Complexity Reduction of 3D-HEVC Based on Depth Analysis for Background and ROI Classification

Giovanni Avila, Ruhan Conceição, Thiago Bubolz, Bruno Zatt, Marcelo Porto, Luciano Agostini, Guilherme Correa
Video Technology Research Group (ViTech) – Federal University of Pelotas (UFPel) – Brazil
{gdadavila, radconceicao, tlabubolz, zatt, porto, agostini, gcorrea}@inf.ufpel.edu.br

***Abstract*—The 3D extension of the High Efficiency Video Coding (HEVC) standard achieves large compression rates thanks to the addition of several tools to encode multiview and depth information on top of those available in HEVC. The use of such tools incur in a very large computational demand, which can be a serious problem in power and computationally-constrained devices and applications. However, not all information contained in an image is fundamental to the viewer, so that different levels of computational effort can be employed when encoding different image regions. The Region of Interest (ROI) concept is used in this work to classify each Coding Unit (CU) as foreground, heterogeneous background and homogeneous background. Then, a simplified encoding process is employed in those regions classified as homogeneous background, terminating earlier the partitioning process in texture CUs, while still still performing the regular decisions in areas classified as ROI. Experimental results show an average reduction of 22.6% in computational complexity for texture coding with negligible or non-perceived image quality degradation.**

Keywords—Video Coding; 3D Video; 3D-HEVC; Complexity Reduction; Subjective Analysis; Region of Interest; Early Termination.

I. INTRODUCTION

Digital video content is present in the most diverse areas of our society nowadays and have major importance in our daily lives. In the last decades, playing, recording, storing and online streaming of 2D digital video content has become possible in several types of multimedia devices, including smartphones, tablets and personal computers, thanks to the development of several efficient video compression tools. More recently, a rise in 3D digital video content has also been perceived, stimulating the research for more efficient compression techniques that allow handling the huge amount of information present in such media.

To represent a 3D video, a different image must be shown to each eye, so that the perception of depth occurs. To accomplish that, there are different data formats to represent 3D video, such as the two-view Conventional Stereo Video (CSV) format, and formats based on the multi-view principle. The Multiview plus Depth (MVD) technology aggregates depth maps to the multiview concept. For each view in MVD, there is a texture image and an associated grayscale depth map image, where each pixel in the depth map represents the distance between the corresponding pixel in the texture image and the camera. One of the main advantages of MVD is the possibility of not actually capturing all views. By allying depth maps to their respective textures (regular 2D views), intermediate views (called synthesized views) can be rendered.

The state-of-the-art 3D video coding standard is the 3D-HEVC [1], which is an extension of the High Efficiency Video Coding (HEVC) standard released in 2015 by the Joint Collaborative Team on 3D Video Coding (JCT-3V). 3D-HEVC is able to reduce significantly the bitrate necessary to represent 3D videos in the MVD format thanks to the addition of several compression tools on top of those already available in HEVC. However, the HEVC encoder is by itself already 500% more complex i.e., it takes around 500% more time to encode the same video than its predecessor (the H.264/AVC standard) [2], so that this complexity issue becomes even more serious in 3D-HEVC. Due to the importance of the problem, recent works have proposed strategies to reduce the 3D-HEVC encoding complexity. In [3-4] the authors propose a set of solutions to reduce the encoding complexity of depth map images. In [5-7], the proposed approaches aim at reducing the encoding complexity of texture frames, whereas the authors in [8] propose a texture and depth mixed solution.

This work proposes a two-step adaptive complexity reduction scheme for texture coding in 3D-HEVC, which is based on information gathered during the encoding of depth map images. Relying on the idea of Region of Interest (ROI) and the concept of subjective video quality perception, the proposed scheme allows the encoder to identify background and foreground image areas and choose which of them must be encoded with higher or lower computational effort. By employing a strategy that chooses the low-complexity encoding areas according to their relevance to the viewer, the proposed method is able to reduce complexity by 22.6% in texture coding with insignificant loss in image quality, which was objectively and subjectively measured.

