Multistream Video Encoder for Generating Multiple Dynamic Range Bitstreams

Cedric Van Goethem, Johan De Praeter, Tom Paridaens, Glenn Van Wallendael, and Peter Lambert
Department of Electronics and Information Systems (ELIS)
Data Science Lab (Ghent University – iMinds), Ghent, Belgium
johan.depraeter@ugent.be, peter.lambert@ugent.be

***Abstract*—High-dynamic-range (HDR) technology allows capturing of video content at a wider range of luminance than low-dynamic-range (LDR) video. The resulting video more closely resembles the scene as perceived by the human eye. However, displays currently support only a limited range of HDR. Therefore, both an HDR version and LDR version of a video should be encoded during content acquisition. This means that the cost of encoder hardware in cameras would double. As a solution, this paper proposes a multistream video encoder that allows generating an HDR and LDR version of the same HDR video footage at roughly the same computational complexity as a single encoder, effectively allowing encoding of two dynamic-range versions of the video with a negligible increase in cost. For the LDR version, this multistream encoder results in a bit rate overhead of only 11.6% for the same quality as a two-encoder solution.**

I. Introduction

Advances in high-dynamic-range (HDR) technology allow video cameras to film content at a wider range of luminance than before [1]. This wider range allows for more details in very bright and very dark areas of the produced picture, which more closely resembles the scene as perceived by the human eye. Although cameras have already evolved to support HDR, the displays to view these videos are still evolving. Therefore, having a low-dynamic-range (LDR) version of the same video is beneficial for these displays.

To create an LDR version of an HDR video, the pixel values must be converted from HDR to LDR by using tone-mapping algorithms [2], [3]. This conversion can either happen in post-processing or on the camera itself. In the first case, the HDR video is first encoded on the camera and transported to the studio. This video is then decoded, tone-mapped, and re-encoded as LDR video. This method does not require extra storage capacity for the LDR version on the camera itself. However, due to the decoding and re-encoding process, the resulting video will have a lower quality than if the video were encoded directly from the original source. In the second case, the camera converts the HDR video to an LDR version and encodes the LDR video in addition to the HDR video. Compared to the first case, the quality of the LDR version will be higher, since the video is directly encoded from the source instead of being transcoded from an HDR version. However, since both an HDR and LDR version must be encoded, two encoders are needed in the camera, doubling the cost of the encoder hardware.

In order to prevent the doubled cost of encoder hardware, we propose a multistream encoder for high efficiency video coding (HEVC) that generates both an HDR and LDR version of the same video for roughly the same computational complexity as a single encoder. This is achieved by performing the rate-distortion optimization (RDO) process on the HDR version and only performing entropy encoding and calculation of the residual for the LDR version.

In Section II, we first give an overview of the computationally costly process of video encoding in HEVC. Section III then describes how related work attempts to reduce the complexity of this process. Next, the proposed method is described in Section IV, followed by the results in Section V. Finally, the conclusion is given in Section VI.

II. High Efficiency Video Coding

In this paper, HEVC [4] is used as the video compression standard since it is expected to be used by many HDR video applications and since it offers twice the compression efficiency for the same quality compared to its predecessor, H.264/AVC. However, this increased compression efficiency results in an increased computational cost. In this section, the coding tools that cause this computational cost are described.

In HEVC, each video frame is divided into Coding Tree Units (CTUs), which are typically blocks of 64×64 pixels. These blocks can be recursively split into Coding Units (CUs) according to a quadtree structure, with a smallest possible CU size of 8×8 pixels. Consequently, for each CTU of 64×64 pixels, the encoder has to determine the optimal combination of CUs out of 83522 possible quadtree structures.

Each CU can be either intra-predicted or inter-predicted. In the case of intra-prediction, the CU is assigned one out of two Prediction Unit (PU) partitioning modes, whereas inter-prediction allows a choice out of eight possible PU partitioning modes. Depending on the partitioning mode, the CU is thus split into one, two, or four different PUs.

