Edge-Aware Depth Motion Estimation – A Complexity Reduction Scheme for 3D-HEVC

[1,3] Gustavo Sanchez, [2] Mário Saldanha, [2] Bruno Zatt, [2] Marcelo Porto, [2] Luciano Agostini, [1] César Marcon
[1] Pontifical Catholic University of Rio Grande do Sul – Porto Alegre, Brazil
[2] Federal University of Pelotas – Pelotas, Brazil
[3] IF Farroupilha – Alegrete, Brazil
gustavo.sanchez@acad.pucrs.br,{mrdfsaldanha, zatt, porto, agostini}@inf.ufpel.edu.br, cesar.marcon@pucrs.br

***Abstract*— This work presents the Edge-Aware Depth Motion Estimation (E-ADME), a complexity reduction scheme developed for depth maps coding on 3D High Efficiency Video Coding (3D-HEVC). This scheme focuses on obtaining an execution time reduction while keeping a high quality of the encoded depth map. E-ADME starts classifying each encoding depth block as edge or homogeneous. If the block is classified as an edge, then the Test Zone Search (TZS) is applied because edges require expensive comparisons to find the best block match. Otherwise, the scheme applies an Iterative-Small Diamond Search Pattern (I-SDSP), which is a lightweight center-biased algorithm for efficient encoding of homogeneous blocks. The proposed solution was capable of achieving a time saving of 6.9% in depth maps coding, increasing less than 0.15% the BD-rate of the synthesized view.**

Keywords—3D-HEVC; Motion Estimation; Depth Maps; Timesaving; Complexity Reduction

I. Introduction

The Joint Collaborative Team on 3D Video Coding Extension Development (JCT-3V) developed the 3D-High Efficiency Video Coding (3D-HEVC) [1], which is the most advanced ISO/IEC and ITU-T standard for 3D video coding. 3D-HEVC is an extension of the well-known 2D-HEVC standard that provides about 46% and 19% of bitrate savings when compared to HEVC simulcast and MV-HEVC (Multiview HEVC) standards [2], respectively. One of the key factors for achieving high efficiency in 3D-HEVC is the adoption of the Multiview Video plus Depth (MVD) [3] data format. MVD uses depth maps, which are information added to the texture frame when encoding a 3D video. Cameras with infrared sensors capture the depth maps and associate them with the corresponding texture frames.

A depth map provides geometrical information of the scene, which is essential to synthesize virtual views using techniques such as the Depth Image Based Rendering (DIBR) [3]. Fig. 1 displays a depth map of *Shark* video sequence. Moreover, Fig. 1 shows that depth maps normally contain large homogeneous regions in the body of the objects and background (1-4 detached in Fig. 1), and sharp edges in the border of the objects (5-8 detached in Fig. 1).

The 3D-HEVC standard uses 2D-HEVC algorithms to encode 3D videos. However, these algorithms were developed focusing on the texture properties. Encoding depth maps using 2D-HEVC algorithms can produce low-quality virtual views.

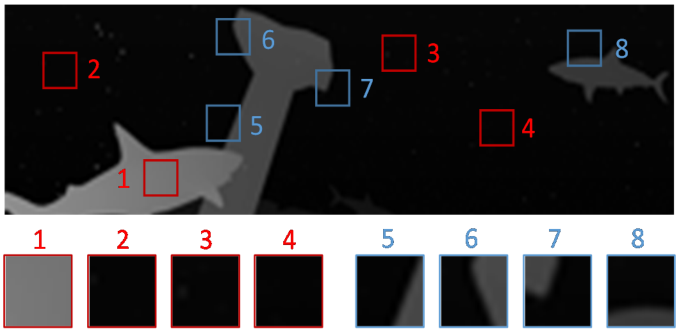


Fig. 1. Homogeneous (1-4) and edge (5-8) regions extracted from the depth maps of *Shark* video sequence [4].

Besides, these algorithms do not explore the characteristics of depth maps, which can lead to an unnecessary effort. In a world dominated by battery-powered embedded devices for video domain, *there is a need for complexity reduction techniques focusing on depth maps coding without inserting significant coding efficiency losses.* Moreover, this scenario demands complexity reduction techniques for achieving real-time encoding performance and energy saving.

