High-Throughput Hardware for 3D-HEVC Depth-Map Intra Prediction

Journal:	*IEEE Design & Test*
Manuscript ID	DTSI-2018-10-0119
Manuscript Type:	SI: SBCCI 2018
Date Submitted by the Author:	31-Oct-2018
Complete List of Authors:	Ucker, Mariana; Universidade Federal de Santa Catarina Afonso, Vladimir; UFRGS, PGMicro; UFPel, GACI Saldanha, Mario; Universidade Federal de Pelotas Audibert, Luan; Universidade Federal de Pelotas Conceição, Ruhan; Universidade Federal de Pelotas, CDTec - Centro de Desenvolvimento Tecnológico Susin, Altamiro; Fed Univ of Rio Grande do Sul, Electrical Engineering; Porto, Marcelo; Universidade Federal de Pelotas, CDTec - Centro de Desenvolvimento Tecnológico Zatt, Bruno; Universidade Federal de Pelotas, CDTec - Centro de Desenvolvimento Tecnológico Agostini, Luciano; Universidade Federal de Pelotas
Keywords:	3D-HEVC, Intra Prediction, Depth Map, Hardware Design

High-Throughput Hardware for 3D-HEVC Depth-Map Intra Prediction

Mariana Ucker; Vladimir Afonso; Mário Saldanha; Luan Audibert; Ruhan Conceição; Altamiro Susin; Marcelo Porto; Bruno Zatt; Luciano Agostini

***Abstract*— The 3D-HEVC is the state-of-art encoder for 3D videos. This article focuses on the 3D-HEVC depth-map intra prediction. Novel heuristics to reduce the computational effort are introduced together with a high-throughput architecture which supports the 3D-HEVC tools and the tools inherited from the HEVC. The architecture was synthesized targeting a 45nm technology and, when running at 908 MHz, it can process nine views of HD 1080p videos at 30 fps with a power dissipation of 41.57mW. This is the first article relating a hardware design supporting both the novel 3D-HEVC and the conventional HEVC intra prediction tools.**

***Index Terms*—3D-HEVC, Intra Prediction, Depth Map, Hardware Design**

I. Introduction

Nowadays, the devices capable of dealing with digital videos become very popular pushed by the consumers' interest and by the video support available from social networks, messaging platforms, and video-streaming services (Facebook, Instagram, WhatsApp, YouTube, Netflix, among others). The increasing interest of the consumers for high-quality and immersive video experiences is collaborating for advances in both the capability and functionalities of the devices, as smartphones and tablets, even with the limitations regarding performance, storage, bandwidth, and energy consumption of battery-powered devices.

In this context, new video technologies are emerging with the goal to provide high-quality and immersive video experience for the users. Due to this movement, entertainment industries have invested in 3D-movies, living games, and virtual-augmented-mixed realities, among others. Considering 3D videos, handheld devices capable of dealing with multiview content under a multiview plus depth (MVD) approach [1] are expected, driven by emerging technologies like Microsoft RealSense 3D, Structure Sensor, Stereolabs ZED, and Apple TrueDepth.

In order to efficiently deal with the huge amount of data to be transmitted and storage considering a multiview approach, and also to improve view synthesis performance, 3D-High Efficiency Video Coding (3D-HEVC) [1] has emerged in 2015 along with the Multi-View plus Depth (MVD) format.

In a MVD format, each view is formed by a texture channel and a depth channel, where the texture pictures are the pictures normally showed to the users and the depth maps are used to represent the distance between the scene objects and the camera. The objects near to the camera are usually represented by light gray tones and the objects far to the camera are represented by dark gray tones.

By adopting the MVD format, the 3D-HEVC enables the generation of intermediate synthetic views through a View Synthesis process called Depth-image-based rendering (DIBR) [1] at the receiver side. This way, the encoder transmits a reduced number of original texture pictures and their respective depth maps whereas the receiver generates synthetic texture pictures at different viewpoints using the DIBR process.

To efficiently encode depth maps, the 3D-HEVC adopts new encoding tools. In intra prediction, 3D-HEVC introduces: (i) Depth Intra Skip (DIS); (ii) Depth Modeling Mode-1 (DMM-1); and (iii) Depth Modeling Mode-4 (DMM-4). These tools increase the coding efficiency but also increase the required computational effort.