Intra-predicted PUs have a choice between 35 possible intra-prediction modes for luma-components, and 5 possible modes (derived from the luma mode) for chroma-components. On the other hand, for each inter-predicted PU, the encoder performs motion estimation, which includes searching for the best motion vector in one or more reference frames and is the most computationally complex encoder operation. However, the encoder can also assign merge- or skip-mode to an inter-predicted PU, which means that the motion vectors of a spatially or temporally adjacent PU are copied. In the case of skip-mode, no residual is calculated for the PU.

Finally, each CU is assigned a residual quadtree (RQT). This tree signals the way in which a CU is split into Transform Units (TUs) for transformation and quantization of the residual picture. The smallest possible size of these TUs is 4×4 pixels.

For each of the above coding tools, the encoder has to determine the most compression efficient combination according to the RDO process. This process is the main cause of the encoder complexity. After the RDO process, the residual picture and coding information are supplied to an entropy encoder.

III. Related Work

Due to the high encoding complexity of HEVC caused by the many options that need to be evaluated by an encoder, much research has focused on limiting the amount of encoding decisions that are evaluated. Most of these works limit the CU decisions [5], since all other coding information is determined for each CU. Additionally, PU and TU options are often constrained as well [6]. Similarly, accelerations of other coding decisions such as intra-direction modes [7], motion estimation [8], and merge/skip mode [9] have been proposed as well. However, none of the above methods would be sufficient to make the computational complexity of an encoder negligible.

Besides fast encoding algorithms, multistream encoders have been proposed as well in HEVC. However, these encoders focus on providing different versions of a video at different bit rates [10], spatial resolutions [11], or a combination of both [12]. Since these works only accelerate CU decisions, the computational complexity of the resulting encoder is still significantly higher compared to the complexity of a single encoder.

Finally, a technique that does allow the reduction of the computational complexity of an encoder to that of a decoder has been proposed for transcoding in the form of control streams [13]. These control streams are defined as a regular video stream with the residual information removed. As such, these control streams contain coding information and can be stored on a server instead of the full bitstream. When the bitstream is needed, a high quality bitstream is fast-transcoded using the control stream. However, these control streams can only be created after first performing a non-accelerated transcode of the video and removing the residual.

IV. Proposed Method

Fig. 1 shows an overview of the proposed use of a multistream encoder in a camera for generating both an HDR and LDR bitstream. The camera first captures HDR footage. Based on this footage, an LDR source video is also generated in the camera by using an automatic color grading algorithm. Both the HDR and LDR source video are then supplied to the internal multistream encoder.

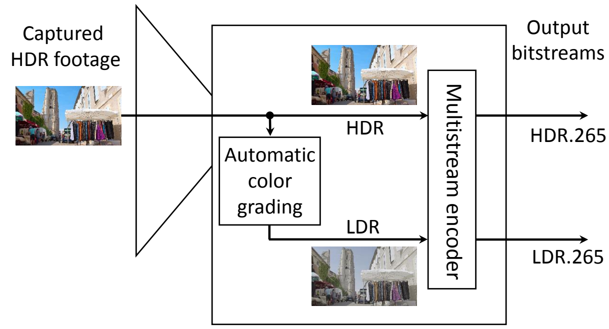

Fig. 1. Proposed system for simultaneous creation of an HDR and LDR bitstream from HDR footage.