This paper is structured as follows. Section II introduces the concept of ROI and presents a pre-analysis that builds the basis for the proposed method. The two-step proposed method is explained in Section III. Sections IV and V present the experimental results and the conclusions, respectively.

This work has been sponsored by the Brazilian agencies CNPq, CAPES, and FAPERGS.

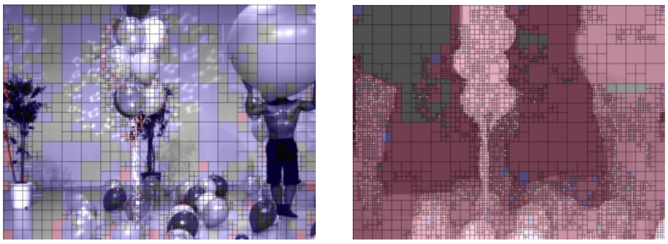
ISBN 978-0-9928626-7-1

II. Region of Interest and Statistical Analysis

The concept of ROI is employed in many works on image processing, such as [9-11]. A ROI corresponds to a certain area of an image or a scene that is of particular interest to the user or the system, such as the face of every person, an individual or a moving object [9,10]. Generally, these works use the ROI to extract relevant information for the viewer that is not interested in the remaining areas of the image (which can be discarded or processed in a different manner), such as in the case of surveillance and medical video processing.

In [11], the objects in a 3D scene that are closer to the camera are considered within the ROI in general purpose video applications. This assumption is employed in many 2D and 3D ROI-based strategies for information extraction, but it fails to detect regions that are located in the background and are still relevant to the viewer. As the next section shows, this work proposes an idea based on the concept of ROI to encode the most relevant image areas with different computational effort. However, in this work the depth map information of 3D scenes determines not only the image area position within the scene, but also whether it belongs to the ROI.

Considering that objects in the foreground of a general purpose 3D scene are usually more dynamic and present a more detailed texture than those at the background, it is expected that smaller partitions are used to encode them. Fig. 1 presents a texture and a depth map frame of one view of the *Balloons* video sequence, in which the Coding Unit (CU) partitioning was performed by the 3D-HEVC Test Model encoder (HTM) [12]. It is noticeable that those objects in the foreground are much more partitioned than those at the background, except for the cases in which there are abrupt variations in the background texture.



(a) (b)

Fig. 1. Coding Unit partitions in (a) texture and (b) depth map of one view of the *Balloons* video sequence.

The characteristics shown in Fig. 1 can be numerically seen in Table I, which presents the percentage of texture pixels belonging to each CU size within the foreground and background of each video. For the statistics calculation, foreground CUs are the 50% of CUs closer to the camera in a frame, whereas background CUs are the remaining 50% farther from the camera. The statistics presented in Table I are calculated considering the three original views of the eight video sequences that compose the 3D-HEVC Common Test Conditions (CTC) [13]. A quantization parameter (QP) equal to 30 for texture images and 39 for depth maps was employed in the tests.

Table I. Texture Pixels Distribution within Background and Foreground CUs

3D Video Sequence	Background				Foreground			
	64x64 (%)	**32x32 (%)**	**16x16 (%)**	**8x8 (%)**	**64x64 (%)**	**32x32 (%)**	**16x16 (%)**	**8x8 (%)**
Balloons	43.0	31.7	20.7	4.6	3.2	58.0	32.0	6.8
Kendo	50.4	25.7	18.0	5.9	3.4	47.9	35.4	13.3
Newspaper	42.0	32.0	19.8	6.2	22.2	42.6	27.2	8.0
GTFly	63.7	20.2	13.9	2.2	4.6	56.3	30.7	8.5
PoznanHall2	75.4	18.3	5.5	0.8	5.0	41.6	46.9	6.5
PoznanStreet	44.8	30.0	20.4	4.8	3.4	54.3	34.4	7.9
UndoDancer	22.3	31.6	35.1	10.9	38.2	32.7	19.1	10.1
Shark	31.3	39.6	21.7	7.4	3.2	62.8	26.0	8.0
Average	**46.6**	**28.7**	**19.4**	**5.4**	**10.4**	**49.5**	**31.4**	**8.6**