The main works published in the literature that propose hardware designs for the 3D-HEVC DIS, DMM-1 and/or the DMM-4 are [2][3][4][5][6]. It is important to notice that there are no published works relating hardware designs supporting all depth maps intra prediction modes.

The work presented in [2] implements the DMM-4 prediction mode for 8x8, 16x16, and 32x32 block sizes. The architecture was synthesized for an Altera Stratix V FPGA, and it is capable of processing HD 1080p videos (1920x1080 pixels) with five views at 31.39 frames per second (fps).

The work presented in [3] implements the DMM-1 and DMM-4 prediction modes, which can be scaled for all block sizes. DMM-1 memory issues are not covered by this work. The architecture was synthesized for ASIC and it is capable of processing HD 1080p videos at 30 fps considering one view.

The work presented in [4] implements the three modes introduced in the 3D-HEVC extension: DIS, DMM-1, and DMM-4 considering all possible block sizes. A strategy to reduce the DMM-1complexity was implemented, but memory issues were not cover. The developed architecture was synthesized for a Virtex 6 FPGA and it can process HD 1080p videos at 30 fps, considering six views.

The work presented in [5] implements a hardware design for DIS mode, considering all possible block sizes. A simplification was implemented to process the similarity criterion. Also, strategies of data reuse were used in the hardware design. This architecture was synthesized for ASIC and it processes UHD 2160p (3840x2160 pixels) videos at 60 fps considering five views.

The work [6] was a previous version of this work, targeting

a low-power and high-throughput architecture but only for the DMM-4 mode. This architecture supports 8x8, 16x16, and 32x32 block sizes and it was synthesized targeting ASIC and FPGAs reaching throughput enough to process at least two views of UHD 2160p 3D videos at 60 fps.

The works [2], [3] and [4] are not fully compliant with the 3D-HEVC standard. The works [2] and [3] present differences in the method used to calculate the texture block average in DMM-4 besides the work [4] uses an alternative method to calculate the DIS.

This article targets to overcome some gaps that remain uncovered among the works published in the literature, as follows: (i) there are no works that implement both the novel 3D-HEVC intra-prediction modes and the modes inherited from HEVC; (ii) some works are not fully compliant with the 3D-HEVC standard; (iii) DMM-1 hardware designs usually store all possible *wedgelets* with a significant cost in the memory-related energy.

II. Background

The intra-frame prediction is responsible for exploiting the spatial redundancy presented in neighboring regions inside the same picture. The 3D-HEVC intra prediction introduced novel coding tools [1] along with the ones inherited of HEVC [7] to enhance the coding efficiency of depth maps.

The 3D-HEVC divides the frames into smaller square-shaped block sizes before applying the coding tools. These blocks are called Coding Tree Units (CTUs) which can have a maximum size of 64x64 pixels. The CTUs can be divided into smaller blocks according to the encoding tool. In this context, the Intra-frame prediction tools are applied to 4x4, 8x8, 16x16, 32x32, and 64x64 blocks.

A. 3D-HEVC intra-prediction modes inherited from HEVC

The 3D-HEVC inherits all the 35 intra-frame prediction modes available in the HEVC standard: (i) Planar; (ii) DC; (iii) Angular modes (33 modes).

The Planar prediction mode is efficient to predict regions with smooth or complex textures that are not efficiently predicted by the angular modes [8]. The Planar mode is calculated through the average of two interpolation filters. These filters use reference samples belonging to the adjacent blocks, located above and to the left of the block being encoded.

The DC prediction mode is capable to efficiently encode homogeneous regions [8]. To use the DC mode, one pre-calculated value is copied for all samples of the predicted block.

The Angular modes aim an efficiency modeling of directional structures that exist in the video textures [8]. Thirty three different angular modes based on no-equidistant angles are available.

B. Novel 3D-HEVC intra prediction modes

As discussed before, the 3D-HEVC introduced DIS, DMM-1, and DMM-4 intra-prediction modes [1] to better deal with the depth-maps features.

