General Video Coding Technology in Responses to the Joint Call for Proposals on Video Compression With Capability Beyond HEVC

Benjamin Bross, *Member, IEEE*, Kenneth Andersson, Max Bläser, Virginie Drugeon, Seung-Hwan Kim, Jani Lainema, Jingya Li, Shan Liu, Jens-Rainer Ohm, *Member, IEEE*, Gary J. Sullivan, *Fellow, IEEE*, and Ruoyang Yu

***Abstract*— After the development of the High-Efficiency Video Coding Standard (HEVC), ITU-T VCEG and ISO/IEC MPEG formed the Joint Video Exploration Team (JVET), which started exploring video coding technology with higher coding efficiency, including development of a Joint Exploration Model (JEM) algorithm and a corresponding software implementation. The technology explored in the last version of the JEM further increases the compression capabilities of the hybrid video coding approach by adding new tools, reaching up to 30% bit rate reduction compared to HEVC based on the Bjøntegaard delta bit rate (BD-rate) metric, and further improvement beyond that in terms of subjective visual quality. This provided enough evidence to issue a joint Call for Proposals (CfP) for a new standardization activity now known as Versatile Video Coding (VVC). All technology proposed in the responses to the CfP was based on the classic block-based hybrid video coding design, extending it by new elements of partitioning, intra- and inter-picture prediction, prediction signal filtering, transforms, quantization/scaling, entropy coding, and in-loop filtering. This article provides an overview of technology that was proposed in the responses to the CfP, with a focus on techniques that were not already explored in the JEM context.**

***Index Terms*— Video compression, standards, JVET, MPEG, VCEG, HEVC, VVC, H.265, H.266.**

Manuscript received March 17, 2019; revised September 19, 2019; accepted September 25, 2019. Date of publication October 25, 2019; date of current version May 5, 2020. This article was recommended by Associate Editor J. Chen. *(Corresponding author: Benjamin Bross.)*

B. Bross is with the Fraunhofer Institute for Telecommunications, Heinrich Hertz Institute, 10587 Berlin, Germany (e-mail: benjamin.bross@hhi.fraunhofer.de).

K. Andersson and R. Yu are with Ericsson Research, 164 83 Stockholm, Sweden.

M. Bläser and J.-R. Ohm are with the Institute of Communications Engineering, RWTH Aachen University, 52056 Aachen, Germany.

V. Drugeon is with Panasonic Business Support Europe, 63225 Langen, Germany (e-mail: virginie.drugeon@eu.panasonic.com).

S.-H. Kim is with LG Electronics, San Diego, CA 92131 USA (e-mail: seunghwan3.kim@lge.com).

J. Lainema is with Nokia Technologies, 33101 Tampere, Finland (e-mail: first.lastname@nokia.com).

J. Li is with Panasonic Research and Development Center Singapore, Singapore 469332.

S. Liu is with Tencent, Palo Alto, CA 94306 USA (e-mail: shanl@tencent.com).

G. J. Sullivan is with Microsoft Corporation, Redmond, WA 98052 USA (e-mail: garysull@microsoft.com).

Color versions of one or more of the figures in this article are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCSVT.2019.2949619

I. Introduction

EXPLORATION of video compression with capability beyond the High-Efficiency Video Coding Standard (HEVC, a.k.a. Rec. ITU-T H.265 and ISO/IEC 23008-2) started in 2015, two years after the first version of HEVC had been finalized. The activities on future video coding were conducted within the Joint Video Exploration Team (JVET) of the ITU-T Video Coding Experts Group (VCEG) and the ISO/IEC Moving Pictures Expert Group (MPEG), which developed a Joint Exploration Model (JEM) and its software implementation. The last version (JEM-7.0) achieved ∼30% bit rate reduction compared to the HEVC test model (HM) [1]. This was considered sufficient evidence by VCEG and MPEG to start a joint activity to develop a new video coding standard by issuing a Call for Proposals (CfP) in October 2017 [2].

Responses to the CfP from 32 organizations were submitted in three categories: standard dynamic range (SDR), high-dynamic range (HDR), and 360° omnidirectional video. Among these, the majority were based on the JEM software, two are based on a previously released alternative implementation of the JEM [3] and one is using a proprietary codec software codebase [4]. Consequently, the majority of CfP responses include technologies present and studied in the JEM or modified versions thereof. Moreover, all responses follow the same block-based hybrid video coding approach that is used in previous video coding standards like H.262/MPEG-2-Video, Advanced Video Coding (AVC) [5], or High Efficiency Video Coding (HEVC) [6]. The subjective evaluation results show that all CfP submissions were superior to HEVC in terms of subjective quality in most test cases and superior to the JEM in a relevant number of test cases [7].

This paper focuses on SDR video coding technology beyond the JEM, while submitted HDR and 360° technologies are reviewed and described in [8] and [9]. For a survey of the neural network-based technology submitted in responses to the CfP, the reader is referred to [10].

Estimates of coding efficiency benefit described in this paper are primarily based on the peak-signal-to-noise-ratio Bjøntegaard-delta bit rate (BD-rate) metric [11] as configured

for three use cases designed to fulfill the needs of particular applications: the random-access (RA), all-intra (AI), and low-delay bi-predictive (LB) configurations [2]. Among these, the most representative one for typical video coding applications is the RA configuration. As measured, BD-rate impacts are given as bit rate differences for equal peak-signal-to-noise ratio, such that negative numbers represent a coding efficiency benefit. Luma BD-rate measures are generally more meaningful, although chroma measures are sometimes reported and tend to be more difficult to interpret. The subjective testing performed for the CfP indicated that these measurements tend to actually be conservative – i.e., that there is more improvement of coding efficiency as measured by subjective quality than is reflected in the BD-rate measurements [7].

Since advances in compression beyond the JEM have been made in many building blocks of the hybrid video coding design, this paper is structured as follows. Section II covers newly proposed block partitioning approaches. Intra- and inter-picture prediction applied at the block level are reviewed and described in Sections III and IV while the different approaches of filtering the prediction signal are discussed in Section V. New approaches in transforms, quantization and scaling of the prediction residual are explained in Section VI, followed by the entropy coding of the quantized residuals with new aspects highlighted in Section VII. Finally, improvements of already known in-loop filters like deblocking and adaptive loop filtering, as well as a quick overview of newly introduced filters, such as neural network-based filters, is given in section VIII.