The inter-frame prediction of 3D-HEVC depth maps, composed of Motion Estimation (ME) and Motion Compensation (MC), is inherited from 2D-HEVC texture coding. This prediction tool is a good candidate to apply complexity reduction algorithms; because these algorithms demand a high computational effort and depth maps can be encoded with simpler algorithms. In this work, execution time is used as a metric to measure the complexity.

The depth maps inter-frame prediction of 3D-HEVC only differs from 2D-HEVC by disabling the Fractional Motion Estimation (FME) algorithm to preserve the edge information in depth maps. It happens because FME smooth the depth maps borders producing incorrect interpretation between background and foreground pixels in the view synthesis [3].

Our previous works [5] and [6] demonstrate that ME was developed focusing on texture coding, which presents a complex behavior and concludes that simpler algorithms can be used for ME of depth maps with small impact on the coding efficiency. These same works employ lightweight fast ME algorithms, such as the Iterative-Small Diamond Search Pattern (I-SDSP) that reduces the coding effort with negligible impact on video quality, instead of using the Test Zone Search (TZS) algorithm [7]. However, the works [5] and [6] suppose that all depth-

ISBN 978-0-9928626-7-1 © EURASIP 2017

encoding blocks require low execution complexity and ignore the fact that edges tend to be harder to predict than homogenous regions in depth maps. Based on this fact, this paper proposes a new complexity reduction scheme for depth maps ME called Edge-Aware Depth Motion Estimation (E-ADME).

The E-ADME scheme can achieve high coding efficiency providing high quality on synthesized views, considering the edge and homogenous regions of depth maps. E-ADME is composed of three algorithms: (i) Simplified Edge Detector (SED) [8] that detects if the encoding block contains an edge or a homogeneous region, (ii) I-SDSP that encodes homogeneous regions, and (iii) TZS that encodes edge regions.

The proposed solution surpasses the coding efficiency of the traditional and related works methods because E-ADME dynamically distributes the computational effort of ME according to the content of the encoding block. The three most important contributions of this paper are listed below:

- **Analysis of the motion estimation in depth maps edges** – the previous works [5] and [6] only considered that depth maps coding can use simpler ME algorithms, considering a depth map as homogeneous regions. This paper presents an analysis to observe the depth maps ME behavior and concludes that edge region is harder for encoding than homogeneous region.
- **Definition of the equations to detect edges for all allowed ME block sizes** – the work [8] that presented SED was capable only of detecting edges in quadratic blocks ranging from 4×4 to 32×32. In this work, we generated new thresholds for ME blocks with the width of 12, 24, and 64 samples.
- **The design of an edge-aware complexity reduction ME scheme for depth maps coding** – the previous solutions always required to change the ME algorithm without considering the regions that need further evaluation. The proposed solution considers that edge regions should be evaluated by a more sophisticated ME algorithm, such as TZS, providing higher quality on the synthesized views.

II. Motion Estimation and Related Work

Similar to the texture inter-frame prediction, the inter-frame prediction of depth maps applies ME using a search algorithm to detect the most similar block inside a search area in the reference frame with the current encoding block, as displayed in Fig. 2. ME is one of the critical tasks inside block-based video encoders [9]. Thus, the selection of a better ME algorithm produces fewer comparisons, resulting in fewer reference memory access (memory that stores the reference frame).

In the 3D-HEVC Test Model (3D-HTM), the TZS is used as the ME search algorithm and the Sum of Absolute Differences (SAD) is used as the similarity criterion.

The TZS algorithm can reach a near optimal performance (i.e., the TZS can obtain results very similar to a brute force approach) regarding the quality of the search process. TZS employs an Initial Search Point (ISP) decision and then performs an iterative approach around the best ISP.

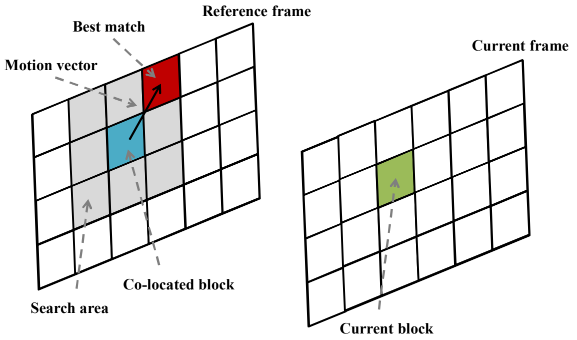


Fig. 2. Example of ME block matching.