The table shows that the great majority of background texture pixels belong to large CUs. On average, 75% of all background pixels are encoded either as 64×64 or as 32×32 CUs, whereas only 5.4% belong to 8×8 CUs. Oppositely, foreground pixels are rarely encoded as 64×64 CUs, but are very often encoded in smaller sizes, especially 32×32 and 16×16. However, notice that in two sequences (*Shark* and *UndoDancer*) 64×64 CUs are not the majority in the background. In fact, the background pixels of *UndoDancer* were mostly encoded as 16×16 CUs. This shows that it is not possible to definitely assume that only the depth is enough to determine how partitioned a CU should be, even though background areas are on average less partitioned than the foreground. In some cases, the background is composed of multiple objects at different depths, so that it must be partitioned as much as the foreground.

III. Adaptive Two-Step Early Termination Based on Depth Information

Based on the analysis presented in the previous section, this work proposes a two-step complexity reduction strategy for 3D-HEVC texture coding. Firstly, an analysis based on average depth is performed over each depth map CU in order to classify the corresponding texture CU as foreground or background. Secondly, for each CU classified as background, a second analysis based on gradient depth is performed over the depth map CU to determine whether it belongs to a ROI. Finally, an adaptive early termination is applied to every CU classified as non-ROI background, avoiding further partitioning.

A. Foreground/Background Classification

The classification of each texture CU as background or foreground is based on the average depth of the corresponding depth CU, which is the CU located at the same position in the depth map image. As previously explained, pixel values in a depth map image represent the distance of the texture pixels to the camera. The larger the value is, the farther it is from the camera. As the encoder divides an image in CUs of variable sizes, it is possible that within the same CU certain regions are closer and others are farther to the camera, so that an average depth must be calculated to classify the CU. A sampled depth average, called here as *SA*, was computed based on the four corner samples (S_1, S_2, S_3, S_4) and the central sample (S_5) of each CU, as shown in Fig. 2(a). The value of *SA* is calculated separately for each CU and then compared to an adaptive

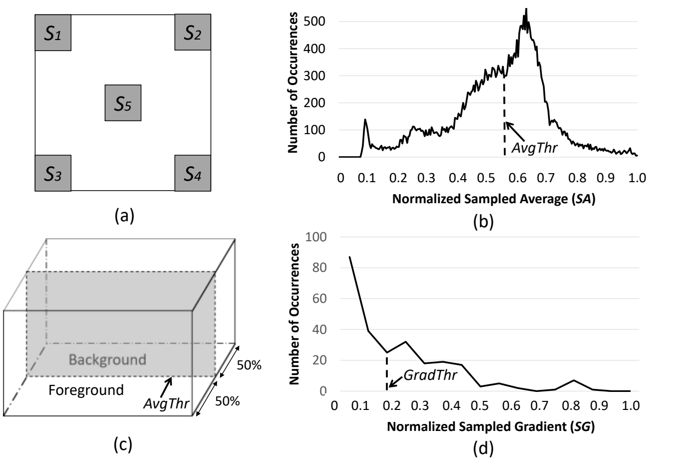


Fig. 2. (a) Pixels used for the calculation of *SA* and *SG* in depth CUs, (b) Distribution of *SA* in *Balloons* sequence, (c) Depth-based *AvgThr* threshold used to classify texture CUs as foreground or background, (d) Distribution of *SG* in *Balloons* sequence.

threshold *AvgThr*, determined as follows, in order to classify the CU as background or foreground.

To determine the threshold *AvgThr*, for each depth CU processed during the encoding of a depth map image, the value of *SA* is computed as explained in the previous paragraph. After that, the value of *AvgThr* is found by normalizing the *SA* of all CUs in the depth image according to the nearest and farthest ones and by selecting the median, i.e. the *SA* value at 50% of the distribution curve, as shown in (1). In (1), *n* represents the number of CUs in the depth frame, *Min* is the minimum function used to find the nearest depth and *Max* is the maximum function used to find the farthest depth to the camera. Fig. 2(b) illustrates an example of threshold calculation for the *Balloons* sequence.

$$AvgThr_{i=1..n} = Median_{i=1..n}\left(\frac{SA_i - Min_{j=1..n}(SA_j)}{Max_{k=1..n}(SA_k) - Min_{l=1..n}(SA_l)}\right) \quad (1)$$

By doing so, the 3D scene can be split into two regions – the foreground, composed of depth CUs with *SA* smaller than *AvgThr*, and the background, composed of depth CUs with *SA* larger than *AvgThr*, as shown in Fig. 2(c). Texture CUs are then classified as foreground or background, according to the classification of their corresponding depth CUs.