Depth Intra Skip (DIS) mode was introduced to deal with the larger homogeneous regions, avoiding the encoding process of the residues for these regions. DIS implements four sub-modes: a) Horizontal Intra Prediction (IP_H); b) Horizontal Single Depth (SD_H); c) Vertical Intra Prediction (IP_V); and d) Vertical Single Depth (SD_V). The IP_H and IP_V modes use the same vertical and horizontal filters of the HEVC Intra prediction modes horizontal and vertical. The Single Depth (SD) modes simply consist in copy the value of one neighboring sample to represent all predicted block. DIS supports 8x8, 16x16, 32x32, and 64x64 blocks.

The Depth Modeling Mode 1 (DMM-1) defines that the samples of the block to be predicted are separated into two partitions by a straight line, called *Wedgelet*. Each partition is predicted through a Constant Partition Value (CPV) calculated by the average using its correspondent samples and, so, each CPV is used to entirely represent each partition. An alternative to obtain maximum efficiency with the DMM-1 partition scheme is to test all 1,908 possible *wedgelets* combining start and end points from the different block borders, which brings a high computational effort. To reduce the complexity associated with the DMM-1 mode, the 3D-HEVC Reference Software (3D-HTM) [9] proposes a simplification. Even so, DMM-1 is a bottleneck in both processing and memory since all possible *wedgelets* are stored in memory.

The Depth Modeling Mode 4 (DMM-4) also divides the block to be predicted into two partitions as the DMM-1, but the DMM-4 mode uses a threshold value for calculating the contour of the predicted-block objects. The threshold value is obtained from an average calculation with specific samples of the collocated texture block. Furthermore, discontinuous regions may belong to each partition, which is not possible with the DMM-1.

DMM-1 and DMM-4 modes support 4x4, 8x8, 16x16 and 32x32 block sizes.

III. 3D-HEVC Depth-map Intra-Prediction Analysis

This section presents a comprehensive statistical analysis of the 3D-HEVC depth-map intra prediction tools usage. Among the data provided by the experiments, the most prominent modes and block sizes used in the depth-map coding were identified.

All experiments were performed using the 3D-HTM in version 15.1 [9]. Also, the experiments considered the 3D-HEVC Common Test Conditions (CTC) [10] that defines eight videos at 1920x1088 and 1024x768 resolutions and four Quantization Parameter (QP) sets. The results presented in this section consider the average of all videos and QPs running under the Random Access (RA) temporal configuration [2].

The statistical analysis of the intra-prediction tools usage is based on the percentage of pixels that are encoded using each encoding tool, mode and block size.

The first important result is that the 3D-HTM depth-map coding used the DIS tool to encode 81.07% of the depth-map pixels, i.e., all the remaining 37 prediction modes are responsible for 18.93% of the pixels. Then, the high importance of the DIS tool becomes evident. But this fact cannot lead us to consider that the other encoding tools are

less important than the DIS, since the other tools better deal with sharp edges and directional structures which are extremely important to the view synthesis process.

Since the DIS tool is much more used than the others, the Intra-frame prediction distribution was analyzed disregarding the DIS tool. Considering the remaining 37 prediction modes, the Planar mode is the one with higher representativeness, encoding 29.37% of the depth-map pixels, followed by the DMM-4 mode with 17.18%, the DMM-1 mode with 13.91%, and the DC mode with 6.47%. The fifth mode with higher representativeness is the Vertical mode, encoding 3.85% of the depth-map pixels, followed by the Horizontal mode with 1.83%. All the other prediction modes encode no more than 1% of the pixels. If these six modes are considered together, they are responsible for encoding 72.61% of the depth-map pixels when DIS is not considered.

Regarding the use of block sizes inside the intra prediction of depth maps, the first conclusion is that the 4x4 block size is unrepresentative, corresponding to only 0.28% of the pixels in the depth-map coding. Disregarding the DIS tool, the results show that the bigger block sizes encode a higher amount of depth-map pixels when compared to the smaller sizes. The 4x4 block size encodes only 1.48% in this case whereas 32x32 and 64x64 block sizes have representativeness in the encoding of 36.12% and 44.37% of the depth-map pixels, respectively. The 16x16 block size encodes other 13.78% whereas the 8x8 block size encodes 4.25% of the depth-map pixels disregarding the DIS tool.

IV. Hardware-Oriented Heuristics for the 3D-HEVC Depth-Map Intra Prediction

Based on the analysis presented in the previous section we propose hardware-oriented computer effort reduction heuristics for the depth maps intra prediction that remove the less significant prediction modes and block sizes.

