Low-Complexity HEVC Transrating Based on Prediction Unit Mode Inheritance

Matheus Lindino, Thiago Bubolz, Bruno Zatt, Daniel Palomino, Guilherme Correa
Video Technology Research Group (ViTech)
Graduate Program in Computing (PPGC)
Federal University of Pelotas (UFPel)
Pelotas, Brazil
{mclindino, tlabubolz, zatt, dpalomino, gcorrea}@inf.ufpel.edu.br

***Abstract*—Video transcoding for bit rate adaptation has become mandatory for over-the-top applications that deliver multimedia content in heterogeneous environments under different network conditions and user capabilities. As transcoding requires sequentially decoding and re-encoding the video bitstream, the computational cost involved in the process is too high, especially when considering current state-of-the-art codecs, such as the High Efficiency Video Coding (HEVC). This work presents a fast HEVC transcoder for bit rate adaptation based on Prediction Unit (PU) mode inheritance, which uses information gathered from the HEVC decoding process to accelerate PU mode decision in the re-encoding process. Experimental results show that the proposed method achieves an average transrating time reduction of 42% at the cost of a bitrate increase of 0.54%.**

***Index Terms*—HEVC, complexity reduction, transrating, transcoding.**

I. INTRODUCTION

With the recent popularization of over-the-top services for multimedia streaming, video transcoding for bitrate adaptation (also called transrating) has become mandatory, since such services are required to provide several versions of the same content to meet different user requirements and network capabilities. As transcoding operations usually require long processing times, the task is usually performed offline and the several bitstream versions of a video are stored in servers for future user requests. Often-accessed contents benefit from this strategy, since they are promptly available for users. However, rarely-accessed videos are also stored in the server in multiple versions, wasting valuable storage resources. Thus, transrating for videos seldom accessed could be performed on-the-fly, i.e., as they are requested.

High Efficiency Video Coding (HEVC) is the current state-of-the-art video coding standard, launched in 2013 by the Joint Collaborative Team on Video Coding (JCT-VC) [1]. HEVC reduces the bitrate of encoded videos by 40% on average [2] while keeping the same image quality of its predecessor, the H.264/AVC [3] standard. However, such compression rates are achieved with a significant increase in terms of computational effort, which can reach up to 500% in comparison to H.264/AVC [4].

The idea of reducing computational cost by inheriting information from previous encoding steps has been explored in previous works for both Coding Unit (CU) and Prediction Unit (PU) size decision. In [5]–[7] the authors reduce the HEVC encoding complexity by employing different strategies based on heuristics and machine learning algorithms to predict partition sizes and avoid unnecessary computations.

Recent works also propose different strategies to reduce HEVC transrating complexity, such as [8]–[10]. However, such solutions lack in some aspect: either they do not achieve significant time savings, or they introduce non-negligible losses in encoding efficiency. More recently, a frame partitioning inheritance approach was presented in [11] for fast CU size decision in HEVC transrating, outperforming previous works in terms of encoding efficiency and time savings. Although [11] focus on fast CU decisions, the remaining partitioning structures are not treated in this work, limiting the achievable levels of complexity reduction. In [12] and [13], the authors focus their techniques on both CU and PU decisions, but [12] is not applied to a transrating scenario and [13] generalizes a single rule for fast PU decision, which can be expanded for further complexity reduction.

This paper presents a fast HEVC transrating approach based on PU mode inheritance. The proposed statistical-based solution allows inheriting PU mode information from the reference bitstream (High Bitrate – HBR) aiming at accelerating PU mode decision during the re-encoding of the same video with lower bitrate (LBR). The solution can be employed jointly with any other approach for complexity reduction at CU level, thus providing additional complexity reduction. Experimental results show an average complexity reduction of 42% in comparison to the original HEVC transrating, at the cost of an average compression efficiency loss of 0.54%.

The paper is organized as follows: Section II presents an overview on HEVC partitioning structures and previous works on transrating complexity reduction. Section III presents a statistical analysis that led to the complexity reduction strategy proposed in section IV. Section V discusses the obtained experimental results and section VI concludes this paper.