B. ROI/Non-ROI Classification

Based on the assumption that foreground information is more relevant to the viewer than background [11], every texture CU at the foreground is directly classified as belonging to a ROI in this work. Oppositely, texture CUs classified as background must be further analyzed in order to detect if they are relevant to the viewer. The analysis presented in section II showed that not all depth CUs at the background are composed of homogeneous areas, so that they cannot be always encoded as large CUs. This happens because the background is generally composed of multiple objects, each one in a different distance from the camera, creating borders in the depth map image. This way, a strategy is required to determine the heterogeneity of the depth CU.

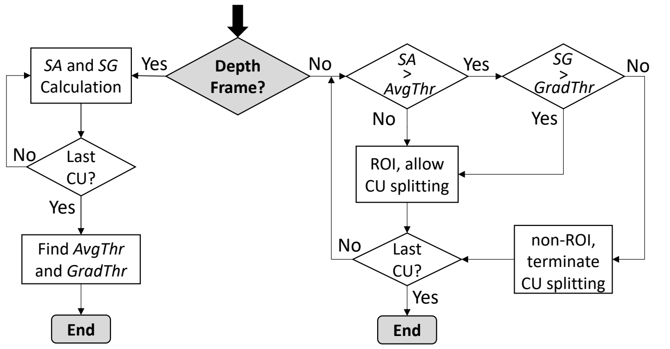


Fig. 3. Flowchart of the proposed early termination algorithm.

An idea similar to that presented in III.A is employed in the calculation of the depth CU gradient for each background CU in the depth map. The same five samples S_1-S_5 presented in Fig. 2(a) are used to calculate a sampled gradient (named here as *SG*). The value of *SG* is obtained simply by choosing the largest absolute difference between every pair of samples in S_1-S_5. Then, a threshold gradient *GradThr* is calculated according to (2) as the median of all the normalized *SG* values, just as previously described for *AvgThr*. As Fig. 2(d) shows, the median of the gradient distribution curve is usually a small value, which is expected because *SG* is only calculated for depth map CUs previously classified as background (generally composed of pixels with very similar values).

$$GradThr_{i=1..n} = Median_{i=1..n}\left(\frac{SG_i - Min_{j=1..n}(SG_j)}{Max_{k=1..n}(SG_k) - Min_{l=1..n}(SG_l)}\right) \quad (2)$$

During the depth encoding process, the *SG* value of each background CU is calculated. At the end, the *GradThr* value is derived. During the texture encoding process, every *SG* is compared to *GradThr*. If *SG* is equal to or larger than *GradThr*, the corresponding texture CU is classified as a ROI within the background. The remaining CUs are classified as non-ROI within the background.

C. Overall Complexity Reduction Algorithm

The proposed early termination takes advantage of the hierarchical block partitioning structure of 3D-HEVC. As each CU can be further split into four equal-sized CUs until the 8×8 size is reached, during the Rate-Distortion Optimization (RDO) process all possible splitting combinations are exhaustively tested and compared in terms of rate-distortion efficiency. In this work, the encoding computational complexity is reduced by identifying those situations in which this greedy solution is not needed or can be terminated without significant image quality degradation.

The flowchart in Fig. 3 presents an overview of the early termination algorithm proposed in this work. For each CU larger than 8×8, the algorithm is applied in two steps. First, the foreground and background classification is performed based on the average depth calculation, as explained in III.A and shown in the *SA* > *AvgThr* conditional of Fig. 3. Then, the background CUs are classified as ROI or non-ROI based on the depth gradient, as explained in III.B and shown in the *SG* > *GradThr* conditional of Fig. 3, which determines if that CU is in a ROI or a non-ROI area, thus deciding if the

ISBN 978-0-9928626-7-1 © EURASIP 2017

splitting can be early terminated. The average threshold calculation restricts dynamically the early termination to background CUs, whereas the gradient threshold calculation further restricts the application of the method to homogeneous background CUs, thus avoiding significant perceptual distortion as section IV will show.