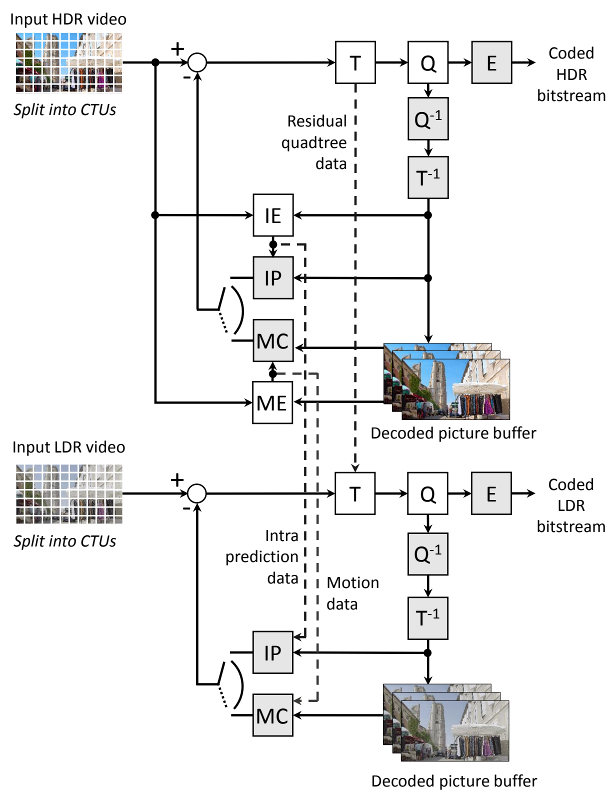

Fig. 2. Proposed multistream encoder architecture. Components for which an equivalent is also found in a typical decoder are colored gray. (T: transform; Q: quantization; E: entropy encode; IE: intra-picture estimation; IP: intra-picture prediction; MC: motion compensation; ME: motion estimation)

The proposed multistream encoder is shown in more detail in Fig. 2. In general, the multistream encoder behaves as a normal HEVC encoder for the input HDR video, with all possible decisions as described in Section II being evaluated. Each frame is split into CTUs for which a residual picture is transformed and quantized before entropy coding. In the encoder, this residual picture is also inversely quantized and transformed back to the pixel domain to be stored in a decoded picture buffer. This buffer is then used for motion estimation and motion compensation. Note that the inverse quantization and transformation, as well as intra-picture prediction and

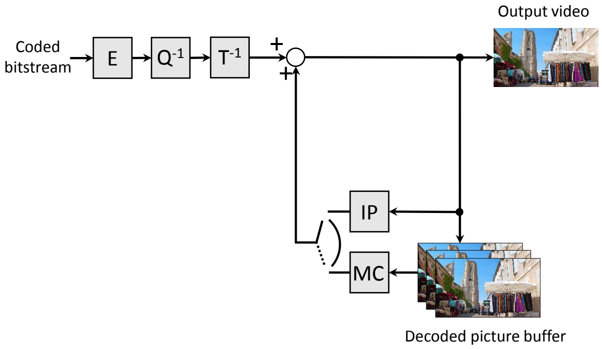

Fig. 3. Typical decoder architecture. (T: transform; Q: quantization; E: entropy decode; IP: intra-picture prediction; MC: motion compensation)

Fig. 4. First frame of each sequence used for the evaluation of the proposed multistream encoder.

motion compensation modules are also found in an HEVC decoder as well as in the encoder.

The LDR video, on the other hand, is processed in parallel with the HDR video. For each CTU in the HDR video, motion data, intra-prediction data, block structure information (not explicitly shown in the figure), and residual quadtree data (TU structure decisions) are copied to the LDR video. As a result, the intra-picture estimation and motion estimation modules of the encoding of the LDR version can be omitted. This means that, except for the transformation and quantization modules, the encoding modules of the LDR version resemble the structure of a decoder as shown in Fig. 3. Since the transformation step is also being accelerated with residual quadtree data, the complexity of encoding the LDR version in the multistream encoder has a similar complexity as an HEVC decoder. Consequently, the total complexity of the proposed multistream encoder is equal to that of a standard HEVC encoder plus an HEVC decoder.

V. Evaluation

In order to evaluate the proposed multistream encoder system, three HDR sequences were selected based on the availability of both an HDR version and an automatic color-graded LDR version. These video sequences were provided by the MPEG working group and consist of a street market with pedestrians (*market*), a night scene with fire breathers (*fire*), and hot air balloons slowly taking off while a crowd watches from below (*balloon*) [14]. The first frame of each of these sequences is shown in Fig. 4.