II. PARTITIONING IN HEVC TRANSRATING

HEVC introduced flexible frame partitioning structures to achieve better coding efficiency for various types of content. Each frame is first partitioned into square Coding Tree Units

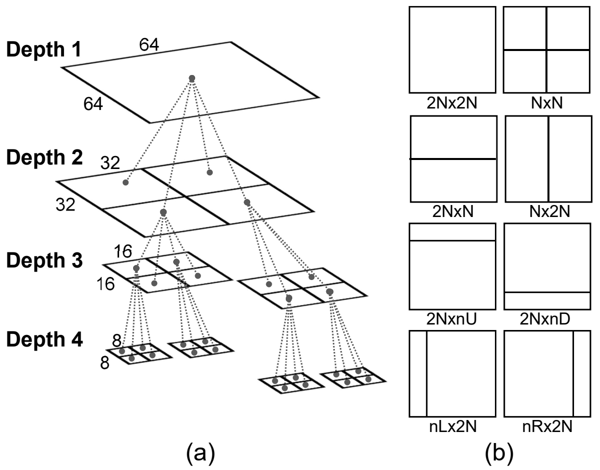


Fig. 1. HEVC partitioning structures: (a) CTU divided into CUs, (b) PU partitioning modes.

(CTUs), typically of 64×64 pixels. Then, each CTU is partitioned into one or more Coding Units (CUs) in a quadtree-based recursive process, in which a CU is split into four smaller CUs, until the minimum CU size (8×8 pixels) is reached, as shown in Fig. 1(a). For prediction purposes, each CU is divided into Prediction Units (PUs), which may assume eight different partitioning modes: 2N×2N, 2N×N, N×2N, 2N×nU, 2N×nD, nR×2N, nL×2N and N×N. When the CU size is 8×8, the PU can assume only one among four modes: 2N×2N, 2N×N, N×2N and N×N. The best partitioning is the one that yields the smallest rate-distortion (RD) cost. However, as the RD-cost requires prediction and residual coding to be computed, the computational complexity involved in this optimization process is extremely high [4].

When considering transcoding, computational complexity is even higher due to the cascaded decoding and re-encoding processes. However, information from the input bitstream can be used to guide the encoding decisions when re-encoding, especially when the same standard is used (e.g. in transrating).

In [8], the authors present a fast HEVC spatial re-scaling, which uses the number of CU partitions in the high-resolution video to limit the partitioning decision while encoding the low-resolution video. In [9], the authors trained a machine learning model using random forests to predict whether a CU should be divided or not based on the co-located blocks in the input video bitstream. The authors in [10] propose a CU early-termination solution based on three methods that use features such as motion vector, average depth and RD-cost of co-located CUs to speed up the transcoding process. A frame partitioning inheritance strategy was approached in [11], using information from the input bitstream to assist and streamline the CU decision process.

The method proposed in [12] aims at reducing transcoding complexity for spatially misaligned HEVC sequences in surveillance systems. In [12], information gathered from an input bitstream is used to halt CU and PU size decisions. The minimum CU size within a Coding Tree Unit (CTU) is predicted based on the mean CU sizes within the co-located region in the input bitstream. Prediction Units (PU) are also limited to 2N×2N if this is the observed partitioning mode in co-located blocks. In [13] the authors propose a complexity-scalable approach that operates both at the CU and PU levels to reduce transcoding time. The CU evaluation can be performed both on a bottom-up or top-down approach, allowing the encoder to abort recursion in early stages or to merge sub-blocks into larger ones, based on the correlation between CU depths in input and output streams.