While encoding the depth map image, for each CU, one position in an array *Avg* of 256 elements (one for each possible average, considering 8-bit samples) is incremented, according to the calculated *SA* of that CU. Similarly, one position in an array *Grad* of 256 elements is incremented, according to the *SG* calculated for that CU. At the end of the depth map coding, each position in both arrays represents the number of times that average or gradient value occurs in the depth map image. Thus, the thresholds can be derived and each depth CU can be classified as foreground, ROI background and non-ROI background.

While encoding the texture image, the co-located depth CU classification is used to early terminate the texture CU partitioning process. Foreground CUs and background CUs classified as ROI are encoded normally, following the RDO process. Oppositely, in background CUs classified as non-ROI the CU splitting process is terminated and the best splitting configuration found so far is chosen.

IV. Experimental Results and Comparisons

In order to evaluate the solution proposed in this work, a complexity and quality assessment was performed. Besides the encoding time reduction analysis, the evaluation aimed at identifying image quality degradation both objectively and subjectively. The subjective analysis is essential in this work, since it relies strongly on the idea that the viewer perceives quality loss differently in areas outside the ROI.

HTM (version 16.2) [12] was used to encode video sequences in the MVD format with the Random Access configuration, QPs 25, 30, 35, 40 for texture and QPs 34, 39, 42, 45 for depth maps, following the CTC specifications [13]. The eight video sequences listed in the CTC document were used in the tests, all of which were encoded using three regular and six synthesized views.

A. Complexity Reduction and Objective Quality Evaluation

Complexity reduction was calculated by measuring the encoding time of the modified HTM and comparing it to the encoding time of the original HTM. The objective quality was assessed in terms of Bjøntegaard Delta-PSNR (BD-PSNR), a metric that represents the average distortion in PSNR for a fixed bitrate. The BD-PSNR was calculated as the PSNR difference between the videos encoded with the modified and the original HTM. As the proposed method affects directly only the texture coding, the average BD-PSNR of all texture images (original and synthesized views) was calculated.

Table II shows that the proposed method achieves an average complexity reduction of 22.6% when considering only the texture encoding process of 3D-HEVC, whereas the overall encoding complexity is reduced in 12.8%. In the best case (*PoznanHall* sequence), the encoding time reduction achieved 35.4% and 19.9% when considering texture and total encoding process, respectively. The smallest complexity reduction was noticed for the *Newspaper* sequence, in which the proposed strategy reduced texture and total complexity in 13.5% and 6.9%, respectively. However, this was also the case in which the smallest quality degradation was noticed. The table also shows a very small objective quality degradation of 0.07 dB, on average. The largest objective quality loss is noticed for the *GTFly* video sequence (0.28 dB), which is a computer graphics video sequence composed mostly of background area. Nevertheless, as the next subsection shows, the objective quality degradation of *GTFly* does not lead to quality loss perception, since these losses are outside the ROI. Although redundant with the BD-PSNR information, for comparison purposes Table II also presents the average BD-rate for each sequence. The values were calculated according to the CTC as the average between the video PSNR/video bitrate (BD-rate of video0 + video1), the video PSNR/total bitrate (BD-rate of video0+video1+depth maps) and the synthesized PSNR/total bitrate (BD-rate of synthesized views) [13]. An average BD-rate increase of 2.3% was noticed considering the eight sequences.