II. Block Partitioning

All CfP responses incorporate the general principle of initially dividing a picture into fixed-sized blocks of samples as basic processing units. As in HEVC and the JEM, such a basic unit will be further referred to as Coding Tree Unit (CTU). CTU sizes larger than the 64×64 maximum size supported in HEVC were generally proposed (typically up to 256×256). For the purpose of adapting the prediction blocks to local picture structures, the CTU is further partitioned into coding units (CUs) as the basic processing units for prediction with subsequent transforms and quantization of the prediction residual. This partitioning is using recursive tree structures, further referred to as coding trees, and is described in greater detail in the current section. Furthermore, a flexible processing order of the leaves of one tree node was introduced as split unit coding order (SUCO) as further detailed in [4]. At the end of this section, proposed fast encoder algorithms, employed to tackle the exponentially increasing search space of recursive coding trees with multiple partitioning options, are reviewed.

A. Multi-Type Tree

Multi-type tree (MTT) structures as were already proposed for the JEM have been used in most CfP responses [4], [12]–[16], [18]–[21]. The MTT (or QTBTTT) is a combination of a quadtree (QT) with nested binary trees (BT) as proposed for the JEM in [22] and triple- / ternary trees (TT). A CTU or CU is first partitioned recursively by a QT into square shaped blocks. Each QT leaf is then further partitioned by a BT or TT, where BT and TT splits can be applied recursively and interleaved but no further QT partitioning can be applied. In all relevant proposals, the TT splits a rectangular block vertically or horizontally into three blocks using a 1:2:1 ratio (thus avoiding non-power-of-two widths and heights). For partition emulation prevention, additional split constraints are typically imposed on the MTT to avoid duplicated partitions (e.g. prohibiting a vertical/horizontal binary split on the middle partition resulting from a vertical/horizontal ternary split). Further limitations are set to the maximum depth of the BT and TT. Proponents who measured the coding gain impact of MTT partitioning separately from other novel coding tools reported luma BD-rate changes of −3.26% [12], −2.3% [13], −2.36% [14] and −1.95% [15] in a tool-on test against JEM-7.0 using QTBT for the RA configuration.

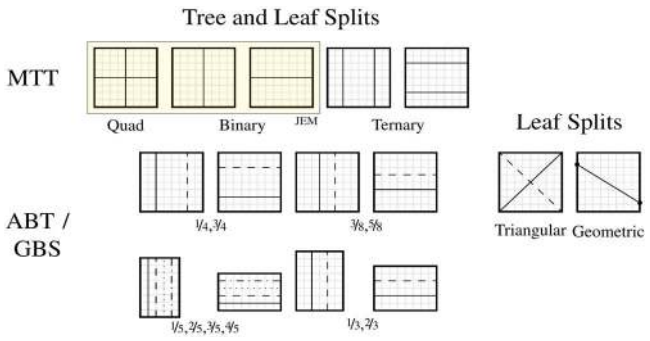


Fig. 1. Overview of the proposed block partitioning schemes: the MTT scheme is an extension of the QTBT partitioning of JEM, further adding a nested ternary tree.

B. Asymmetric Binary Trees and Generalized Binary Trees

Besides this static triple partitioning, more generalized variants have been proposed such as asymmetric binary tree (ABT) partitions [16] and generalized binary tree with shifts (GBS) [23]. The ABT concept adds asymmetric binary horizontal and vertical splits to the MTT using a 1:3 (or 3:1) ratio. Unlike HEVC, where such partitions are available as asymmetric motion partitions (AMP) for inter-coded leaf CUs, the ABT partitions can be further split recursively and can be used for inter- and intra-prediction. This concept is also applied in [18]. As ABT results in block sizes which may be multiples of 3×2^n with $n\geq1$ in each dimension, other parts of the codec design are also adapted to accommodate these new block shapes, such as additional separable primary transforms and transform coefficient entropy coding. Coding gain in terms of luma BD-rate change was reported as −3.2% for ABT+QTBT [17]. The GBS scheme further adds binary splits with ratios of 1:2, 2:3, 1:4 and 3:5 (and vice versa) to the BT split options. The resulting width and height of the CU must be multiples of 4 luma samples. An overview of the proposed partitioning schemes for tree and leaf splits can be seen in Fig. 1.

Additionally, ABS and GBS partitions were proposed as tree and leaf splitting options. For leaf nodes only, two non-rectangular splitting options were proposed (geometric and triangular for the triangle merge mode).

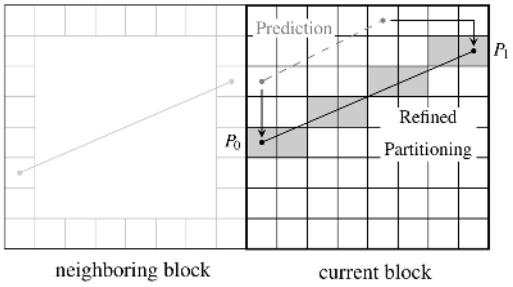


Fig. 2. Example of a geometrically partitioned block. The partitioning parameters P_0 and P_1 are obtained by spatial prediction from a neighboring block and refined by additional coordinate offsets.

C. Non-Rectangular Partitions

For the handling of the partitioning tree leaves, or coding units in HEVC and the JEM, additional geometric partitioning (GEO) for both inter and intra was proposed [24]. Previously, GEO had been investigated in the context of AVC [25] and not extensively investigated during the standardization of HEVC [26]. Apart from 2D video coding, wedgelet-based subblock partitions in combination with intra-prediction for the depth information is specified in the 3D extension of HEVC, uniquely suited for coding of sharp edges and homogenous areas, typically observed in depth maps. By segmenting the depth map of a coding block, implicit non-rectangular block partitioning for inter-picture prediction of dependent texture views can also be invoked in 3D-HEVC, using depth-based block partitioning (DBBP) [27].

GEO in the proposed variant of non-rectangular block partitioning allows the splitting of a QTBT leaf block by a straight line into two distinct segments. The partitioning line is parametrized by two points P_0 and P_1 which are located on the boundary of the current block, as illustrated in Fig. 2.

For block sizes of 16×16 luma samples and smaller, a look-up table with 16 pre-defined geometric partitioning shapes is used to indicate the partitioning. For larger block sizes of up to 128×128 luma samples, prediction of the partitioning line from spatially or temporally neighboring blocks can be employed. Finally, the geometric partitioning can optionally be further refined by signaling additional coordinate offsets.