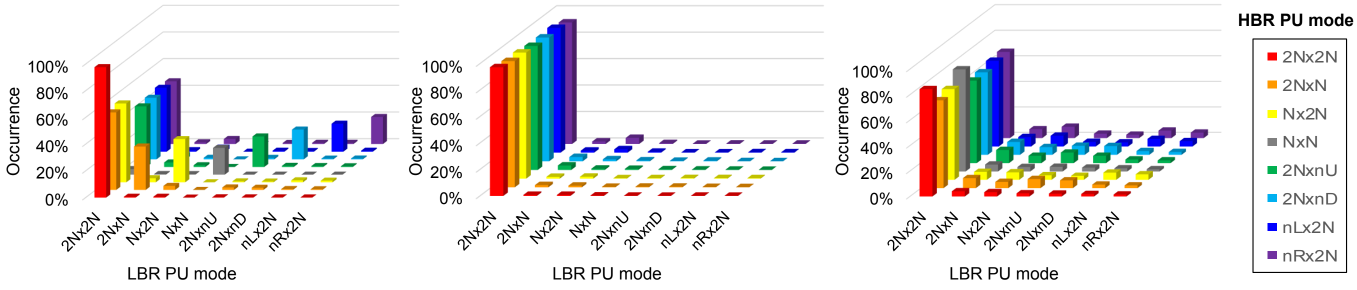
Even though the related works achieve interesting levels of complexity reduction, they focus on the inheritance of CU size or motion vector from the input bitstream. The only work that allows inheritance of PU modes along with CU size is [13]. However, the proposed strategy for PU early termination is based on a single rule based on one CU size and does not take into account all PU mode inheritance possibilities that different CU sizes would include.

III. PU MODE STATISTICAL DISTRIBUTION

This section presents a statistical analysis of the correlation between PU mode partitioning observed in High Bitrate (HBR) bitstreams and their transcoded versions to different Low Bitrate (LBR) values. For this analysis, the HEVC test Model (HM) reference software, version 16.14 [1], was used in all experiments. The settings defined in the Common Test Conditions (CTC) [14] document were followed and the Random Access Main HEVC encoder configuration was used. The following videos were used for the statistical evaluation: *BasketballDrive*, *BlowingBubbles*, *KristenAndSara*, *RaceHorses*, *Rollercoaster*, *SlideEditing*, *Traffic*, and *TrafficFlow*. These sequences were first encoded with the HM software and a Quantization Parameter (QP) set to 22, to guarantee an HBR bitstream with good image quality. Then, the transrating bit rates for each video were calculated as 80%, 60%, 40% and 20% of the bit rate obtained in the HBR encoding. This way, each video was transcoded four times, once for each LBR. The CU size and the PU partitioning mode observed both in the HBR decoding and in the LBR re-encodings were saved for the comparison in the analysis.

The three charts of Fig. 2 show average PU mode correlation between HBR and LBR considering the four transratings performed. The x-axis represents each PU mode chosen by the encoder during the re-encoding process (LBR bitstream), whereas the color scheme and the z-axis represent PU modes observed when decoding the original HBR. The y-axis in Fig. 2 corresponds to the average correlation between the partitioning modes observed in co-located image regions in the HBR and the LBR bitstreams.

The correlation analysis can be generalized in three different behaviors. Fig. 2(a) shows the case in which the co-located region is encoded with the same CU size in both LBR and HBR bitstreams. In this case, the most frequently chosen PU modes in the LBR encoding are either 2N×2N or the same mode observed in the HBR bitstream. Fig. 2(b) shows a different scenario, in which the HBR CU is transcoded to



(a) (b) (c)

Fig. 2. Average PU mode decision correlation when transrating a CU (a) to a CU of same size, (b) to a smaller CU, and (c) to a larger CU.

a smaller CU in the LBR bitstream. In this case, the vast majority of chosen PUs modes are 2N×2N. The last case, shown in Fig. 2(c), represents a HBR CU that is transcoded to a larger CU size in the LBR bitstream. Fig. 2(c) shows that although the distribution tends to 2N×2N PUs, there are significant occurrences in other modes that cannot be disregarded.

IV. Fast PU Mode Decision Algorithm

The transrating scheme proposed in this work was designed based on the statistical observations presented in the previous section. Fig. 3 shows the flowchart of the proposed algorithm implemented in the HEVC encoder. The algorithm input consists of a CU depth map and a PU mode map extracted while decoding each frame in the HBR bitstream. Both values are then used by the algorithm to limit PU partitioning mode decisions in LBR encoding.