We decided to maintain the seven most representative prediction modes and the four most representative block sizes, which corresponds to 94.6% of all encoded depth-map pixels considering the regular encoding process. Therefore, DIS, Planar, DC, Horizontal, Vertical, DMM-1, and DMM-4 modes and 8x8, 16x16, 32x32, and 64x64 block sizes were selected to be supported in the hardware design.

Furthermore, we decide to introduce other simplifications to reduce DMM-1 processing and memory requirements, since this is the most computational intensive mode among the supported ones. The used heuristic reduces the number of *wedgelets* tested during the encoding process to six. The six *wedgelets* are defined using a pre-processing through a gradient calculation using the samples along the four neighboring-block borders. Then, all DMM-1 *wedgelets* are available, but only six will be evaluated. The heuristic also avoids memory usage for *wedgelet* patterns storage using the Bresenham [11] algorithm to computed at run-time the DMM-1 bitmaps for the six evaluated *wedgelets*. The Bresenham is widely used in *Computer Graphics* to perform the rasterization of lines and polygons. Our modified Bresenham implementation receives the start and end points of the *wedgelet* and processes a pixel of the line representing the *Wedgelet* at time. Then, the 1,908 bitmap *wedgelet* patterns that must be stored when using the conventional approach were reduced to only six evaluations without any storage.

In order to verify the impact of the developed heuristics on the coding efficiency, these hardware oriented heuristics were inserted in the 3D-HTM and compared with the original results using the same test conditions presented in Section III. For this evaluation, two parameters are considered, as follows: (i) the encoding time, used to estimate the computational effort reduction; and (ii) the Bjontegaard Delta Rate (BD-rate) [12] parameter, to evaluate the bit-rate variation for the same image quality.

Considering the RA configuration, the BD-Rate increased 2.64% in the synthetic views, on average. On the other hand, the computational effort was reduced in 32.8% when encoding depth maps and in 16.2% for the global 3D-HEVC encoder. Regarding the All Intra (AI) temporal configuration, i.e., only intra prediction tools are used in the encoding process, the simplifications increase the BD-Rate by 7.16% in the synthetic views, on average. In this case, the encoding-time of depth maps is reduced in 54% and the total encoding time is reduced in 46.6%.

V. 3D-HEVC Depth-Map Intra-Prediction Architecture

This section presents the architecture designed for the depth-map intra prediction using the heuristics presented in the last section. Therefore, the architecture was designed to efficiently process the modes Planar, DC, Horizontal, Vertical, DIS, DMM-1 and DMM-4, supporting 8x8, 16x16, 32x32 and 64x64 block sizes. The heuristic to simplify the DMM-1 mode was also considered.

The designed architecture is presented in Fig. 1 and is divided in Processing Cores (PC), where each PC is responsible for processing a subset of the pre-defined intra-prediction modes in an interleaved way, as presented in Fig. 2. Then it is possible to reach the desired throughput, avoiding unnecessary calculations.

The seven supported modes are distributed in three processing units that work in parallel, as presented in Fig. 1. The intra tools are grouped according to their similarities which allows a reduction in area usage (and power dissipation) by reusing the hardware inside each PC. In the high-level diagram of the architecture presented in Fig. 1, one can see the mode distribution on the three PCs: (i) Planar, DC and DMM-4; (ii) IP_H, IP_V, SD_H, SD_V; (iii) DMM-1. The IP_H and IP_V modes are used both as HEVC conventional modes as DIS sub-modes, allowing the reuse of PC-2 results. Although the DMM-1 and DMM-4 modes have similarities, these modes were not grouped in the same PC in order to obtain the desired throughput, since DMM-1 is much more complex than the other modes.

The designed hardware is a multiplierless solution, where the multiplications were replaced by shift-adds to save hardware resources (and power). Clock-gating was also

