for QP 22, representing 15.97% of the total. On average, luminance represents about 87% of the total encoding time.

The subsequent analyses evaluate the Bjontegaard Delta bitrate (BDBR) [28] impact on the luminance channel and the Encoding Time Reduction (ETR) of limiting and removing some partition types of the QTMT structure for VVC intra-frame prediction. We applied these modifications for both QTMT structures of luminance and chrominance.

Table I presents the BDBR impact and ETR when limiting the maximum luminance and chrominance QTMT depth levels to 2, 3, and 4. Considering that QTMT starts with a depth level equals to one for the intra-frame prediction, QTMT depth 1 was not evaluated because only 64×64 luminance and 32×32 chrominance CUs would be available, reducing the coding efficiency drastically.

One can notice that limiting the QTMT depth can provide expressive encoding time reduction. As expected, the fewer possibilities of block partitioning to evaluate, the greater the ETR gain and BDBR loss, where QTMT depth 2 obtained the highest ETR and BDBR loss of 92.82% and 25.40%, respectively. In contrast, more possibilities of block partitions are evaluated for QTMT depth 3 and 4, resulting in less ETR gains and BDBR losses.

In these evaluations, the highest video resolutions (classes

A1, A2, and B) presented a lower BDBR impact, while the highest BDBR losses were obtained in class E for QTMT depth 2 and class D for QTMT depth 3 and 4. Regarding encoding time, class A1 attained the lowest ETR gain for all QTMT depths evaluated. For QTMT depth 2 and 3 the highest ETR gains were obtained with class C, whereas class D presented the highest ETR gain for QTMT depth 4.

Table I – Coding efficiency and encoding time reduction when limiting the QTMT depth.

Class	QTMT Depth 2		QTMT Depth 3		QTMT Depth 4	
	BDBR (%)	ETR (%)	BDBR (%)	ETR (%)	BDBR (%)	ETR (%)
A1	11.43	89.73	3.69	77.29	1.02	52.47
A2	14.79	93.48	5.32	83.89	1.62	60.82
B	21.00	93.97	8.85	85.22	2.64	64.73
C	34.68	94.88	16.79	87.27	5.53	71.68
D	33.42	92.90	17.17	86.17	6.05	71.96
E	37.10	91.99	14.96	83.21	4.51	63.21
Avg	**25.40**	**92.82**	**11.3**	**83.97**	**3.56**	**64.14**

Table II shows BDBR and ETR when removing BT (BT less) and TT (TT less) partitions. “BT+TT less” refers to remove both partitions, allowing only QT splitting (i.e., the same partitioning structure used in HEVC). This analysis demonstrates that BT and TT partitions are responsible for providing a high compression performance at the cost of a high encoding time; removing both partitions results in a 93.59% of ETR gain and 26.11% of BDBR increase. From this analysis, one can conclude that BT partition provides a higher compression performance than TT partition at the cost of a higher encoding time.

Table II – Coding efficiency and encoding time reduction when removing BT and/or TT partitions.

Class	BT less		TT less		BT+TT less	
	BDBR (%)	ETR (%)	BDBR (%)	ETR (%)	BDBR (%)	ETR (%)
A1	4.40	72.93	0.67	42.10	12.39	90.90
A2	4.91	78.71	0.97	48.61	16.04	94.47
B	5.84	77.58	1.11	48.38	22.10	94.71
C	8.46	79.73	1.55	51.92	36.17	95.50
D	6.68	77.49	1.32	51.38	30.72	93.10
E	8.56	76.00	1.76	48.12	39.25	92.86
Avg	**6.57**	**77.07**	**1.23**	**48.42**	**26.11**	**93.59**

Like the previous analysis, the highest video resolutions (A1, A2, and B) presented a lower BDBR impact, however, the highest BDBR losses were noticed in class E for all cases in Table II. Considering the ETR, classes A1 and C presented the lowest and the highest ETR gains for all cases evaluated, respectively. On the one hand, when removing TT partitions class D presented a higher ETR gain than class B. On the other hand, disable BT partitions and both partitions (BT+TT less) allow class B to obtain a higher ETR gain than class D.

Table III presents BDBR and ETR when removing horizontal (BTH and TTH) and vertical (BTV and TTV) partitions. This analysis displays that both horizontal and vertical partitions provide similar results of ETR and BDBR increase. The horizontal partition removal provides 79.95% of ETR gain with a 5.32% increase in BDBR; disabling the vertical partitions reduces about 79.42% of the encoding time with a 5.50% of BDBR increase.