The *depth* value in Fig. 3 indicates the current coding tree CU depth (i.e., during re-encoding), whereas *HBR_CUdepth* indicates the CU depth obtained from the CU depth map observed for the co-located image region in the HBR bitstream. The value of *HBR_PU* indicates the PU partitioning mode chosen for the corresponding CU in the HBR bitstream, which is also obtained from the PU mode map.

Based on the three distinct cases observed in section III, the proposed solution follows three rules, as shown in Fig. 3. The first test represents the scenario described in Fig. 2(a): if the current CU depth (*depth*) is the same observed in CU depth map obtained from the HBR bitstream (*HBR_CUdepth*), the only modes tested are 2N×2N and the exact same PU mode present in the PU mode map. Then, the usual RD optimization process takes place and the mode that leads to the lowest RD-cost between the two tested modes is chosen (*choose best mode*, in Fig. 3). The second test, based on the scenario shown in Fig. 2(b), applies when the current CU depth is bigger than the used in a co-located CU of the HBR bitstream. In this case, the algorithm chooses the 2N×2N mode directly, avoiding the test of any other partitioning modes. Finally, the third rule applies when the current CU depth is smaller than the co-located CU in the HBR bitstream. In this case, the best PU mode is chosen after testing all possibilities, since the occurrences are distributed among all partitioning modes, as shown in Fig. 2(c).

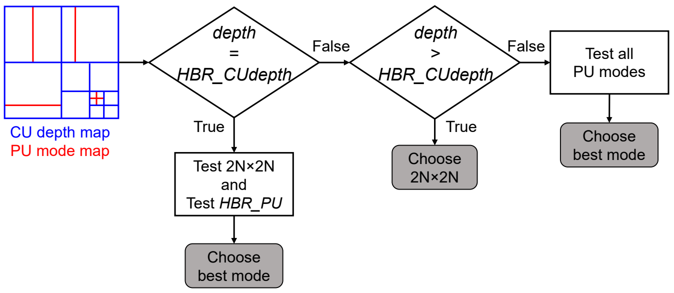


Fig. 3. Flowchart of the proposed algorithm.

V. Experimental Results

The same setup described in section III was used in the experiments to evaluate the proposed method, except for the video sequences. The testing sequences, also recommended in the CTC [14] document, are presented in Table I. In order to evaluate the proposed method in terms of compression efficiency and transrating time, all videos were first encoded with the original HEVC encoder without any modification. The 14 videos were encoded with a QP 22 and then re-encoded for the four LBR cases (80%, 60%, 40% and 20% of the HBR bit rate). Then, the same process was repeated with the HM software modified with the proposed algorithm for PU mode inheritance. Thus, the experimental results discussed in this section are comparative numbers between the proposed transrating strategy and the original transcoder.

Fig. 4 shows PU mode decisions in a cropped region of a B frame in the *Johnny* sequence. Fig. 4(a) and Fig. 4(b) show PU modes for the 80% LBR bitstream, which were decided by the original and the modified HM encoder, respectively. Similarly, Fig. 4(c) and Fig. 4(d) show the same decisions for the 40% LBR bitstream obtained by the original and the modified encoders. It is possible to notice that the two encoders decided for very similar partitioning, although the modified version evalutes significant less modes than the original.