The sequences were encoded with different quantization parameter (QP) values using either the standard encoder included in the HEVC reference software (HM) version 16.5 [15], or by using the proposed multistream encoder similarly

TABLE I
Number of frames, frame rate in frames per second (fps), and used intra-period for each sequence.

Sequence	Frames	Frame rate (fps)	Intra-period
market	400	50	48
fire	200	25	24
balloon	240	24	24

based on HM 16.5. For all encodes, the 10-bit random access configuration *encoder_randomaccess_main10* was used. This configuration inserts an intra-frame at every intra-period, whereas the other frames consist of hierarchically bi-predicted inter-frames. This intra-period was chosen as approximately 1 second. The intra-period in frames for each video sequence is shown in Table I. Additionally, each encode in this paper was conducted on a single core of machines running on a dualsocket octa-core Intel Xeon Sandy Bridge (E5-2670) processor @ 2.6 GHz.

In the following subsections, the compression efficiency of the proposed multistream encoder is compared to that of a standard encoder. Next, the multistream encoder and standard encoder are also compared in terms of computational complexity.

A. Compression Efficiency

Since the encoding of the HDR version of the video is identical to a standard encoder, the focus of the evaluation of the multistream encoder in terms of compression efficiency lies on the encoding of the LDR version. The resulting LDR bitstream is thus evaluated in terms of Bjøntegaard delta rate (BD-rate) [16], which measures the bit rate overhead for the same quality compared to a standard encoding of the LDR version. The quality is measured as Peak Signal-to-Noise Ratio (PSNR).

In a first experiment, the effect of choosing different QP values for the quantization of the HDR and LDR version of the video was examined, since copying information from an HDR version encoded at the same QP as the LDR version might not necessarily achieve the most compression-efficient result. Therefore, the HDR version was encoded with HDR QP (QP_{HDR}) values of 22, 27, 32, and 37, whereas the LDR version was encoded with $QP_{LDR} = QP_{HDR} + \Delta QP$, with ΔQP varying from -4 to 5.

A rate-distortion (RD) curve for the *market* sequence with $\Delta QP = 2$ is shown in Fig. 5. Note that the RD-points for the standard encoder were obtained by encoding the LDR video with QP values of 22, 27, 32, and 37, hence the difference not only in bit rate, but also in quality. A full overview of the results is given as BD-rates in Fig. 6. As seen in this figure, the best results in terms of BD-rates are obtained when $\Delta QP = 2$, with an average BD-rate of 12.8%. This BD-rate is lower compared to the average BD-rate of 15.2% when using the same QP to encode both the HDR and LDR version, showing that both versions should not be encoded with the same QP when the bit rate overhead should be low.

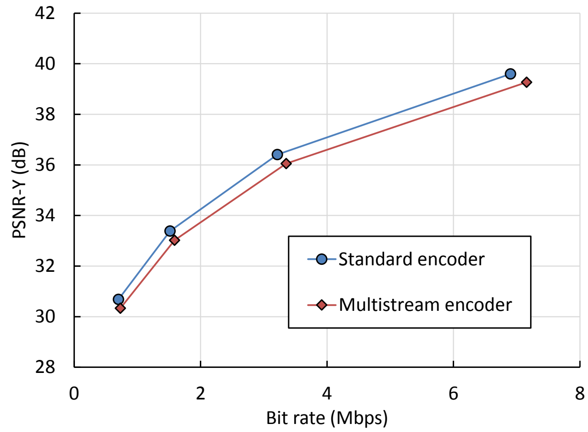

Fig. 5. RD-curve of the LDR version encoded with a standard encoder and with the multistream encoder, for the *market* sequence with $\Delta QP = 2$.

TABLE II
BIT RATES OF STANDARD ENCODINGS OF THE HDR AND LDR VERSION FOR EACH SEQUENCE. WHEN THE QP OF THE LDR VERSIONS DIFFERS FROM THE QP OF THE HDR VERSION BY 2, BOTH VERSIONS HAVE A SIMILAR BIT RATE.