TABLE II. Encoding Time Reduction and Objective Evaluation

3D Video Sequence	Texture ΔTime (%)	Total ΔTime (%)	Average BD-PSNR (dB)	Average BD-rate (%)
Balloons	23.8	14.1	-0.07	2.1
Kendo	21.0	11.7	-0.02	0.6
Newspaper	13.5	6.9	-0.02	0.4
GTFly	20.6	12.6	-0.28	8.4
PoznanHall2	35.4	19.9	-0.06	2.6
PoznanStreet	20.4	11.0	-0.02	0.5
UndoDancer	21.5	12.7	-0.05	1.9
Shark	24.8	13.8	-0.06	1.7
Average	**22.6**	**12.8**	**-0.07**	**2.3**

B. Subjective Quality Evaluation

The ITU-T Recommendation P.910 [14] was used as basis for the subjective test methodology employed in this work. The test was applied to 30 female and male volunteers between 18 and 44 years old, in an environment adjusted to allow viewing distance of 3 times the image height, viewing angle of 0º and low room illuminance (under 20 lux in front of the screen). The viewers watched all video sequences in a 46-inch Samsung 3D LED TV with active shutter glasses and registered their evaluations in a handheld device. The evaluation metric used in the tests was the Mean Opinion Score (MOS) with the five following possible grades: (5) Excellent, (4) Good, (3) Fair, (2) Poor, (1) Bad.

Each evaluator watched four times each sequence, as follows. First, (*i*) the original, uncompressed video sequence is presented. Then, (*ii*) the sequence encoded using the original, unmodified HTM encoder is presented and a rating is required to be registered for the second sequence within 10 seconds (using the first one as reference). After that, (*iii*) the original, uncompressed video is presented once again, and finally (*iv*) the sequence encoded using the modified HTM encoder is presented, which is also rated within 10 seconds (using the third sequence as reference).

Fig. 4 presents the subjective evaluation results for the eight video sequences. On average, the perceived quality degradation is around 0.14 MOS units when the proposed scheme is employed in the encoding process, which is very small when considering the full MOS range. The worst-case scenario was noticed for the *PoznanHall* sequence, which actually received low ratings even when the original HTM is used. Oppositely, the *Newspaper* sequence presented an unexpected result: for 76% of the viewers, the quality of the sequence encoded with the proposed scheme was equal to or better than the same sequence encoded by the original HTM, which resulted in a MOS increase of 0.33 units. Also, notice that a degradation of only 0.13 units was noticed for the *GTFly* sequence, which was the one with larger objective quality decrease (see Table II). This divergence between objective and subjective evaluations happens because *GTFly* is a computer graphics video sequence composed mostly of homogeneous background areas, which leads to the application of the early termination strategy more frequently. Differently from the viewers, the objective quality metrics does not differentiate between ROI and non-ROI areas, so that the PSNR loss does not correspond to the perceived degradation.

C. Comparison with Related Works

Since the finalization of 3D-HEVC, several works have been published aiming at reducing its complexity. To the best of the authors' knowledge, none of these works explore the subjective quality perception of 3D scenes to tackle complexity issues. Besides, there are no works that present subjective quality evaluation results for their proposed methods, which precludes any comparison between this work and related works in terms of perceptual quality.

Even though objective quality metrics are not completely fair mechanisms to evaluate the method proposed in this paper (which relies heavily on the concept of subjective quality perception), Table III presents a comparison with some of the best works found in the literature that focus on complexity reduction for 3D-HEVC texture coding. As separate results for texture encoding time are usually not shown in related works, the comparisons are limited to the overall encoding time reduction. Except for [5], Table III shows that this work presents the largest complexity reduction between all compared works. However, although [5] surpasses this work in terms of complexity reduction, its compression efficiency loss is much higher. The comparison in Table III shows that the proposed scheme is competitive with related works even when an objective evaluation is performed, which is not the focus of the strategy. Finally, it is important to notice that the scheme can be integrated with any other approach, increasing even more the level of complexity reduction.

V. Conclusions

This work presented an adaptive two-step complexity reduction scheme for the 3D-HEVC encoding process that takes advantage of depth map information to early terminate the splitting decisions of texture CUs outside the ROI. To determine whether a texture CU is within the ROI, each corresponding depth map CU is analyzed and classified as belonging to foreground, homogeneous background or heterogeneous background area. Only CUs within homogeneous background area are considered outside the ROI. Experimental results showed that the adaptive ROI-based solution proposed in this work decreases the texture encoding time in up to 35.4% and in 22.6% on average, with negligible or non-perceived image quality degradation, measured in accordance to subjective tests.