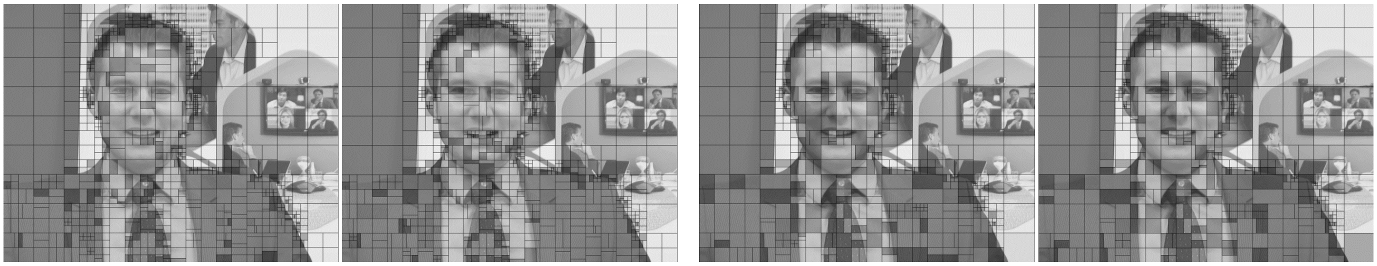


Fig. 4. PU partitioning for cropped *Johnny* sequence transcoded to (a) LBR 80% (original), (b) LBR 80% (proposed), (c) LBR 40% (original) and (d) LBR 40% (proposed).

A. Time Savings and Compression Efficiency

The experimental results in terms of time savings (TS) and compression efficiency are presented in Table I. TS results show that the strategy is able to significantly reduce transrating time, with an average reduction of 42% in comparison to the original transcoder. The best results in terms of TS are achieved for the *Johnny* sequence (54.33%), which is composed mainly of static, homogeneous areas – usually encoded with the largest CU size even in the HBR bitstream. Thus, in such case larger CUs are chosen more frequently in the LBR bitstream, leading the algorithm to test only the 2N×2N size.

Bjontegaard Delta (BD) metrics [15] were used to evaluate encoding efficiency. BD-rate is usually calculated based on the bitrate and the Peak Signal-to-Noise Ratio (PSNR) obtained when encoding video sequences under four different QPs. In this work, however, QP cannot be fixed during the full encoding process, since it is adjusted by the rate control algorithm to achieve the target bitrate when transrating. Thus, bitrate and PSNR values obtained when transrating to the four LBR ratios are used in this work to calculate BD-rate.

The obtained BD-rate results are also presented in Table I and they show that the proposed method results in an average compression efficiency loss of 0.54% in comparison to the original tandem transcoder. In general natural sequences with dynamic motion and complex texture are those that achieve the best compression efficiency results (e.g. *CampFireParty*, with a BD-rate of -0.29%), since in these cases the HBR bitstream presents more heterogeneous partitioning structures, leading to the test of all PU modes in the LBR encoding more frequently. On the other hand, these sequences are those that achieve the lowest TS values, since a larger number of computations are required to decide the best partitioning. The only exception is the *SlideShow* sequence, which shows good results both in terms of BD-rate (-0.66%) and TS (52.9%). This is a specific case composed of very homogeneous and almost identical frames. Notice that avoiding the test of smaller partitions in this case does not cause any decrease in compression efficiency and also reduces significantly the encoding time.

B. Combined Strategies for Fast Transrating

As this work follows a different approach from those found in the literature for transrating complexity reduction, which generally focus on the inheritance of motion information, prediction modes or Coding Unit (CU) size, it can be combined to any of them, thus providing further speed up to levels beyond the achieved separately by each one.

TABLE I
COMPRESSION EFFICIENCY AND COMPLEXITY REDUCTION RESULTS

Class	Videos	BD-rate (%)	TS (%)
A1	*Tango*	0.76	45.8
	CampfireParty	-0.29	35.0
A2	*CatRobot*	1.69	45.2
	DaylightRoad	1.09	41.5
B	*Kimono*	0.43	49.7
	ParkScene	0.53	40.6
C	*BasketballDrill*	0.76	35.9
	BQMall	0.48	38.1
D	*BasketballPass*	0.32	32.6
	BQSquare	-0.05	34.7
E	*FourPeople*	0.56	51.7
	Johnny	0.74	54.3
F	*ChinaSpeed*	1.18	32.9
	SlideShow	-0.66	52.9
All	**Average**	**0.54**	**42.0**

To show this, the strategy proposed by Bubolz [11] was selected to be implemented along with the proposed method. In [11], the authors propose an algorithm that halts the recursive search for the best CU size when the current depth reaches a CU depth threshold inherited from the HBR bitstream. Oppositely, if the current depth is below the HBR threshold, the CU splitting process is continued.