As GEO in this proposal was implemented on top of JEM-7.0, inter-prediction is restricted to translational motion compensation with predictor-based signaling of motion vectors as already known from HEVC, e.g. the block merging mode and motion vector predictors. Intra-prediction of geometric segments is enabled using a modified version of planar-prediction only. The combination of both predicted segments (inter and/or intra) into a rectangular block is achieved by multiplication of the samples using two partitioning-dependent weighting masks with an overlapping area at the geometric partitioning line. This results in a smoothing effect similar to overlapped block motion compensation (OBMC).

Due to the lack of a transform tree in the JEM, a shape-adaptive DCT (SA-DCT) was added as an additional transform option for geometrically partitioned block residuals next to the DCT-II [28]. Overall, GEO was reported to achieve a luma BD-rate change of -0.79% for the RA configuration and -0.84% for the LB configuration.

Another variant of non-rectangular partitioning using triangular splits for inter-coded CUs was proposed in [15]. The triangle merge mode will be further described in Section IV-C.

D. Encoder Optimizations

One key aspect for all proposals, which increases the product space of splitting possibilities, is an appropriate fast encoder search method. A general approach for fast encoder decisions has been employed in [23] and is further detailed in [29]. Another proposal uses a convolutional neural network (CNN) to derive splitting decisions based on a texture analysis of 64×64 blocks [17].

III. Intra-Picture Prediction

When it comes to predicting the current block using only spatially neighboring samples from the same picture, the JEM extends the angular prediction in HEVC from 33 to 65 angles while using 1/32 sample accuracy for calculating the fractional sample reference values. For the fractional sample interpolation, the JEM replaces the HEVC bilinear filtering by a 4-tap interpolation filter. Furthermore, the JEM incorporates new methods of boundary prediction filtering, position dependent intra prediction combination with unfiltered reference samples, and cross-component, i.e. from luma to chroma, linear model sample prediction. For details on the JEM intra-picture prediction, the reader is referred to [1].

In this section, proposed intra sample prediction and mode coding tools beyond the JEM are reviewed and described in greater detail.

A. Wide Angle Prediction

The 67 spatial intra prediction modes in the JEM included DC prediction, planar prediction and 65 directional modes. The directional modes were covering a 180-degree range of angles extending from the bottom-left diagonal to the top-right diagonal direction. It was noted in a CfP response [30] that when predicting non-square blocks from the "narrow side" of the block, the reference samples for the intra prediction process tend to be relatively far away from the locations of the predicted samples and a similar directional texture could be created using closer reference samples if the opposite prediction direction was used. This phenomenon is illustrated in Fig. 3.

Motivated by this finding, it was proposed in [30] to define wide-angle alternatives for the 10 prediction directions closest to the bottom-left diagonal when the height of the block was smaller than the width, and the 10 prediction directions closest to top-right diagonal when the width of the block was smaller than the height. These wide-angle modes were specified using the inverse of the prediction directions used in the JEM software.

Signaling of the intra prediction mode followed that of the JEM, with the exception that if the indicated intra prediction direction had a corresponding wide-angle alternative, a 1-bin

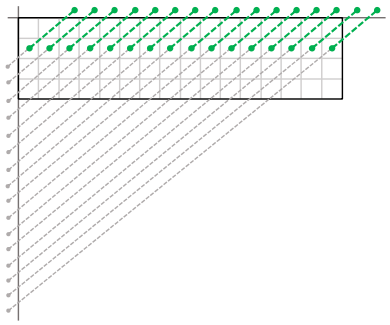


Fig. 3. Illustration of predicting one line of samples using a traditional prediction angle from the left reference (deviating less than 45 degrees from the horizontal direction) and with a wide-angle alternative from the above reference (deviating more than 45 degrees from the vertical direction).

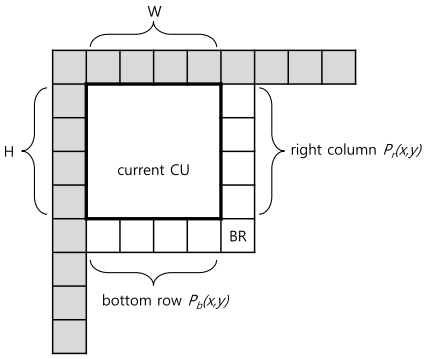


Fig. 4. Bottom right (BR), bottom row, and right column reference samples.

identifier determined which one was to be applied for the block. It was reported in the follow-up core experiment [31] that in AI configuration the proposed tool provided a BD-rate impact of -0.28%, -0.38%, -0.42% for the luma and the two chroma components, respectively.

B. Multiple Reference Lines

Several CfP responses use multi-reference lines, i.e. the possibility to predict from reference samples that are not directly neighboring to the current block. In case of discontinuities in the picture, the correlation between the samples to be predicted and already reconstructed samples that are one or more samples away may be higher than the correlation between the samples to be predicted and the directly neighboring samples. Two basic approaches have been suggested. One approach performs a (weighted) average of different reference lines [32], [33], [16] and the other approach allows the encoder to choose between several lines and signal the reference line used for prediction [20], [23].

C. Line-Based Prediction

Line-based intra coding is based on the same principle as what had been known as short-distance intra prediction, which relies on the principle that the correlation between samples increases with decreasing spatial distance. Since the prediction is performed by lines, and a line needs to be reconstructed before predicting the next line, one-dimensional transforms are introduced. In addition, the prediction direction together with the processing order of transform blocks can be changed depending on the intra prediction mode and the splitting direction. In terms of coding efficiency, line-based intra prediction as proposed in [23] achieves -1.2% luma BD-rate for RA. More details on line-based intra prediction can be found in [23].

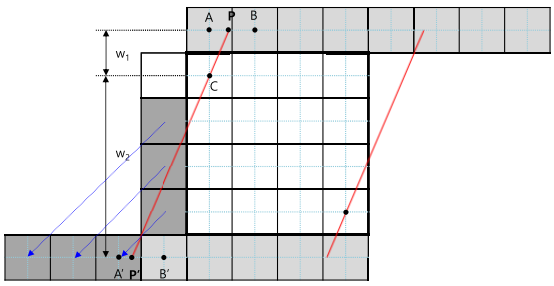


Fig. 5. Sample prediction process for linear weighted bi-intra-prediction (LWP).