TZS selects the best ISP comparing the SAD results of: (i) the motion vector pointing to the co-located block (i.e., the block in the same position in the reference frame); (ii) the median vector of the encoding block neighborhood (composed of previously encoded blocks around the current block); and (iii) the motion vector of the largest block for the current Coding Unit (CU) [10]. Besides, TZS uses different search patterns in the search process, such as expansive diamond pattern, raster search and refinement step [7]. Applying the ISP technique combined with these search machines allows reducing the Full Search (FS) complexity in around of 23 times with almost no impact on the encoded video quality [7]. Although reducing the FS encoding time significantly, TZS still requires a considerably high number of SAD comparisons when compared with lightweight ME algorithms such as I-SDSP.

There are many proposed solutions to reduce the ME complexity in texture data for 2D/3D videos with low impact on coding efficiencies, such as [11]-[14].

The works [11] and [12] use TZS as the basic ME algorithm with some simplifications. Purnachand et al. [11] propose a different search pattern as an alternative to the TZS search pattern and its refinement step. Besides, an early termination algorithm based on past-encoded frames was proposed in [11]. Pan et al. [12] propose to accelerate the TZS process based on the median predictor probability and the size of the current CU.

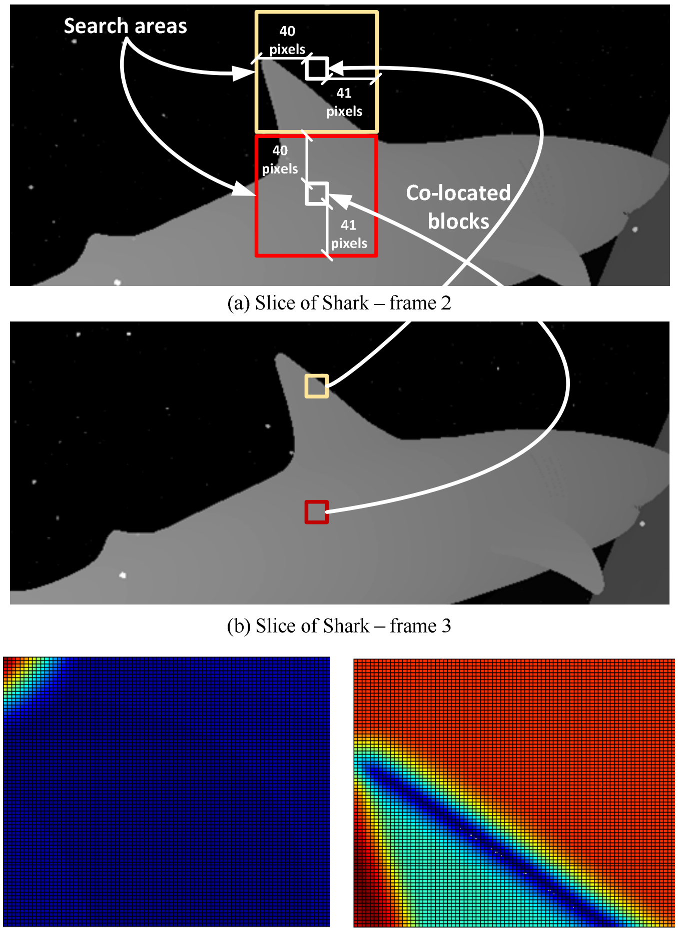
The work of Liao and Shen [13] reduces the ME search window according to the motion vector obtained by the 64×64 block. Consequently, a simplification is possible for smaller size blocks since these blocks can be searched using a smaller search area, according to the vector obtained in the 64×64 block.

Sanchez, Porto, and Agostini [14] present an ME algorithm that starts with multiple search points to surpass local minima. They proposed a hardware design that spreads through the search area for finding low SAD regions quickly. However, all these solutions were developed focusing on texture information, which presents complex content characteristics. Due to the distinct features of the texture and depth channels, these algorithms can lead to an unnecessary computational effort, mainly when homogenous regions of depth maps are considered.

To the best of our knowledge, there is no other work for the motion estimation of depth maps targeting complexity reduction and coding efficiency beyond our previous works [5] and [6]. Besides, these works propose to use only a classic lightweight fast algorithm focusing on reducing the encoding complexity of depth map ME or energy consumption, such as I-SDSP, Diamond Search (DS), and One at a Time Search (OTS).