HDR version				LDR version			
	Bit rate (kbps)				Bit rate (kbps)		
QP	Market	Fire	Balloon	QP	Market	Fire	Balloon
22	6726	1455	3757	24	6902	1405	3979
23	5729	1258	3253	25	5879	1211	3464
24	4900	1086	2829	26	5033	1048	3022
25	4194	938	2461	27	4320	909	2649
26	3597	808	2147	28	3714	786	2323
27	3092	698	1879	29	3215	680	2055
28	2661	603	1643	30	2778	585	1816
29	2293	519	1445	31	2387	503	1590
30	1973	445	1271	32	2050	433	1391

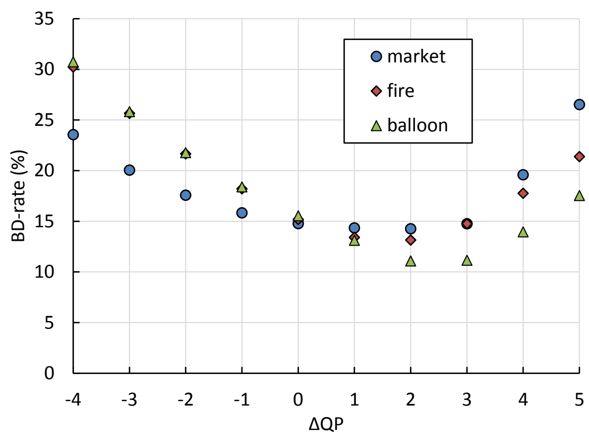

Fig. 6. BD-rate overhead (%) for the tested sequences when $QP_{LDR} = QP_{HDR} + \Delta QP$.

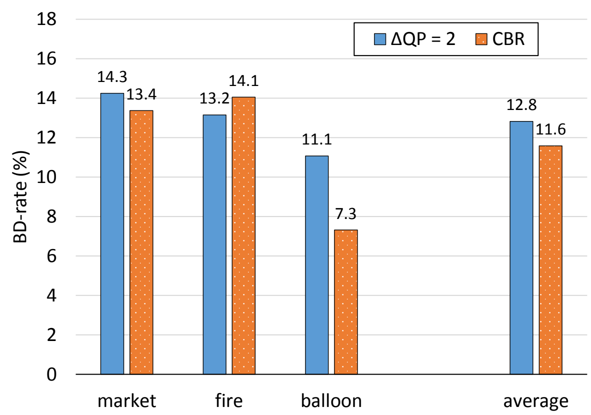

Fig. 7. BD-rate overhead (%) for a configuration with $\Delta QP = 2$ compared to a configuration using constant bit rate (CBR).

In order to further investigate the choice of QP, the bit rates of each sequence were retrieved when performing a standard encode on both the HDR and LDR versions of the video with a QP varying from 19 to 41. As seen for a subset of these values in Table II, when the QP of the LDR version differs from the QP of the HDR version by 2, the bit rates of the HDR version and LDR version are the most similar. For example, the bit rate of the Market sequence for a QP_{HDR} of 27 is 3092 kbps, which lies much closer to the bit rate when $QP_{LDR} = 29$ (3215 kbps) than when $QP_{LDR} = QP_{HDR} = 27$ (4320 kbps).

Since the above results appear to suggest that copying the coding information of the HDR version to the LDR version is more compression-efficient if both versions are encoded at a similar bit rate, a second experiment was conducted by encoding both versions at a constant bit rate. Both the LDR version and HDR version were thus encoded by enabling rate control in the HM software. This rate control was performed on a picture level, with equal bit rate allocation enabled. The target bit rates were set as the bit rates corresponding to QP_{HDR} values of 22, 27, 32, and 37.