Table II presents results obtained by the combined algorithms. Notice that TS levels reach 68.7% on average in comparison with the original HEVC transcoder. Videos in classes E and F presented the best results in terms of TS, reaching a complexity reduction of up to 86.1% (*SlideShow*).

C. Comparison with Related Works

The four best related works found in the literature [8]–[11] were selected for comparison with the proposed method. To allow for fair comparisons, only the same video sequences used in related works are considered in the comparison (*BQTerrance*, *Cactus*, *Kimono* and *ParkScene*).

Table III shows that Praeter [9] reaches the second best time saving results in comparison to the original transcoder (61.3%). However, this comes at the cost of a considerable compression efficiency loss of 5.40% in terms of BD-rate. On the other hand, Schroeder [8] is the related work with the

TABLE II
COMPRESSION EFFICIENCY AND COMPLEXITY REDUCTION RESULTS – PROPOSED + [11]

Class	Videos	BD-rate (%)	TS (%)
A1	Tango	0.58	76.5
	CampfirteParty	0.34	59.9
A2	CatRobot	3.18	72.8
	DaylightRoad	-1.71	67.6
B	Kimono	1.49	80.6
	ParkScene	1.71	64.8
C	BasketballDrill	1.69	60.1
	BQMall	1.11	61.5
D	BasketballPass	0.94	53.3
	BQSquare	0.61	56.0
E	FourPeople	1.43	81.6
	Johnny	2.60	85.1
F	ChinaSpeed	1.60	56.2
	SlideShow	0.16	86.1
All	**Average**	**1.12**	**68.7**

TABLE III
COMPARISON WITH RELATED WORKS

	BD-rate (%)	TS (%)	BD/TS (x100)
Proposed	**0.65**	**40.9**	**1.583**
Bubolz [11]	0.88	45.4	1.934
Proposed + [11]	**1.58**	**64.3**	**2.453**
Schroeder [8]	0.76	37.5	2.025
Praeter [9]	5.40	61.3	8.806
Yang [10]	2.19	54.3	4.032

second best results in compression efficiency (loss of 0.76%), but time savings are limited to 37.5%.

The algorithm proposed in this paper achieves the best results in BD-rate, with a negligible decrease in compression efficiency of 0.65%. Considering time savings, a speed up of 40.9% is obtained with the proposed method. Aiming at a fair comparison between related works with different TS levels, the BD-rate/TS ratio is frequently used to measure the compression efficiency loss per time savings. The BD-rate/TS ratio is presented in Table III as BD/TS (multiplied by 100 for clarity). Notice that this work achieves the best tradeoff between encoding efficiency and complexity, with the smallest cost in BD-rate increase per each percent in TS.

When combining this work to Bubolz [11] for higher levels of TS, the experiments show that a complexity reduction of 64.3% is achieved, with BD-rate of 1.58%, still smaller than those achieved by other works that allow similar TS levels.

VI. CONCLUSIONS

This paper presented a method for reducing the HEVC transrating complexity based on PU partitioning inheritance. The proposed heuristic strategy is based on a statistical analysis performed over information gathered from High Bitrate (HBR) and Low Bitrate (LBR) bitstreams, which allowed identifying important correlation between PU partitioning modes decided for co-located image regions in both encodings. An early-termination algorithm based on three general rules was proposed, which uses information obtained from the HEVC decoding process to speed up the PU decision process during re-encoding. Experimental results show that an average transrating time reduction of 42% was achieved at the cost of a slight increase in BD-rate of only 0.54% in comparison to the original transcoder. Finally, the method can be combined with other strategies to achieve further complexity reduction. When combined with the best related work, average time savings reach 64.3%, with a small increase of 1.58% in BD-rate.

VII. ACKNOWLEDGEMENTS

This work was supported by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) – Finance Code 001, the Fundação de Amparo à Pesquisa do Estado do Rio Grande do Sul (FAPERGS) and the Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), Brazil.