D. Bi-Prediction

The HEVC (and JEM) intra prediction generates the predicted samples by extrapolating from the neighboring reconstructed samples along a certain direction. This directional intra prediction is based on the assumption that the texture in a region is directional, which means the sample values have a strong correlation and change smoothly along a specific direction. Thus, for smooth region, a directional intra prediction can work well. However, sample values may change substantially for high-detailed regions, and a fixed direction cannot describe them well. A new approach to improve intra prediction is to generate prediction samples by combining two prediction samples from two opposite directions. This was proposed as linear weighted bi-prediction [13] and distance weighted directional intra [4]. Both are combining two prediction samples from two opposite directions. The linear weighted bi-prediction approach introduced in [13] is depicted in Fig. 5. It first computes the intensity value of the bottom right (BR) neighboring sample based on distance weighted sum between two existing prediction samples as follows:

$$BR(W,H) = \frac{W \times R(2W-1,-1) + H \times R(-1,2H-1)}{(W+H)}$$

where, $R(x, y)$ denotes the intensity value of the reconstructed neighboring sample at (x, y) coordinate. W and H are the block width and height. Then, it generates the bottom row ($P_b(x, y)$) and the right column ($P_r(x, y)$) of the neighboring samples shown in Fig. 4. These samples serve as references on the opposite side of the CU and are calculated as follows:

$$P_b(x,H) = \frac{(W-1-x)\times R(-1,H) + (x+1)\times BR(W,H)}{W}$$

$$P_r(W,y) = \frac{(H-1-y)\times R(W,-1) + (y+1)\times BR(W,H)}{H}$$

In linear weighted bi-prediction approach, intra prediction is performed by applying linear weighted sum between the conventional reference sample, which is usually located above or left, and the opposite side reference sample (sample from bottom row or right column). Fig. 5 illustrates the detailed sample prediction process using linear weighted bi-prediction

approach for an arbitrary angular prediction mode. The red line represents the prediction direction. The points A and B represent the samples at the integer sample positions and the point P indicates the sample at 1/32 sample position in the reference sample array. The points $A\prime$ and $B\prime$ represent the samples at the integer sample positions and the point $P\prime$ indicates the sample at 1/32 sample position in the opposite side reference sample array.

The first stage of the intra angular mode prediction involves extrapolating samples from a projected reference sample location based on a given direction. The projected reference sample location is computed with four-tap intra interpolation filter which is used to obtain the value of the first projected reference sample P using four closest reference samples located at integer position. The second stage in the prediction process requires an identical approach to generate a second predicted sample value $P\prime$ at the same sample location. The second predicted sample value is obtained by projecting the location of the sample onto the opposite side of the reference sample array by applying the selected prediction direction and interpolating the value to 1/32 sample position accuracy. Then bilinear interpolation is performed linearly using the two closest opposite side reference samples in the direction of prediction. The final predicted sample value is calculated as

$$C = \frac{(W_1 \times P + W_2 \times P^{'})}{(W_1 + W_2)}$$

where, C is the final predicted sample value, P and P are the predictors generated from the reference sample location and the opposite side reference sample location, respectively. W_1 and W_2 are weighting factors determined according to the distance between the current sample position and the reference sample position and between the current sample position and the opposite side reference sample position, respectively.

E. Intra Block Copy

The basic concept of Intra block copy (IBC) mode is similar to conventional block-based motion compensation, with differences mainly in two aspects: 1) the IBC reference blocks are located in the reconstructed area within the same (current) picture; 2) the displacement vectors (referred as block vectors) are constrained to make sure only the already reconstructed samples may be referenced. In some implementations the block vectors are further restricted to be in integer precision to reduce complexity. This mode was first proposed during the standardization of H.264/AVC [34], as a coding tool to deal with natural, camera-captured content. During the standardization of HEVC extensions, this was proposed again in [35] and was found to be very efficient for coding screen content (i.e., computer-generated content) due to its frequently duplicated textures and patterns. Hence, it was adopted into the HEVC screen content coding extensions (SCC), where the decoded part of the current picture prior to in-loop filtering operations is treated as a reference picture and put into the reference picture list [36]. By doing this, the operations of the IBC mode and the regular inter prediction mode (motion compensation) are unified. At coding block level, when a reference index pointing to the current picture is selected, the IBC mode is used.

Due to the emerging market needs for screen content coding tools, IBC was proposed again in [20] and [37] as response to the VVC CfP. Several differences were introduced compared with the HEVC SCC IBC. First, IBC is treated as a third mode, on top of intra and inter prediction modes and is signaled at the coding block (unit) level. A block vector in integer resolution is used to indicate the displacement from the current block to a reference block inside the current picture. In addition, the IBC mode is only involved in uni-prediction and is enabled for blocks with at least one side containing fewer than or equal to 16 luma samples. The IBC block vectors are derived or constructed as follows:

- If a block is coded in merge mode, a merge candidate index is used to indicate which of the block vectors in the candidate list from neighboring IBC coded blocks will be used to predict the current block. Similar to HEVC, the IBC merge candidate list consists of five spatial neighboring blocks, if IBC coded.
- If a block is coded as a non-merge mode, the block vector difference is coded in the same way as a motion vector difference. The block vector prediction method uses two candidates as predictors, one from the left neighbor and one from the above neighbor, if IBC coded. When a neighbor is not available or not IBC coded, a default block vector is used as a predictor. A 1-bin flag is signaled to indicate the block vector predictor choice. The block vector predictor and block vector difference add up to the block vector.

Another new feature, introduced in the CfP response [20] and further extended in [37], is the use of IBC in the separate-tree structure. In a separate-tree structure, the prediction and coding partition trees for luma and chroma components are independent of each other. As a result, different parts of the same chroma CU may correspond to collocated luma samples which belong to different luma CUs. These luma CUs may or may not all be coded in IBC mode; or, even if they are all coded in IBC mode, they may contain different block vectors. A straightforward and efficient solution was used: a chroma CU may be coded in IBC mode only when all the luma samples in the collocated luma area for this chroma CU are coded in IBC mode. For each 2×2 chroma subblock in the chroma CU, its chroma block vector is derived from the luma vector of the collocated luma samples.

According to the simulation results provided in the response [20], by enabling IBC, about -0.3% BD-rate impact can be achieved in the RA configuration for VVC CfP test sequences, and about -37% BD-rate impact can be achieved in the RA configuration for screen content in the text and graphics test set used in HEVC SCC development. No apparent runtime increase was observed.