III. ME Analysis of Depth Maps

This section presents the evaluation done to observe the depth maps ME behavior. Fig. 3(a) and (b) show a slice of two consecutive frames from depth maps of *Shark* video sequence (frame 2 and 3 of view 9). Two detached blocks with 16×16 pixels in Fig. 3(b) were applied to the FS algorithm, whose search area of [-40, 41] are displayed in Fig. 3(a) with the same color. The yellow box in Fig. 3(b) starts in the pixel (240, 340) and the red box starts in the pixel (240, 440). Fig. 3(c) and (d) show the SAD heat map for the ME process of a given encoding block in homogeneous (red box in Fig. 3(b)) and an edge region (yellow box in Fig. 3(b)), respectively. Dark blue regions denote the lowest SAD values, whereas red regions represent the higher SAD values, meaning the best and worst values, respectively.



(a) Slice of Shark – frame 2

(b) Slice of Shark – frame 3

(c) Homogeneous region heat map

(d) Edge heat map

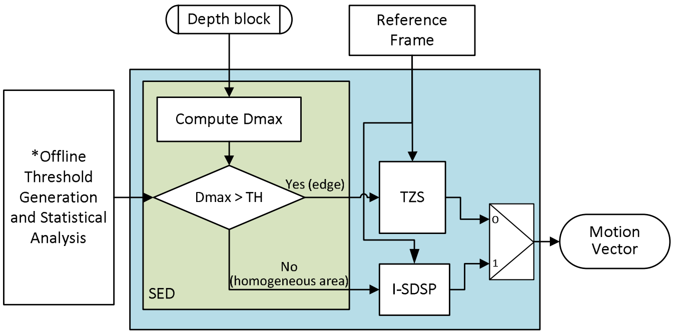
Fig. 3. Slices of Shark video sequence with (a) two detached search areas and (b) two detached encoding blocks. SAD heat map regarding (c) homogeneous and (d) edge regions.

One can notice from Fig. 3 that when encoding a homogeneous region block, the heat map shows smooth changes, with large regions containing low SAD values around the center of the search area. Considering it, homogeneous regions in depth map can be encoded using center-biased (i.e., algorithms that start searching in the center of the search area) light-weight fast ME algorithms, such as I-SDSP or DS, as described in [5] and [6], achieving near-optimal results in fewer SAD comparisons than TZS. However, the analysis of the SAD heat map of an edge in Fig. 3(d) allows seeing that the map has higher pattern variability and, in this case, more sophisticated ME algorithms are necessary for providing higher quality on synthesized views.

One can notice that low SAD values are found in a region that is not center-biased. In this case, an ME algorithm should be capable of converging quickly to this region. Therefore, a scheme able to identify if the content of the encoding block is an edge or a homogeneous region is necessary for both to reduce the computational effort of ME and preserve the coding efficiency. Thereby, the ME of depth maps can perform a more sophisticated algorithm for edge regions or a simple center-biased (i.e., that performs its search around the central position) algorithm for homogeneous regions and can reduce the implementation complexity with a negligible impact on the video quality. Moreover, solutions based on these considerations enable to build a real-time hardware design and reduce the energy consumption, since fewer memory accesses are required as an outcome of reduction in SAD computations.

IV. E-ADME Scheme

Fig. 4 illustrates the E-ADME scheme, which starts using the SED algorithm [8] to identify if the encoding block contains an edge or a homogeneous region. When the encoding block is classified as homogeneous, the lightweight I-SDSP algorithm is applied due to its efficiency for this kind of scenario. I-SDSP can find low SAD values around the center-biased position, thus accelerating the ME encoding flow. When SED classifies the encoding block as an edge, the conventional 3D-HEVC ME encoding flow is performed using the TZS algorithm, because it raises the probability of obtaining higher SAD values distant from the co-located block, as demonstrated in Section III.



*Statically Defined

Fig. 4. Scheme proposed for reducing the E-ADME depth maps complexity.