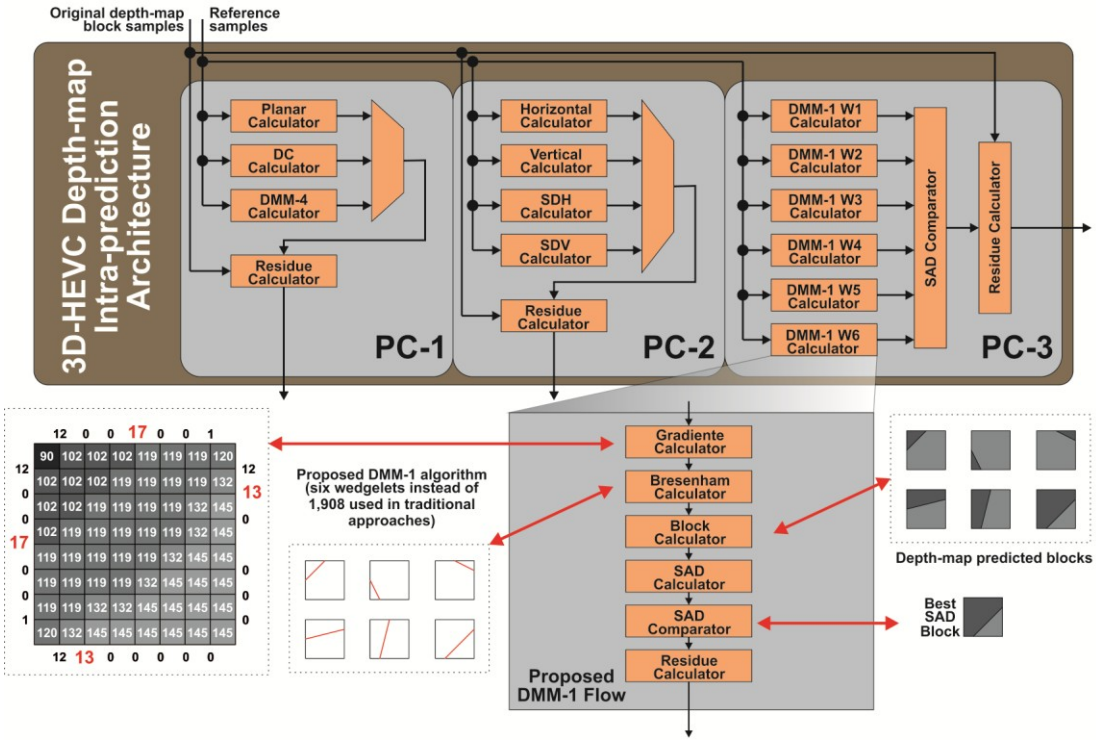

Fig. 1 – High-level block diagram of the developed architecture.

widely used to idle modules during the architecture operation. The architecture has a parallelism level enough to deliver one block line per clock cycle independently of the block size.

The Processing Core-1 (PC-1) is responsible for processing the Planar, DC, and DMM-4 modes. Each mode is also processed by an independently module, as one can observe in Fig. 1. As inputs, the PC-1 receives the left and upper edges of neighboring blocks to the block being encoded (previously reconstructed blocks), which are used for DC and Planar calculations. For the DMM-4 calculations, the PC-1 architecture also receives four 8-bit width samples from the texture, which are required for the threshold calculation. PC-1 also receives as inputs the samples of the depth-block being predicted to calculate the residues. As one can see in the Timing Analysis presented in Fig. 2, the residue calculation starts at the moment that the Planar mode avails the first line of samples. The DC calculator needs one cycle before the residue calculation.

The PC-2 is responsible for calculating the IP_H, IP_V, SD_H and SD_V modes and each mode is processed by an independent module, as presented in Fig. 1. As inputs, PC-2 receives the left and upper edges of neighboring blocks to the block being encoded (previously reconstructed blocks). Also, the PC-2 receives as inputs the samples of the block being predicted to calculate the residues. A multiplexer selects which samples can follow to the Residue Calculator at each moment. The coding of the four modes starts along with the residue calculation, avoiding any additional clock cycle besides the ones used by the Residue Calculator, as presented in Fig. 2.

The PC-3 is responsible for calculating the DMM-1. As discussed in Section IV, we used a heuristic to drastically reduce the DMM-1 computational effort and to avoid the storage of *wedgelet* patterns in memory. The heuristic consists of processing only the six *wedgelets* formed from the four higher gradients along the four borders of the block to be predicted (one gradient per border), as detailed in the example given in Fig. 1.

PC-3 processes these six *wedgelets* in parallel, as presented in Fig. 1 and Fig. 2. As inputs, PC-3 receives the samples of the block being predicted to calculate the residues.

The first DMM-1 module consists of a Gradient Calculator that generates six outputs determining the start and end points of the six *wedgelets*.