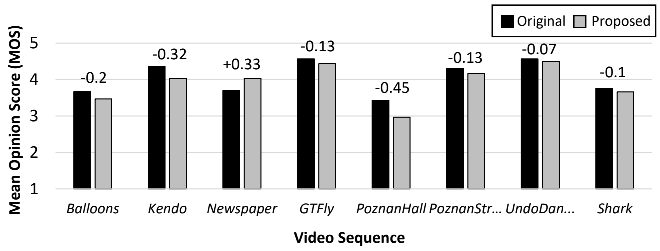


Fig. 4. Subjective quality evaluation results measured in MOS.

TABLE III. Comparison with Related Works

Work	Total ΔTime (%)	BD-Rate (%)
Zhang [4]	3.8	0.43
Tohidypour [5]	29.6	3.56
Zhang [6]	4.1	0.10
Song [7]	6.5	0.30
Proposed	**12.8**	**2.30**

References

[1] G. Tech, et al., "Overview of the Multiview and 3D Extensions of High Efficiency Video Coding", IEEE Transactions on Circuits and Systems for Video Technology, vol. 26, no. 1, pp. 35-49, Jan. 2016.

[2] G. Correa, et al., "Performance and Computational Complexity Assessment of High-Efficiency Video Encoders", IEEE Transactions on Circuits and Systems for Video Technology, vol. 22, no. 12, pp. 1899-1909, Dec. 2012.

[3] R. Conceição, et al., "Complexity reduction for 3D-HEVC depth map coding based on early Skip and early DIS scheme", 2016 IEEE International Conf. Image Processing (ICIP), pp. 1116-1120, Sept. 2016.

[4] Q. Zhang, et al., "Intra Mode Selection for Depth Map Coding in 3D-HEVC", Smart Computing Review, vol. 4, no. 5, 2014, pp. 360-370.

[5] H.R. Tohidypour, et al., "A Content Adaptive Complexity Reduction Scheme for HEVC-Based 3D Video Coding," in Proc. of 18th Int. Conf. on Digital Signal Processing, Fira, July 2013, pp. 1-5.

[6] Q. Zhang, et al., "Early SKIP mode decision for three-dimensional high efficiency video coding using spatial and interview correlations," J. Electronic Imaging, vol. 23, no. 5, 2014, pp. 53017-8.

[7] Y.-X. Song, K.-B. Jua, "Early Merge Mode Decision for Texture Coding in 3D-HEVC," Journal of Visual Communication and Image Representation archive, vol. 33, n. C, 2015, pp. 60-68.

[8] E. Mora, et al., "Initialization, Limitation, and Predictive Coding of the Depth and Texture Quadtree in 3D-HEVC", IEEE Trans. on Circuits and Systems for Video Tech., vol. 24, no. 9, pp. 1554-1565, Sept. 2015.

[9] M. Xu, et al., "Region-of-Interest Based Conversational HEVC Coding with Hierarchical Perception Model of Face", IEEE Journal of Selected Topics in Signal Processing, vol. 8, no. 3, pp. 475-489, June, 2014.

[10] P. Xing, et al. "Surveillance video coding with quadtree partition based ROI extraction", Picture Coding Symposium, pp. 157-160, Dec. 2013.

[11] L. Karlsson; M. Sjostrom, "Region-of-Interest 3D Video Coding Based on Depth Images", 2008 3DTV Conference: The True Vision - Capture, Transmission and Display of 3D Video, pp. 141-144, May, 2008.

[12] Y. Chen, et al., "Test Model 11 of 3D-HEVC and MV-HEVC", Document: KCT3v-K1003, Geneva, Switzerland, Feb. 2015.

[13] K. Müller, A. Vetro, "Common Test Conditions of 3DV Core Experiments", Document JCT3V-G1100, Jan. 2014.

[14] ITU-T Recommendation, "Subjective video quality assessment methods for multimedia applications", Document E33690, April, 2008.

ISBN 978-0-9928626-7-1 © EURASIP 2017