As shown in our previous work [6], using I-SDSP instead of TZS in ME of depth maps tends to obtain a good tradeoff between computational effort and encoding efficiency. However, as presented at the beginning of Section III, the ME process over depth map blocks in edge regions is harder than for homogeneous regions because many candidate blocks with low SAD values are found distant from the center-biased block and, in this case, I-SDSP can be inefficient. Considering these two aspects, the usage of TZS in edge blocks tends to obtain a sound tradeoff regarding encoding quality and computational effort. One can notice that the presented solution does not add any computational effort or an additional memory access to the 3D-HEVC depth maps inter-prediction.

The SED algorithm was previously presented in [8]. Its strategy classifies blocks according to their content characteristic. An analysis presented in [8] shows that the highest difference of the four corner samples (D_{max}) of an encoding block could be used to perform a classification into an edge or a homogeneous block with a high level of precision.

The efficiency of the SED algorithm is dependent on the threshold definition that should be defined statically according to the block size and the video resolution. In [8], the thresholds (TH) for classifying a block as homogeneous or edges were determined regarding the block size. In that work, only four blocks sizes should be used: 4×4, 8×8, 16×16 and 32×32, while ME also requires encoding 64×64 blocks, asymmetrical blocks such as 64×32, 32×24, 16×12, and others [15]. Thus, it is necessary to define new thresholds for those block sizes. The thresholds for the new block sizes were generated using the polynomial interpolation of Lagrange to make a second-degree polynomial equation, which computes the new thresholds based on the thresholds described in [8] and the correspondent block width. The new thresholds were generated for blocks with the widths of 12, 24, and 64 samples.

Equations 1 and 2 show the polynomials for 1024×768 and 1920×1088 video resolutions, respectively; W indicates the encoding block width and TH the resulting threshold. These polynomials for threshold generation allow classifying block sizes for future video standards that should be capable of using these blocks.

$$TH = ceil(-0.0186 \times W^2 + 2.2 \times W + 3.5) \qquad (1)$$

$$TH = ceil(-0.0038 \times W^2 + 0.74 \times W + 5.1) \qquad (2)$$

V. Results & Discussion

The E-ADME scheme has been inserted into the 3D-HTM version 16.0 and evaluated under Common Test Conditions (CTC) for 3D experiments in Random Access configuration [4]. In this evaluation, eight videos were used, considering four quantization parameters. In these videos, three texture views are encoded along with their associated depth maps. Then, the view synthesis software of 3D-HTM is applied to obtain six synthesized views, whose video quality is needed for displaying 3D videos and for verifying the real impact of depth maps coding since synthesized views are obtained from texture and depth maps data.

The results acquired in this evaluation are compared with the results obtained using the original 3D-HTM 16.0 (i.e., depth maps ME only using TZS algorithm for all encoding blocks). Table I depicts this comparison showing the encoding efficiency in synthesized views using the Bjontegaard Delta-rate (BD-rate) criterion and the timesaving obtained by the E-ADME scheme considering the entire encoder (texture and depth) and only depth maps coding.

The proposed technique reduces 6.9% the average time for depth maps coding and 3.2% in the whole encoder (that consider texture and depth coding execution times). As a drawback, E-ADME requires a BD-rate increase of 0.148%. The lowest BD-rate increase of 0.021% is noticed in *Balloons* video sequence because this video has low movement and edges and homogeneous areas are easy to be predicted in videos with low movement rates. The *Shark* video sequence presented a high BD-rate increase of 0.374% because this video has many movements and details, making the edges prediction difficult.

TABLE I. E-ADME results in CTC evaluation.

Video	BD-rate	Timesaving	
	Synthesized views	Texture and depth	Depth only
Balloons	0.021%	3.6%	7.7%
Kendo	0.025%	3.4%	7.6%
Newspaper_CC	0.106%	3.7%	7.2%
GT_Fly	0.095%	3.3%	7.5%
Poznan_Hall2	0.290%	3.2%	6.8%
Poznan_Street	0.071%	3.6%	7.3%
Undo_Dancer	0.204%	2.3%	5.4%
Shark	0.374%	2.9%	6.0%
Average	0.148%	3.2%	6.9%

Fig. 5 shows the percentage of SAD computation reduction obtained by E-ADME in comparison with the TZS implementation. One can notice that E-ADME guarantees more than 60% of SAD calculations reduction for all evaluated video sequences. On average, the use of E-ADME provided a reduction of 68.2% in SAD computations. Therefore, when analyzing the obtained tradeoff between SAD calculation and BD-rate, it is possible to see that E-ADME can considerably reduce the number of SAD computation of 3D-HTM 16.0 with negligible losses in the encoding efficiency. This reduction in SAD computations reduces the necessary accesses to the memory with reference samples and then, contributing to reduce the power dissipation. Consequently, the E-ADME scheme contributes to the development of real-time encoding systems with negligible impact on the encoding efficiency.