Once the positions to compose the *Wedgelet* are available, the bitmap of the predicted block can be obtained using the Bresenham algorithm in order to indicate which samples belong to each partition. It is important to notice that the Bresenham algorithm does not predict the number of samples the line will have. Thus, it is not possible to predict the total number of required clock cycles. The architecture was designed to cover the worst case of the Bresenham algorithm.

The next step is the Block Calculator that starts its operation processing the bitmap line by line. After the Block Calculator generates the predicted block lines, the SAD calculator generates the sum of absolute differences related with each one of the six evaluated *wedgelets*.

Finally, PC-3 has a comparator to compare the six SAD values and decide the most efficient *wedgelet* before calculating its residue.

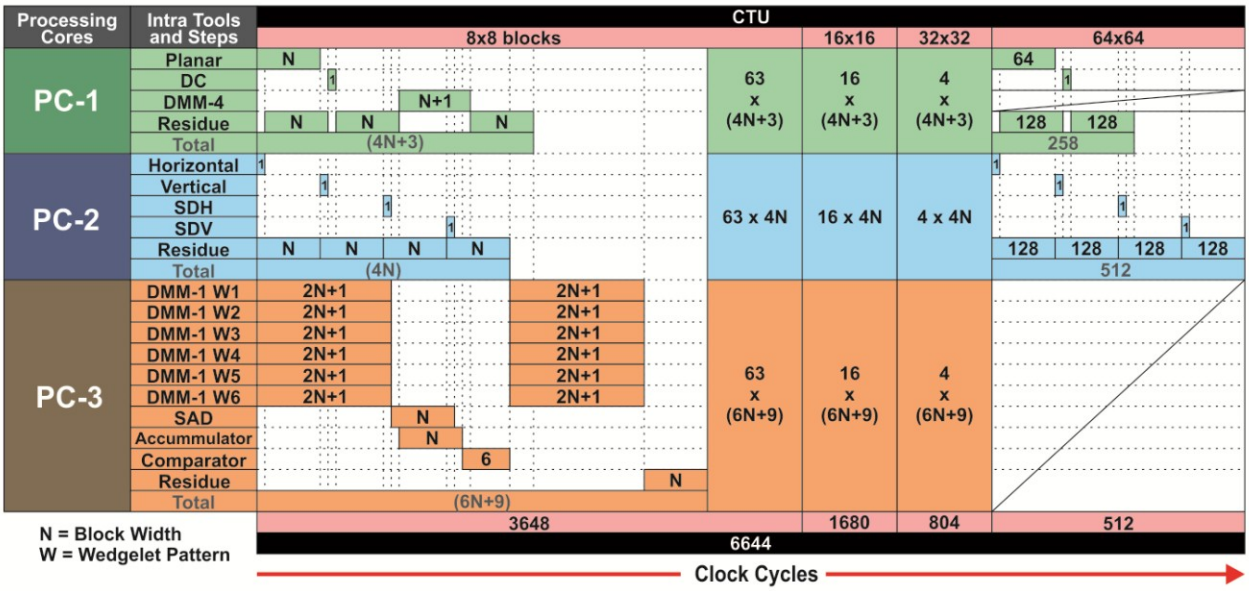

Fig. 2 – Timing analysis of the developed architecture.

VI. Synthesis Results and Discussion

The developed architecture was described in VHDL and synthesized targeting the Nangate 45nm technology. Table I summarizes our hardware results in the rightmost column.

Table I also presents comparisons with related works. To the best of our knowledge, this is the first hardware design to process depth maps that supports both the novel 3D-HEVC intra prediction modes and the HEVC inherited intra prediction modes. Although the literature presents some works with hardware designs targeting the HEVC intra prediction modes, those works do not consider the MVD approach, making unfair the comparisons with our work.

Among the related works targeting 3D-HEVC intra prediction tools, the five hardware designs found in the literature were selected for comparisons. It is important to emphasize that the work [5] implements only the DIS, the works [2] and [6] implement only the DMM-4, the work [3] implements both DMM-1 and DMM-4, and the work [4] implements DIS, DMM-1, and DMM-4. It is also important to emphasize that works [2], [3] and [4] are not fully compliant with the 3D-HEVC standard.