F. Intra Region-Based Template Matching

The principle of IBC, which requires signaling the motion vector, was extended by a decoder-side motion vector search with intra region-based template matching. This method

divides the reconstructed samples into different template search regions and signals only the index to the region where the motion vector is searched at the decoder to reduce the decoder complexity. Intra region-based template matching provides −1.1% luma BD-rate impact for RA in the CfP response as reported and further detailed in [23].

G. Decoder-Side Intra Mode Derivation

HEVC and the JEM both employ most probable mode (MPM) based intra mode coding where a list of MPMs is generated from spatially neighboring and default modes with various numbers of modes in the MPM list, i.e. three in HEVC and six in the JEM. In one CfP response, a decoder-side intra mode derivation is proposed that derives the mode at the decoder from neighboring reconstructed samples. The coding efficiency reported for the CfP response is −0.4% luma BD-rate impact for RA with further details provided in [12].

H. Neural Network-Based Prediction and Mode Coding

A novel approach for intra prediction and mode coding that is based on neural networks was proposed in one CfP response. This technique generates the prediction samples using a neural network with two lines of reference samples as input. In addition, another neural network is used in intra mode coding to determine the most probable mode. In the CfP response, neural network-based intra prediction achieves −1.8% luma BD-rate for RA. Further details can be found in [23].

IV. Inter-Picture Prediction

The JEM includes a variety of inter-picture prediction technology beyond HEVC such as higher motion vector storage and compensation precision up to 1/16 of a sample, adaptive motion vector difference resolution, local illumination compensation, overlapped block motion compensation, subblock-based temporal merging candidate, affine motion compensation, decoder-side mode vector derivation using template matching, pattern refinement or bi-directional optical flow [1]. Consequently, most CfP responses used modified versions of these tools mostly targeting simplifications without sacrificing coding efficiency. The tools presented in greater detail subsequently were not part of the JEM.

A. Affine Motion Compensation Flexing

The block-based affine motion compensation in the JEM applies affine motion model on PU level to calculate translational motion vectors for each 4×4 luma subblock within the PU. A motion vector field generated this way obviously has some discontinuity at the 4×4 block boundaries. Fig. 6 illustrates operation of block-based affine motion compensation for four blocks of samples when motion parameters indicate presence of relatively strong rotational motion.

The affine flexing tool, as proposed in a CfP response [30], addresses the block boundary discontinuity issue by applying line-based remapping or displacement of samples after the block-based affine motion compensation. Based on whether the nature of motion is more rotational or zooming, the method

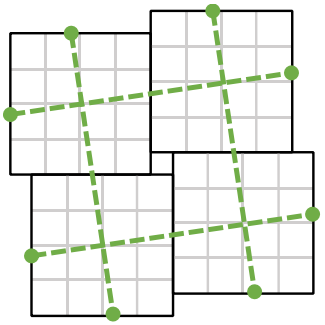


Fig. 6. Illustration of reference blocks for block-based affine motion compensation when motion parameters indicate rotational motion.

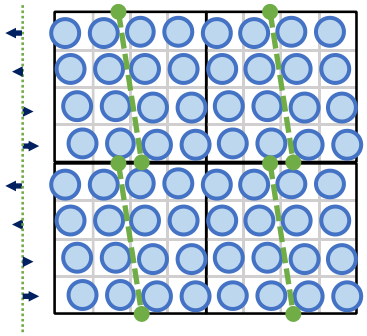


Fig. 7. Illustration of affine flexing in the horizontal direction with line-based lateral remapping offsets.

decides to apply either lateral compensation (moving the subblock boundary samples laterally with respect to the samples in neighboring subblock) or perpendicular compensation (moving the subblock boundary samples away or closer to the samples of the neighboring subblock), respectively. The remapping offsets applied to different lines are calculated based on motion vector difference between subblocks and the distance from the subblock border. Fig. 7 is illustrating the proposed operation in horizontal direction. A similar operation would also be applied in vertical direction in the case of rotational motion.

Performance of affine flexing was evaluated in a core experiment set up after the evaluation of the CfP responses [38]. It was reported that for RA test conditions and the sequences defined in the JVET common test conditions, the proposed tool provided −0.33% and −0.49% BD-rate impact when affine motion compensation used 4×4 and 8×8 subblocks, respectively. It was further reported that the coding efficiency impact was significantly higher for a test set containing affine-like motion characteristics. For that extended test set, the BD-rate impact for RA configuration was reported to be −1.79% and −4.21% for 4×4 and 8×8 subblock-based motion compensation, respectively.

B. Combined Intra- and Inter-Picture Prediction

Inter picture prediction and motion compensation have shown great effectiveness in video compression. On the other hand, intra prediction tends to preserve more accuracies which may be non-uniformly distributed inside a prediction block. Typically, more accurate prediction is observed when a sample in the current block is closer to reference samples. When a

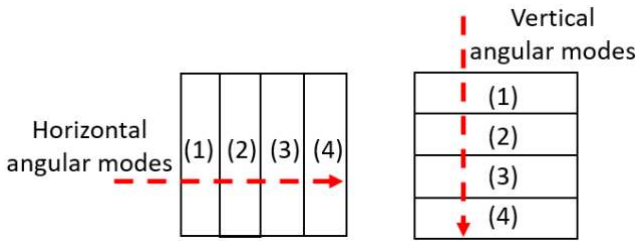


Fig. 8. Weighing assignments for regions with horizontal (left) and vertical (right) intra prediction modes.

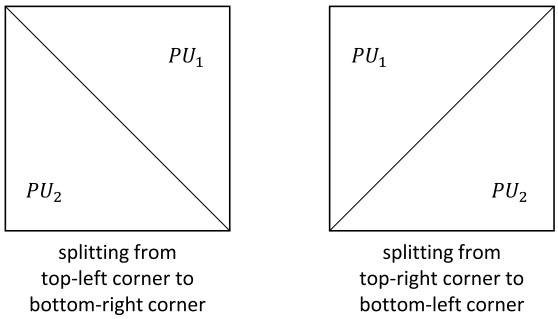


Fig. 9. Two possible partitionings of a CU into two triangle shaped prediction units.

sample is far away from reference samples, statistically larger prediction error would be expected. In order to take advantage of both prediction types, a combined intra/inter prediction mode was firstly proposed in [39] based on an H.264/AVC-like platform. In this new prediction mode, a weighted combination of intra- and inter-picture predictions can be utilized for predicting inter 16 × 16, 16× 8, 8×16, 8 × 8 ref0, 8 × 8, intra 4 × 4 and four intra 16 × 16 macroblock prediction modes. The sum of prediction weights is equal to 1, such that only the weight for the inter-coded predictor is signaled in the bitstream. The same weighting parameters are used for the combined intra-/inter-picture chroma prediction.