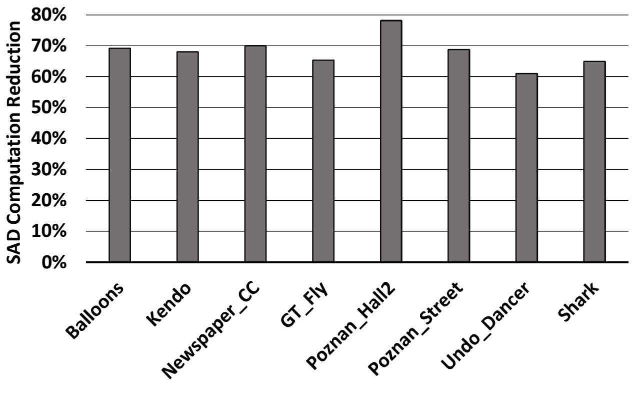


Fig. 5. SAD computation reduction obtained using E-ADME.

Furthermore, as described in [16], one of the biggest problems in ME algorithms relies on the fact that it has a high I/O communication with the reference memory. This I/O communication is strictly related with fetching information from previous reference frames to use in its SAD computation. Therefore, reducing SAD computation in 68.2% tends to reduce the memory access in the same proportion. Since the encoder bottleneck is the I/O communication with the memory [16], the proposed scheme tends to increase the performance of the entire system when implemented into a dedicated hardware design.

The solution provided in [6] was implemented in the 3D-HTM 16.0 (for a fair comparison because it was evaluated in 3D-HTM 10.2 in [6]). On average, E-ADME is capable of providing a reduction of 0.262% in BD-rate when compared to [6]. As a drawback, the solution requires 0.7% (in average) of increase in the encoding execution time, considering the whole encoder. If only depth maps are considered, E-ADME increases 1.5% the encoding execution time, in average.

In this comparison, our highest gain regarding BD-rate (0.597%) is noticed in Shark video sequence. As previously mentioned, *Shark* is a sequence with high movement and many details. In [6], edges were not well predicted since only information around the center-biased position was evaluated. Since edges are hard to predict in this kind of video, the proposed scheme that introduced the SED algorithm as an edge detector technique in its evaluation was capable of anticipating the edge regions inside the encoding frame and distributing the encoding complexity according to the difficulty to predict the blocks. Therefore, E-ADME has obtained a good tradeoff between the encoding efficiency and the provided complexity reduction.

VI. Conclusions

This paper presented an inter-frame prediction scheme for 3D-HEVC depth maps coding called Edge-Aware Depth Motion Estimation (E-ADME). This scheme aims to reduce the depth maps coding execution time while inserting minor reduction in the encoding efficiency. The main idea of the proposed scheme is early detecting edges or homogeneous areas by applying the Simplified Edge Detector (SED) algorithm and reducing the ME complexity for homogeneous blocks. Experimental results were obtained with 3D-HTM reference software, based on the 3D CTC for random access configuration. These results show that the proposed scheme saves 6.9% of the time in depth maps coding with a drawback of only 0.148% in BD-rate of the synthesized views. Moreover, the E-ADME scheme provides a notable reduction in the SAD computation of up to 79% with small impact in BD-rate performance compared with TZS.

Acknowledgment

This paper was achieved in cooperation with Hewlett-Packard Brazil Ltda. using incentives of Brazilian Informatics Law (Law nº 8.248 of 1991). Authors also would like to thank CAPES (processes 88881135737/2016-01 and 88881119481/2016-01), CNPq (processes 309707/2015-3 and 486136/2013-2) and FAPERGS (process 16/2551-0000241-0) Brazilian research agencies to support the development of this work.