As one can observe in Table I, only the architecture designed in this work processes HEVC inherited modes (Planar, DC, Horizontal, Vertical) besides the DIS, DMM-1 and DMM-4 modes. Whereas the works [2] and [4] focused on FPGA devices, this work and the works [3], [5] and [6] are focused on ASIC technology. This work and the other three ASIC-based works present power results and even our hardware supporting a high number of prediction modes, it presents competitive results.

As one can notice in Table I, only our design and the designs presented in [4], [5] and [6] are able to process 3D-

TABLE I
Synthesis Results

Work	Afonso [5]	Sanchez [2]	Ucker [6]	Sanchez [3]	Amish [4]	This Work
Encoding Tools	DIS	DMM-4	DMM-4	DMM-1 and DMM-4	DIS, DMM-1, and DMM-4	DIS, DMM-1, DMM-4, H, V, Planar, and DC
Technology	ASIC Nangate 45nm	FPGA Altera StratixV 5SGXMABN3F45I32 (28nm)	ASIC Nangate 45nm	ASIC 65nm	FPGA Xilinx Virtex 6 (40nm)	ASIC Nangate 45nm
Area	35.88 k gates (2-input NAND)	5,568 ALMs 4,405 Registers	139.84 k gates (2-input NAND)	219.95 k gates* (2-input NAND)	55 k LUTs 67 k Registers	486,804 gates (2-input NAND)
Memory for DMM-1 wedgelets	-	-	-	Yes (101,352 bits*)	Yes (1,947 k bits)	No
Maximum Frequency	550 MHz	31.3 MHz	750 MHz	53.2 MHz*	275 MHz	940 MHz
Frequency for HD1080p@30@5views	63.18 MHz	22.47 MHz	210.5 MHz	Not reached	165 MHz	504.5 MHz
Power for HD1080p@30fps	19.89 mW / 0.95 V (40 views)	Not available	22.3mW / 0.95 V (5 views)	166.5 mW* (1 view)	Not available	41.57 mW / 0.95 V (9 views)
3D-video processing at UHD2160p@30fps	Yes (10.88 views)	No (1.74 views)	Yes (4.46 views)	No (0.25 views)	Yes (2.08 views)	Yes (2.33 views)

*Values consider only the architecture to process 32x32 blocks.

videos at UHD2160p@30fps (with at least two views). However, the work [4] requires an external memory to store all the possible DMM-1 *wedgelets* patters which is not necessary in our architecture. Similarly, the work [3] also needs the external memory. These works also did not present the power dissipation related to the memory used to store the *wedgelets*.

Since the other works only support a few intra prediction modes and do not present the BD-rate results for the complete intra prediction, as we presented, it was not possible to compare the BD-rate results of the these works with our work.

VII. Conclusions

This article presented the first dedicated hardware architecture for the 3D-HEVC depth maps intra prediction covering both the novel 3D-HEVC encoding tools and the HEVC inherited coding tools. In order to enable an efficient hardware design, experiments were carried out with the 3D-HTM to identify possible simplifications in the number of supported modes and block sizes. In addition, this work introduced a new heuristic for DMM-1 *wedgelets* calculation using the Bresenham algorithm at run time, avoiding the use of memory. Synthesis results demonstrate that the architecture can process 3D videos at UHD2160p@30fps (two views) and, when operating at 908 MHz, the architecture can process nine views at HD1080p@30fps with a power dissipation of 41.57mW.

VIII. References

[1] G. Tech, et al. "Overview of the multiview and 3D extensions of high efficiency video coding*." IEEE Trans. Circuits Syst. Video Technol.*, vol. 26, n. 1, pp. 35-49, Jan. 2016.

[2] G. Sanchez, et al."A real-time 5-views HD 1080p architecture for 3D-HEVC Depth Modeling Mode 4". *Integrated Circuits and Systems Design (SBCCI), Symp* , pp. 1-6. 2014.

[3] G. Sanchez, et al."Real-time scalable hardware architecture for 3D-HEVC bipartition modes". *J. Real-Time Image Processing*, pp. 71-83.

[4] F. Amish, and E. Bourennane. "An efficient hardware solution for. 3D-HEVC intra-prediction". *J. Real-Time Image Processing*, pp 1-13.

[5] V. Afonso et al. "Low-Power and High-Throughput Hardware Design for the 3D-HEVC Depth Intra Skip". *IEEE International Symposium on Circuits & Systems*,[S.l.], May 2017.