REFERENCES

[1] High Efficiency Video Coding, “Recommendation itu-th. 265,”International Standard ISO/IEC, pp. 23008–2,2013.

[2] G. J. Sullivan, J. Ohm, W. Han and T. Wiegand, ”Overview of the High Efficiency Video Coding (HEVC) Standard,” in IEEE Transactions on Circuits and Systems for Video Technology, vol. 22, no. 12, pp. 1649-1668, Dec. 2012.

[3] T. Wiegand, “Draft itu-t recommendation and final draft international standard of joint video specification (itu-t rec. h. 264— iso/iec 14496-10 avc),” JVT- G050, 2003.

[4] G. Correa, P. Assuncao, L. Agostini and L. A. da Silva Cruz, ”Performance and Computational Complexity Assessment of High-Efficiency Video Encoders,” in IEEE Transactions on Circuits and Systems for Video Technology, vol. 22, no. 12, pp. 1899-1909, Dec. 2012.

[5] G. Correa, P. A. Assuncao, L. V. Agostini and L. A. da Silva Cruz, ”Fast HEVC Encoding Decisions Using Data Mining,” in IEEE Transactions on Circuits and Systems for Video Technology, vol. 25, no. 4, pp. 660-673, April 2015.

[6] K. Tai, M. Hsieh, M. Chen, C. Chen and C. Yeh, ”A Fast HEVC Encoding Method Using Depth Information of Collocated CUs and RD Cost Characteristics of PU Modes,” in IEEE Transactions on Broadcasting, vol. 63, no. 4, pp. 680-692, Dec. 2017.

[7] Y. Zhang, H. Wang and Z. Li, ”Fast Coding Unit Depth Decision Algorithm for Interframe Coding in HEVC,” 2013 Data Compression Conference, Snowbird, UT, 2013, pp. 53-62.

[8] D. Schroeder, A. Ilangovan, M. Reisslein and E. Steinbach, ”Efficient Multi-Rate Video Encoding for HEVC-Based Adaptive HTTP Streaming,” in IEEE Transactions on Circuits and Systems for Video Technology, vol. 28, no. 1, pp. 143-157, Jan. 2018.

[9] J. De Praeter et al., ”Fast simultaneous video encoder for adaptive streaming,” 2015 IEEE 17th International Workshop on Multimedia Signal Processing (MMSP), Xiamen, 2015, pp. 1-6.

[10] S. Yang and C. Zhong, ”Fast Coding-Unit Mode Decision for HEVC Transrating,” 2017 IEEE International Conference on Computer and Information Technology (CIT), Helsinki, 2017, pp. 93-100.

[11] T. Bubolz, R. Conceição, M. Grellert, B. Zatt, L. Agostini and G. Correa, ”Fast and energy-efficient HEVC transrating based on frame partitioning inheritance,” 2018 IEEE 9th Latin American Symposium on Circuits Systems (LASCAS), Puerto Vallarta, 2018, pp. 1-4.

[12] J. De Praeter, G. Van Wallendael, T. Vermeir, J. Slowack, and P. Lambert, “Spatially misaligned HEVC transcoding with computational-complexity scalability,” JOURNAL OF VISUAL COMMUNICATION AND IMAGE REPRESENTATION, vol. 40, no. A, pp. 149–158, 2016.

[13] L. Pham Van, J. De Praeter, G. Van Wallendael, S. Van Leuven, J. De Cock and R. Van de Walle, ”Efficient Bit Rate Transcoding for High Efficiency Video Coding,” in IEEE Transactions on Multimedia, vol. 18, no. 3, pp. 364-378, March 2016.

[14] K Sharman and C Rosewarne, “Common test conditions and software reference configurations for hevc,” in Proceedings of the Meeting of Joint Collaborative Team on Video Coding (JCT-VC) of ISO/IEC Z1100, 2017.

[15] Gisle Bjontegaard, “Calculation of average PSNR differences between RD-curves,” VCEG-M33, 2001.