This combined intra/inter prediction method was refined and proposed in [14]. In this proposal, the combined intra/inter prediction mode is enabled at the coding block level while taking into consideration both coding efficiency and signaling overhead. A merge assistant prediction (MAP) flag is signaled at the block level to indicate the usage of this prediction method. In an inter-coded CU, the MAP flag is signaled when the merge mode flag is true. If this combined prediction mode is used, an intra mode index is additionally signaled to indicate the selected intra mode from an intra mode candidate list. The intra mode candidate list consists of 19 intra prediction modes including DC, PLANAR, and 17 angular modes which are uniformly subsampled from 65 angular modes. In an intra coded CU, the MAP flag is signaled for luma component. If this prediction mode is used, a merge index is additionally signaled to indicate the selected merge mode from a merge candidate list. The merge candidate list is derived from merge modes without subblock-based candidates (e.g., affine or temporal), and the merge candidate list size is set to 4. In both of these two signaling schemes, DM mode is always used for chroma prediction, thus no additional signaling is required for processing chroma components.

The weighting between intra and inter prediction signals depends on both the selected intra prediction mode and the block size. When DC or PLANAR mode is selected or the CU width or height is smaller than 4, equal weights are applied to intra and inter predictors. In other cases, angular prediction modes are first divided into two groups by evaluating if the mode is closer to the horizontal or vertical direction. Then, different weights are applied to intra and inter predictions from region (1) to (4) as shown in Fig. 8. For regions (1) and (2), which are closer to spatial neighboring reference samples, the weight for inter prediction is smaller than that for intra prediction. For regions (3) and (4), which are relatively far away from the neighboring reference samples, the weight for inter prediction is greater than that for intra prediction. Specifically, the weights for (intra, inter) combination in regions (1) through (4) are (3/4, 1/4), (5/8, 3/8), (3/8, 5/8), (1/4, 3/4), respectively.

According to subsequent CE test results after the evaluation of the CfP responses [40], this method can achieve about −0.7% BD-rate impact for the RA configuration with 10% encoder runtime increase.

C. Triangle Merge Mode

Triangle merge mode is an inter-picture prediction technology proposed in CfP response [15]. It is based on an idea previously presented in [42]. The basic idea consists in splitting a CU into two triangle shaped prediction units separated by a diagonal from the top-left corner to the bottom-right corner of the CU, or from the bottom-left corner to the top-right corner of the CU, as illustrated in Fig. 9. Each triangle is assigned different motion vectors and reference picture indices. While only one of two types of triangle partitioning is allowed to be selected for a given CU, this technology enables to represent many different diagonal directions when used at leaf nodes of a flexible rectangular partitioning structure, such as MTT partitioning. Since the triangle shaped partitions are only applied to predict the signal, transform and quantization are applied to the entire CU consisting of two triangle shapes, therefore reusing the rectangular shaped residual coding framework already introduced for MTT partitioning.

The triangle merge mode shares similarities with geometric partitioning presented in section II-C of the present paper, with the major difference that a lot of the flexibility of the geometric partitioning is removed to save overhead. Similar although more flexible technologies have also been introduced in previous video codecs, such as compound wedge prediction [43]. One main difference between the triangle merge mode and the compound wedge prediction is the way the motion vectors for the two different regions of the block are signaled and derived, since motion vector differences can be signaled for wedge prediction, but only one index from a list of motion vectors prediction candidates is allowed for the triangle merge mode. In addition, while wedge prediction allows a triangular kind of partitioning, this is only allowed for non-square blocks, whereas triangle merge mode can also be applied to square blocks.

One major difference of the technology proposed in [15] compared to the idea in [42] is that triangle shaped prediction

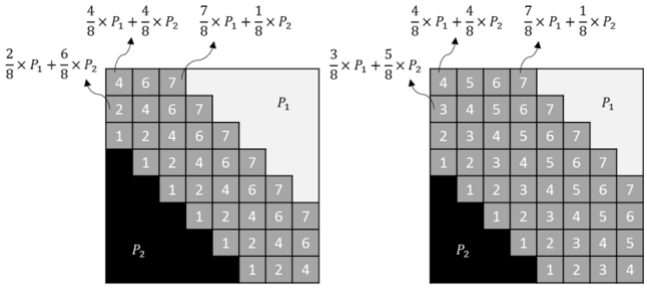


Fig. 10. Two sets of weights to combine the two triangle predictions for luminance samples.

TABLE I
INDEX BINARIZATION SCHEMES

Index	Weight value	Binarization Schemes
0	-1/4	0000
1	3/8	001
2	1/2	1
3	5/8	01
4	5/4	0001

units are only applied to CU coded in merge or skip modes. It saves encoding time and some signaling overhead by reusing the motion information of the neighboring blocks. When the triangle merge mode is used in a CU, the motion information (motion vectors and reference picture indices) for each of the two triangles are picked within the merge candidate list, the same one as derived for the normal merge mode (excluding any subblock motion vectors if present). As triangle shaped prediction units are only used in CUs coded with merge or skip modes, the motion information for the whole CU is completely determined by the combination of three pieces of information:

- The diagonal splitting direction, as illustrated in Fig. 9,
- The merge candidates list index for the motion information of the first triangle,
- The merge candidates list index for the motion information of the second triangle.

It was observed that some combinations are used more often than others. A look-up table was derived based on observed statistics of combinations, which is used to encode and decode a triangle index that signals the selected combination. Signaling the motion information for the whole CU as a combination may be used to control the encoder complexity, since only some of the most probable combinations from the look-up table need to be evaluated at the encoder. However, later work showed that signaling the three pieces of information mentioned above separately result in similar coding efficiency.