The lowest BDBR impacts here also were noticed in the highest video resolutions, whereas the highest BDBR impact was obtained with class E followed by class C and D. The lowest and the highest ETR gains were attained in classes A1 and C for the cases evaluated in Table III, respectively.

Table III – Coding efficiency and encoding time reduction when removing the horizontal or vertical partitions.

Class	Horizontal less		Vertical less	
	BDBR (%)	ETR (%)	BDBR (%)	ETR (%)
A1	3.75	74.73	3.33	74.16
A2	3.67	80.42	4.04	79.97
B	5.63	80.99	4.34	79.95
C	6.54	83.21	6.45	82.55
D	5.74	81.94	6.17	80.90
E	6.61	78.42	8.64	78.99
Avg	**5.32**	**79.95**	**5.50**	**79.42**

According to our experiments, the ETR gains decrease as QP increases for all presented evaluations. This happens because low QP values retain more image details, producing more heterogeneous regions, which are encoded with smaller block sizes and different block shapes. In contrast, high QP values attenuate the image details, producing more homogenous regions that are better encoded with larger block sizes. Besides, the ETR gain significantly reduces for 4K video sequences compared to other video resolutions when considering higher QP values since this video resolution is already encoded with larger blocks than the others, especially for high QP values.

IV. Discussion and Conclusions

This work presented an encoding time and compression performance assessment for the new block partitioning structure (QTMT) for the VVC intra-frame prediction. The VVC intra coding time increased expressively compared to its predecessor standard, and most of that encoding time is related to the block partitioning structure.

This work presented the block size usage distribution to demonstrate the efficiency of the new VVC partition types, which are used more than 50% for both luminance and chrominance components. Moreover, although VVC allows the dual-tree structure, the luminance encoding consumes the highest encoding time. However, real-time processing cannot neglect 13% of encoding time associated with the chrominance encoding. The BDBR and encoding time reduction analyses demonstrated that effective solutions for early terminating the QTMT depth evaluation could provide significant encoding time reduction; however, inaccurate decisions tend to generate significant drops in the coding efficiency. Besides, solutions capable of deciding between BT and TT partitions or horizontal and vertical direction also can provide interesting encoding time reductions, and for some cases with less encoding efficiency loss compared to limiting the QTMT depth.

Considering the presented results, one can conclude that heuristics, based on statistical analysis and/or machine learning, able to adaptively explore a combination of the different encoder reduction methods considering the video features and encoder behavior are promising approaches to provide high encoding time reduction while maintaining the coding efficiency. Then, the presented analysis is important to understand the VVC block partitioning structure and to support outcoming solutions focusing on complexity reduction, complexity control, and real-time hardware design for VVC intra-frame prediction.

Acknowledgment

The authors thank FAPERGS, CNPq and CAPES (Finance Code 001) Brazilian research support agencies that financed this investigation.

REFERENCES

[1] Versatile Video Coding, Recommendation ITU-T H.266 and ISO/IEC 23090-3 (VVC), ITU-T and ISO/IEC JTC 1, Jul. 2020.

[2] G. J. Sullivan et al., "Overview of the High Efficiency Video Coding (HEVC) Standard," *IEEE Trans. Circuits Syst. Video Technol. (TCSVT)*, vol. 22, no. 12, pp. 1649-1668, Dec. 2012.

[3] Y. Huang et al., "A VVC Proposal with Quaternary Tree Plus Binary-Ternary Tree Coding Block Structure and Advanced Coding Techniques," *IEEE Trans. Circuits Syst. Video Technol. (TCSVT)*, vol. 30, no. 5, pp. 1311-1325, May 2020.

[4] J. Chen, Y. Ye, S. Kim. "Algorithm description for Versatile Video Coding and Test Model 10 (VTM 10)," JVET 19th Meeting, JVET-S2002, Teleconference, Jul. 2020.

[5] B. Bross et al. "CE3: Multiple reference line intra prediction (Test 1.1.1, 1.1.2, 1.1.3 and 1.1.4)," JVET 12th Meeting, JVET-L0283, Macao, Oct. 2018.

[6] S. De-Luxán-Hernández et al., "An Intra Subpartition Coding Mode for VVC," *IEEE Int. Conf. Image Process. (ICIP)*, Taipei, Taiwan, 2019, pp. 1203-1207.

[7] M. Schäfer et al., "An Affine-Linear Intra Prediction with Complexity Constraints," *IEEE Int. Conf. Image Process. (ICIP)*, Taipei, Taiwan, 2019, pp. 1089-1093.