[6] M. Ucker et al. "Low-Power and High-Throughput Architecture for 3D-HEVC Depth Modeling Mode 4". *Integrated Circuits and Systems Design (SBCCI), Symp* , pp. 1-6. 2018.

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

[8] J. Lailema. Intra-Picture Prediction in HEVC. In: V. Sze; M. Budagavi; G. Sullivan. (Ed.). High efficiency video coding (HEVC). [S.l.]: Springer, 2014. p.91–112.

[9] Three-Dimensional High Efficiency Video Coding (3D-HEVC). Reference Software Retrieved Oct. 29, 2016: https://hevc.hhi.fraunhofer.de/svn/svn_3DVCSoftware/tags/HTM-15.1/.

[10] K. Müller, and A. Vetro, "Common Test Conditions of 3DV Core Experiments," *JCT3V-G1100*, San José, US, USA, Jan. 11-17, 2014.

[11] J. Bresenham. Algorithm for computer control of a digital plotter. IBM Systems Journal, [S.l.], v.4, n.1, p.25–30, 1965.

[12] G. Bjontegaard, "Improvements of the BD-PSNR model," VCEG-AI11. Jul. 2008.

Mariana Ucker is an Electrical Engineering graduate at Sul-rio-grandense Federal Institute, Brazil. She is a master student in Electrical Engineering at Federal University of Santa Catarina, Brazil.

Vladimir Afonso received the M.S. degree in computer science from Federal University of Pelotas, Brazil, in 2013. He is a Ph.D. candidate in microelectronics at the Federal University of Rio Grande do Sul, Brazil. Since 2009 he is a Professor at the Sul-rio-grandense Federal Institute, Brazil. His research interests include 2D/3D video-coding algorithms, and VLSI designs for video coding.

Mario Saldanha received the M.Sc. degree in Computer Science from Federal University of Pelotas (UFPel), Pelotas, Brazil, in 2018. He is currently pursuing the Ph.D. degree in Computer Science at the UFPel. His research interests include complexity reduction and hardware-friendly solutions for 2D/3D video coding.

Luan Audibert is a computer-science graduate at the Federal University of Pelotas, Brazil.

Ruhan Conceição received the M.S. degree in computer science from the Federal University of Pelotas, Brazil in 2016. He is currently pursuing the Ph.D. degree in computer science in the same institute. Since 2018, he is a professor at Sul-rio-grandense Federal Institute. His research interests include 2D/3D video coding standards and light-field imaging.

Altamiro Susin received the Ph.D. degree in microelectronics from the Polytechnique National Institute of Grenoble, France, in 1981. He is Full Professor at the Federal University of Rio Grande do Sul, Brazil, since 1975. He was co-founder of Brazilian Microelectronics Society and past President of the society. His main research interests include integrated circuit architecture, system-on-chip design, and signal processing.

Marcelo Porto received the his PhD. degree in computer science from the Federal University of Rio Grande do Sul, Brazil, in 2012. He is Professor since 2009 at Federal University of Pelotas, Brazil, and since 2016 he is a Brazilian CNPq PQ-2 distinguished researcher. His research interests include video coding, motion estimation algorithms, FPGA-based and VLSI design for video coding. He is Senior Member of IEEE and also member of the IEEE CAS and of the IEEE SPS.

Bruno Zatt received the his PhD. degree on microelectronics from the Federal University of Rio Grande do Sul, Brazil, in 2012. He is Professor since 2013 at Federal University of Pelotas, Brazil, and since 2017 he is a Brazilian CNPq PQ-2 distinguished researcher. His research interests include video coding, embedded systems, FPGA-based and VLSI design for video coding. He is Senior Member of IEEE and also member of the IEEE CAS and of the IEEE SPS.

Luciano Agostini received the Ph.D. degree from Federal University of Rio Grande do Sul, Brazil, in 2007. He is a Professor since 2002 at Federal University of Pelotas, Brazil. Since 2013 he is a Brazilian CNPq PQ-2 distinguished researcher. His research interests include 2D and 3D video coding, algorithmic optimization, arithmetic circuits and microelectronics. Agostini is a Senior Member of IEEE and ACM, and he is a member of the IEEE CAS, CS, and SPS societies.