The two sets of motion information correspond to two different prediction signals that must be combined to obtain the triangle merge prediction for the CU. While in the technology described in [15], up to four reference blocks had to be accessed in case each of the two sets of motion information used bi-prediction, a subsequent version of the technology limited the number of blocks to be retrieved from memory to two, one for each of the triangles, therefore reducing the memory bandwidth for hardware and the complexity impact of the triangle merge mode. The two different predictions for the two triangles are combined by weighting the two prediction signals along the signaled diagonal splitting direction, as illustrated in Fig. 10 for a diagonal splitting direction from the top-left corner to the bottom-right corner. Two sets of weights can be applied based on the motion vectors and reference indices used by each triangle. A narrower filter is applied along the diagonal when the two triangles share similar motion information (as on the left side of Fig. 10), and a broader filter is applied along the diagonal otherwise (as on the right side of Fig. 10). However, later work showed that using only one set of weights (the set illustrated on the right side of Fig. 10) result in similar coding efficiency.

The triangle merge technology was reported to achieve luma BD-rate impacts from -0.3% for the RA configuration to -1% for the LB configuration.

D. Multi-Hypothesis Prediction

One CfP response employs multi-hypothesis inter-picture prediction by extending the bi-prediction with two predictors as used in HEVC and the JEM with one additional predictor. The uni-/bi-prediction signal is combined with the additional predictor using weights $(1-\alpha)$ and α where α is signaled and can either be 1/4 or $-1/8$. The proposed approach provides -1.1% luma BD-rate impact for the RA configuration as reported and further detailed in [23].

E. Bi-Prediction With CU-Level Weights

Bi-prediction with CU-level weights, also called generalized bi-prediction (GBI), is a variant of weighted prediction with the weights being signaled at the CU-level instead of assigned to a specific reference picture [41]. GBI can adapt to different luminance changes both across frames and across regions within one frame more flexibly than weighted prediction. It extended the concept of bi-directional prediction from using equal weights to unequal weights as follows:

$$P_{\mathrm{GBI}} = (w * P_{\mathrm{L0}} + (1-w) * P_{\mathrm{L1}} + offset) >> shift$$

where P_{GBI} denotes the prediction signal of the current-block; $P_{Li}, \forall i \in \{0, 1\}$, is a motion-compensated prediction signal of the current block from a reference picture in reference list *Li*; w and 1-w represent the weight values applied to P_{L0} and P_{L1}, respectively. *offset* and *shift* are used to normalize the final predictor in GBI. Five weight values are specified in the candidate set: **W**={-1/4, 3/8, 1/2, 5/8, 5/4}.

For merge mode, the weight selection is inherited from the selected merge candidate; otherwise (for motion vector prediction mode), the weight selection in GBI is explicitly signaled for a bi-predictive coded CU. The binarization of the weight index is as in TABLE I.

Post-CfP investigations of GBI report -0.9% BD-rate for the RA and -1.1% BD-rate for the LB configuration [41].

F. Merge With Motion Vector Difference

Merge with motion vector difference (MMVD) is a technology for motion vectors coding that was first proposed in [4] as ultimate motion vector expression (UMVE). It allows refinement of motion vectors in merge or skip mode, whereby only refinement by a set of pre-defined offsets is allowed.

For a CU coded with MMVD, the motion vector applied to the block is completely determined by a base candidate, chosen among the list of normal merge candidates, and refinement data applied to that base candidate. The refinement data itself consists of an offset and a direction. The offset is constrained to be equal to a power of two: only the values 1/4 sample, 1/2, 1, 2, 4, 8, 16 or 32 samples are allowed. The direction is only purely horizontal (left or right) or purely vertical (up or down). Therefore, contrary to normal motion vector differences, only a discrete set of refinement values are allowed, which can only be vertical or horizontal displacement of the base candidate. MMVD can be considered as a new form of "hybrid" coding mode between the normal merge mode and the normal motion vector difference signaling.

Symmetric coding of motion vectors can be used for bi-predicted blocks by deriving inverted motion vectors and applying scaling based on the Picture Order Count of the reference pictures. As a result, this technology tends to provide the most gain for coding configurations using bi-prediction from reference pictures preceding and following the current picture in output order.

According to subsequent test results, this method achieves about -0.9% BD-rate for the RA configuration.

G. Planar MV Field Prediction

Planar MV field prediction derives motion vectors of CU subblocks by interpolation of the motion vectors of all neighboring blocks in a manner similar to planar intra prediction. On top of the JEM with the JEM inter-picture prediction tools disabled, planar MV field prediction is reported to provide -0.4% BD-rate for the RA configuration [19].

H. Non-Adjacent MVP

Motion vector predictors can be taken from previous, non-adjacent spatial blocks. Similar methods were presented in [30], [16] and [20]. The motivation follows the concept of HEVC merge, which is to allow a block to reuse the motion parameters from previously decoded blocks. Compared to the HEVC merge where only adjacent neighboring blocks are considered for generating spatial merge candidates, the methods relax this constraint such that non-adjacent spatial blocks are also considered.

The method in [30] includes two additional non-adjacent spatial positions, one from the closest non-adjacent block HN in straight horizontal spatial distance and the other one from the closest non-adjacent block VN in straight vertical spatial distance, as illustrated in Fig. 11. The blocks are limited within a maximum distance of 1 CTU from the current block. It was reported in the subsequent core experiment [44] that the method gives -0.45% luma BD-rate impact for the RA configuration. Other CfP responses allow to use even more non-adjacent blocks for fetching motion information as in [16], [20].

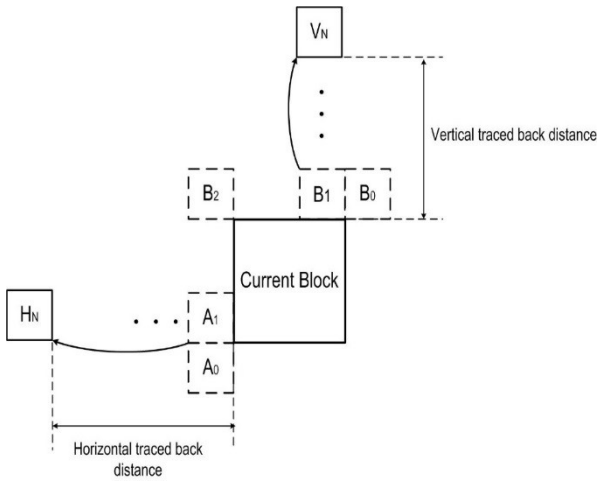


Fig. 11. Illustration of fetching non-adjacent spatial merge candidates in vertical (V_N) and horizontal (H_N) direction.