[8] K. Zhang et al., "An Improved Framework of Affine Motion Compensation in Video Coding," *IEEE Trans. Image Process. (TIP)*, vol. 28, no. 3, pp. 1456-1469, Mar. 2019.

[9] X. Zhao et al., "Enhanced Multiple Transform for Video Coding," *Data Compression Conf. (DCC)*, Snowbird, UT, 2016, pp. 73-82.

[10] M. Koo et al., "Low Frequency Non-Separable Transform (LFNST)," *IEEE Picture Coding Symp. (PCS)*, Ningbo, China, 2019, pp. 1-5.

[11] F. Bossen et al., “AHG report: Test model software development (AHG3),” JVET 20th Meeting, JVET-T0003, Teleconference, Oct. 2020.

[12] VVC Test Model (VTM). Available at: https://vcgit.hhi.fraunhofer.de/jvet/VVCSoftware_VTM, access in Feb. 2021.

[13] HEVC Test Model (HM). Available at: https://vcgit.hhi.fraunhofer.de/jct-vc/HM, access in Feb. 2021.

[14] M. Lei et al., “Look-Ahead Prediction Based Coding Unit Size Pruning for VVC Intra Coding,” *IEEE Int. Conf. Image Process. (ICIP)*, 2019, pp. 4120-4124.

[15] M. Saldanha et al., "Fast Partitioning Decision Scheme for Versatile Video Coding Intra-Frame Prediction," *2020 IEEE Int. Symp. Circuits Syst. (ISCAS)*, Seville, Spain, 2020, pp. 1-5.

[16] Y. Fan et al., "A Fast QTMT Partition Decision Strategy for VVC Intra Prediction," in *IEEE Access*, vol. 8, pp. 107900-107911, 2020.

[17] H. Yang et al., "Low-Complexity CTU Partition Structure Decision and Fast Intra Mode Decision for Versatile Video Coding," *IEEE Trans. Circuits Syst. Video Technol. (TCSVT)*, vol. 30, no. 6, pp. 1668-1682, June 2020.

[18] T. Fu et al., "Fast CU Partitioning Algorithm for H.266/VVC Intra-Frame Coding," *IEEE Int. Conf. Multimedia Expo (ICME)*, Shanghai, China, 2019, pp. 55-60.

[19] A. Tissier et al., "CNN Oriented Complexity Reduction Of VVC Intra Encoder," *2020 IEEE Int. Conf. Image Process. (ICIP)*, Abu Dhabi, United Arab Emirates, 2020, pp. 3139-3143.

[20] J. Zhao, Y. Wang, Q. Zhang, "Adaptive CU Split Decision Based on Deep Learning and Multifeature Fusion for H. 266/VVC," *Scientific Programming*, v. 2020, 2020.

[21] T. Li, M. Xu, R. Tang, "DeepQTMT: A Deep Learning Approach for Fast QTMT-based CU Partition of Intra-mode VVC," *arXiv preprint arXiv:2006.13125*, 2020.

[22] P. Topiwala, M. Krishnan, W. Dai, "Performance comparison of VVC, AV1, and HEVC on 8-bit and 10-bit content," *SPIE Appl. Digit. Image Process. XLI*, vol. 10752, pp. 305-314, 2018.

[23] N. Sidaty et al., "Compression Performance of the Versatile Video Coding: HD and UHD Visual Quality Monitoring," *IEEE Picture Coding Symp. (PCS)*, Ningbo, China, 2019, pp. 1-5.

[24] A. Cerveira et al., "Memory Assessment of Versatile Video Coding," *IEEE Int. Conf. Image Process. (ICIP)*, Abu Dhabi, United Arab Emirates, 2020, pp. 1186-1190.

[25] F. Pakdaman et al., "Complexity Analysis of Next-Generation VVC Encoding and Decoding," *IEEE Int. Conf. Image Process. (ICIP)*, Abu Dhabi, United Arab Emirates, 2020, pp. 3134-3138.

[26] M. Saldanha et al., "Complexity Analysis of VVC Intra Coding," *IEEE Int. Conf. Image Process. (ICIP)*, Abu Dhabi, United Arab Emirates, 2020, pp. 3119-3123.

[27] F. Bossen et al., “JVET common test conditions and software reference configurations for SDR video,” JVET 20th Meeting, JVET-T2010, Oct. 2020.

[28] G. Bjontegaard, "Calculation of Average PSNR Differences between RDCurves," VCEG Meeting, pp. 1-5, 2001.