The results of forcing equal bit rates for the HDR and LDR version of the video are shown in Fig. 7. When compared to a ΔQP of 2, constant bit rate appears to perform better on average with a BD-rate of 11.6% compared to a BD-rate of 12.8% for a constant QP. This indicates that it is indeed advisable to target a similar bit rate for the HDR and LDR version of the same video in order to have the smallest bit rate overhead for the same quality in the proposed encoder architecture.

B. Encoder Complexity

The proposed multistream encoder is conceptually similar in complexity to a standard encoder combined with a standard decoder as shown in Section IV. However, since in a practical scenario, the multistream encoder generating both an HDR and LDR bitstream would replace a standard encoder generating only an HDR bitstream (which can later on still be transcoded to an LDR bitstream), the multistream encoder is also compared to a standard encoder in terms of complexity. As an approximate measure of complexity, the encode time is used.

In the comparison, HDR bitstreams were encoded with the standard encoder at QP values of 22, 27, 32, and 37.

TABLE III
ENCODE TIME OF THE MULTISTREAM ENCODER COMPARED TO A STANDARD ENCODER THAT ONLY ENCODES AN HDR VERSION.

Sequence	QP_{HDR}	Standard encode time (s)	Multistream encode time (s)	Overhead
Market	22	117440	117908	1.0040
	27	112026	112406	1.0034
	32	108640	108966	1.0030
	37	106603	106897	1.0028
Fire	22	56725	56918	1.0034
	27	53670	53834	1.0031
	32	51453	51599	1.0028
	37	49996	50131	1.0027
Balloon	22	68419	68698	1.0041
	27	65325	65559	1.0036
	32	63015	63217	1.0032
	37	61145	61325	1.0029

These same values were used for the multistream encoder as QP_{HDR}, whereas $\Delta QP = 2$ was used for the LDR version. The complexity overhead of the multistream encoder was then calculated by dividing the multistream encode time by the standard encode time. The results are shown in Table III. In this table, it is seen that the overhead is always smaller than 1.0041, suggesting that the increase in encode time of the multistream encoder compared to a standard encoder is negligible.

In the above results, it should be noted that the encode times were obtained based on the HM reference software. Other encoders may implement encoding accelerations (e.g. by evaluating only a subset of the possible encoding decisions described in Section II), which would reduce the standard encode time. If the multistream encoder is also based on such an accelerated encoder, the multistream encode time will be reduced as well. However, since the motion estimation and intra-picture estimation modules in an accelerated encoder contribute less to the overall encoding complexity, the relative complexity overhead of the remaining blocks for encoding the LDR version is expected to be larger compared to using HM. However, even for faster encoders, the fact that the multistream encoder only has a combined complexity of an encoder and decoder still holds.

VI. CONCLUSION

In this paper, a multistream encoder architecture to generate a bitstream for both an HDR and LDR version of the same video was proposed. This multistream encoder has the same computational complexity as one standard HEVC encoder combined with one standard HEVC decoder. Compared to a standard encode, the bitstream of the LDR version has an average bit rate overhead of 11.6% for the same quality. In order to achieve the smallest bit rate overhead, both the HDR and LDR version should be encoded at a similar bit rate.

The proposed architecture can be implemented in cameras to automatically encode both an LDR version and HDR version from HDR footage at a computational cost of a single encoder, thus making the proposed system feasible for practical use.

ACKNOWLEDGMENT

The work in this paper was performed as part of the iMinds HD^{2}R (Highly Delightful Dynamic Range for Next Generation Digital Cinema and Television) project and was funded by the Data Science Lab (Ghent University – iMinds), Flanders Innovation & Entrepreneurship (VLAIO), the Fund for Scientific Research Flanders (FWO Flanders), and the European Union.

The computational resources (STEVIN Supercomputer Infrastructure) and services used in this work were kindly provided by Ghent University, the Flemish Supercomputer Center (VSC), the Hercules Foundation, and the Flemish Government department EWI.