References

[1] K. Muller, H. Schwarz, D. Marpe, C. Bartnik, S. Bosse, H. Brust, T. Hinz, H. Lakshman, P. Merkle, F. Rhee, G. Tech, M. Winken, T. Wiegand. "3D High-Efficiency Video Coding for Multi-View Video and Depth Data," *IEEE Transactions on Image Processing* (TIP), v. 22, n. 9, pp. 3366-3378, Sep. 2013.

[2] G. Tech, Y. Chen, K. Muller, J. Ohm, A. Vetro, Y. Wang. "Overview of the Multiview and 3D extensions of High Efficiency Video Coding," *IEEE Transactions on Circuits and Systems for Video Technology* (TCSVT), v. 26, n. 1, pp. 35-49, Jan. 2016.

[3] P. Kauff, N. Atzpadin, C. Fehn, M. Muller, O. Schreer, A. Smolic, R. Tanger. "Depth Map Creation and Image-based Rendering for Advanced 3DTV Services Providing Interoperability and Scalability," *Signal Processing: Image Communication* (SPIC), v. 22, n. 2, pp. 217-234, Feb. 2007.

[4] D. Rusanovskyy, K. Muller, A. Vetro. "Common Test Conditions of 3DV Core Experiments," *ISO/IEC JTC1/SC29/WG11 MPEG2011/N12745*, Geneva, Switzerland, Jan. 2013.

[5] M. Saldanha, G. Sanchez, B. Zatt, M. Porto, L. Agostini. "Complexity Reduction for 3D-HEVC Depth Maps Coding," *IEEE International Symposium on Circuits and Systems* (ISCAS), pp. 621-624, 2015.

[6] M. Saldanha, G. Sanchez, B. Zatt, M. Porto, L. Agostini. "Energy-Aware Scheme for the 3D-HEVC Depth Maps Prediction," *Journal of Real-Time Image Processing* (JRTIP), pp. 1-15, 2016.

[7] X. Tang, S. Dai, C. Cai. "An Analysis of TZSearch Algorithm in JMVC," *International Conference on Green Circuits and Systems* (ICGCS), pp. 516-520, 2010.

[8] G. Sanchez, M. Saldanha, G. Balota, B. Zatt, M. Porto, L. Agostini. "Complexity Reduction for 3D-HEVC Depth Maps Intra-frame Prediction using Simplified Edge Detector Algorithm," *International Conference on Image Processing* (ICIP), pp. 3209-3213, 2014.

[9] Cheng, Y. , et al. "An H.264 Spatio-Temporal Hierarchical Fast Motion Estimation Algorithm for High-Definition Video". *IEEE International Symposium on Circuits and Systems* (ISCAS) 2009. pp. 880-883.

[10] X.L. Tang, S.K. Dai, C.H. Cai, "An analysis of TZSearch Algorithm in JMVC", ICGCS, pp. 516-520, 2010.

[11] N. Purnachand, L. N. Alves, A. Navarro. "Fast Motion Estimation Algorithm for HEVC," *IEEE International Conference on Consumer Electronics - Berlin* (ICCE-Berlin), pp. 34-37, 2012.

[12] Z. Pan, Y. Zhang, S. Kwong, X. Wang, L. Xu. "Early Termination for TZSearch in HEVC Motion Estimation," *IEEE International Conference on Acoustics, Speech and Signal Processing* (ICASSP), pp. 1389-1393, 2013.

[13] Z.-T. Liao, C.-A. Shen. "A Novel Search Window Selection Scheme for the Motion Estimation of HEVC Systems," *International SoC Design Conference* (ISOCC), pp. 267-268, 2015.

[14] G. Sanchez, M. Porto, L. Agostini. "A Hardware Friedly Motion Estimation Algorithm for the Emergent HEVC Standard and its Low Power Hardware Design," *IEEE International Conference on Image Processing* (ICIP), pp. 1991-1994, 2013.

[15] G. Sullivan, J. Ohm, W. Han, T. Wiegand. "Overview of the High Efficiency Video Coding Standard," *IEEE Transactions on circuits and systems for video technology* (TCSVT), v. 22, n. 12, pp. 1649-1668, 2012.

[16] Jen-Chieh Tuan, Tian-Sheuan Chang, and Chein-Wei Jen, "On the data reuse and memory bandwidth analysis for full-search block-matching VLSI architecture," *IEEE Transactions on Circuits and Systems for Video Technology* (TCSVT), vol. 12, no. 1, pp. 61–72, Jan 2002.