REFERENCES

[1] R. Boitard, M. T. Pourazad, P. Nasiopoulos, and J. Slevinsky, "Demystifying High-Dynamic-Range Technology: A new evolution in digital media," *IEEE Consum. Electron. Mag.*, vol. 4, no. 4, pp. 72–86, Oct. 2015.

[2] K. Kim, J. Bae, and J. Kim, "Natural hdr image tone mapping based on retinex," *IEEE Trans. Consum. Electron.*, vol. 57, no. 4, pp. 1807–1814, Nov. 2011.

[3] B. Gu, W. Li, M. Zhu, and M. Wang, "Local Edge-Preserving Multiscale Decomposition for High Dynamic Range Image Tone Mapping," *IEEE Trans. Image Process.*, vol. 22, no. 1, pp. 70–79, Jan. 2013.

[4] G. Sullivan, J. Ohm, W.-J. Han, and T. Wiegand, "Overview of the High Efficiency Video Coding (HEVC) Standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1649–1668, Dec. 2012.

[5] H. S. Kim and R. H. Park, "Fast CU Partitioning Algorithm for HEVC Using an Online-Learning-Based Bayesian Decision Rule," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 26, no. 1, pp. 130–138, Jan. 2016.

[6] G. Correa, P. A. Assuncao, L. V. Agostini, and L. A. da Silva Cruz, "Fast HEVC Encoding Decisions Using Data Mining," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 25, no. 4, pp. 660–673, Apr. 2015.

[7] N. Hu and E. H. Yang, "Fast Mode Selection for HEVC Intra-Frame Coding With Entropy Coding Refinement Based on a Transparent Composite Model," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 25, no. 9, pp. 1521–1532, Sept. 2015.

[8] S. H. Yang, J. Z. Jiang, and H. J. Yang, "Fast motion estimation for HEVC with directional search," *Electron. Lett.*, vol. 50, no. 9, pp. 673–675, Apr. 2014.

[9] Q. Hu, X. Zhang, Z. Shi, and Z. Gao, "Neyman-Pearson-Based Early Mode Decision for HEVC Encoding," *IEEE Trans. Multimedia*, vol. 18, no. 3, pp. 379–391, Mar. 2016.

[10] D. Schroeder, P. Rehm, and E. Steinbach, "Block structure reuse for multi-rate high efficiency video coding," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sept. 2015, pp. 3972–3976.

[11] D. Schroeder, A. Ilangovan, and E. Steinbach, "Multi-rate encoding for HEVC-based adaptive HTTP streaming with multiple resolutions," in *Proc. IEEE Int. Workshop Multimedia Signal Process. (MMSP)*, Oct. 2015, pp. 1–6.

[12] J. De Praeter, A. J. Diaz-Honrubia, N. Van Kets, G. Van Wallendael, J. De Cock, P. Lambert, and R. Van de Walle, "Fast simultaneous video encoder for adaptive streaming," in *Proc. IEEE Int. Workshop Multimedia Signal Process. (MMSP)*, Oct. 2015, pp. 1–6.

[13] G. Van Wallendael, J. De Cock, and R. Van de Walle, "Fast transcoding for video delivery by means of a control stream," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sept. 2012, pp. 733–736.

[14] E. François, J. Sole, J. Ström, and P. Yin, "Common test conditions for HDR/WCG video coding experiments," ITU-T Joint Collaborative Team on Video Coding (JCT-VC), Tech. Rep. JCTVC-W1020, Feb. 2016.

[15] C. Rosewarne, B. Bross, M. Naccari, K. Sharman, and G. Sullivan, "High Efficiency Video Coding (HEVC) Test Model 16 (HM 16) Improved Encoder Description Update 2," ITU-T Joint Collaborative Team on Video Coding (JCT-VC), Tech. Rep. JCTVC-T1002, Feb. 2015.

[16] G. Bjøntegaard, "Calculation of average PSNR differences between RD-curves," ITU-T Video Coding Experts Group (VCEG), Tech. Rep. VCEG-M33, Apr. 2001.