In general, more objective performance gain can be achieved when including more non-adjacent blocks for deriving the spatial merge candidates. However, the methods described here require accessing more motion information from the decoded area. Furthermore, the methods require additional buffer memory for loading the neighboring motion information for hardware decoding. Hence, a solution that allows non-adjacent motion information to be used for predicting the current block but comes with low memory requirements would be desirable.

V. Prediction Signal Filtering

In contrast to in-loop filtering, which is applied to the reconstructed signal, two filters have been proposed to improve the fidelity of the intra- and inter-picture prediction signals before the residual is added [23].

A. Diffusion Filter

In order to remove noise in the prediction signal while preserving edges, a signal-adaptive diffusion filter was proposed. The prediction samples are filtered using neighboring reconstructed samples in multiple iterations whereas linear and non-linear, position-dependent filter kernels can be selected. The reported gain of this signal-adaptive, iterative filter is -0.7% luma BD-rate for the RA configuration with a detailed description of the filter in [23].

B. DCT Thresholding

As another prediction signal refinement that was proposed, DCT thresholding generates prediction samples by applying a DCT to an extended block that includes reconstructed neighboring samples. All absolute coefficients that are below a fixed threshold are set to zero and an inverse DCT is performed with subsequent masking out of the prediction block from the extended block area. The DCT thresholding prediction filtering

was reported to provide −0.5% luma BD-rate for the RA configuration and is further detailed in [23].

VI. Transform, Scaling, and Quantization

The two main transform tools in the JEM [1], namely adaptive multiple transforms (AMT) or multiple transform selection (MTS) and a non-separable secondary transform (NSST), are part of most CfP responses. In this section modifications of both tools, targeting mainly a complexity reduction, are described followed by a short review of spatial varying transforms (SVT) for inter-picture prediction.

For quantization and scaling, a low-complexity variant of vector quantization named dependent quantization, adaptive scaling and an extended quantization step size have been proposed, all of which are shortly discussed in the following.

A. Restricted Multiple Transform Selection

In addition to type II discrete cosine transform (DCT)-II and the 4 × 4 type VII discrete sine transform (DST)-VII, which have been employed in HEVC, multiple transform selection (MTS) is used in the JEM for both intra- and inter-picture prediction residual blocks. MTS enables to utilize more transforms from the DCT/DST family, such as DST-VII, DCT-VIII, DST-I and DCT-V, and can be applied to residual blocks with both width and height smaller than or equal to 64 luma samples. Given the different residual signal statistics of different intra prediction modes, three transform subsets with 2 transform types each have been defined, and the transform subset is selected based on the intra prediction mode. The information of whether MTS is applied (flag) and if applied, which combination of transforms in the set is used (index), is signaled on CU level.

In [13], a restricted MTS is proposed by applying three modifications to reduce the complexity and one to increase the coding efficiency as follows:

- The number of additional transforms is reduced from four to two, i.e. DST-VII and DCT-VIII. This reduced set of additional transforms allows to remove the different intra mode-dependent transform subsets.
- The length of the additional transforms is restricted to 32 luma samples to reduce not only the memory usage to store the transform cores but also the computational complexity required for 64-point transforms.
- A discrete Fourier transform (DFT)-based DST-VII design enables 68% and 77% reduction of multiplication counts compared to direct multiply in 16-point and 32-point DST-VII, respectively.
- MTS index coding efficiency is improved by using truncated unary binarization and context modeling based on intra prediction mode.

In the JEM, large block-size transforms up to 128-point (DCT-II) are introduced to improve the coding efficiency by increasing the frequency domain resolution especially for high spatial resolution. For the 128-point DCT-II, the 64 high frequency transform coefficients are zeroed out. In [13], the 128-point DCT-II and, as mentioned above, the additional 64-point MTS transforms are removed. Therefore, the largest transform sizes are 64 and 32 for DCT-II and MTS, respectively.

It has been reported that the reduced complexity MTS based on the above simplifications achieves around −2% luma BD-rate for the RA configuration [45]. Compared with the original JEM MTS design, luma BD-rate is increased by 0.2% for the RA while applying above listed MTS modifications for complexity reduction.

B. Reduced Secondary Transform

In the JEM, a mode dependent non-separable secondary transform (MD-NSST) is applied at the encoder between forward core transform and quantization, and at the decoder between de-quantization and inverse core transform [1]. To keep low complexity, MD-NSST is only applied to the low frequency coefficients after the primary transform. There are 35×3 non-separable secondary transforms for both 4 × 4 and 8 × 8 NSST, where 35 is the number of transform sets specified by the intra prediction mode, and 3 is the number of NSST candidates for each intra prediction mode. The mapping from the intra prediction mode to the transform set is predefined. Instead of matrix multiplication, a Hypercube-Givens Transform (HyGT) with butterfly implementation is used to reduce memory space to store transform coefficients. The JEM NSST (i.e. HyGT) requires multiple iterations where transform output is fed back as input to the transform logic, and multiple iterations are required to produce final transform output. For example, 24 iterations (multiple passes) are required to perform an 8×8 HyGT. It eventually causes significant delay. Therefore, a reduced secondary transform (RST) is introduced in [13]. RST is based on direct multiplication approach so that it is implemented in a single pass without multiple iterations. Further, the NSST matrix dimension is reduced to minimize computational complexity and memory space to store the transform coefficients. The main idea of a Reduced Transform (RT) is to map an **N** (N is equal to 64 for 8×8 NSST) dimensional vector to an **R** dimensional vector in a different space, where **N/R** (**R** < **N**) is the reduction factor. Hence, instead of NxN matrix, RST matrix is an **R**×**N** matrix:

$$T_{R\times N} = \begin{bmatrix} t_{11} & t_{12} & t_{13} & \dots & t_{1N} \\ t_{21} & t_{22} & t_{23} & \dots & t_{2N} \\ \vdots & & & \ddots & \vdots \\ t_{R1} & t_{R2} & t_{R3} & \dots & t_{RN} \end{bmatrix}$$

where the **R** rows of the transform are **R** bases of the **N** dimensional space. The inverse transform matrix for RST is the transpose of its forward transform.

In [13], RST for 8 × 8 NSST (or RST 8 × 8) with a reduction factor of 4 is applied. Instead of 64 × 64, which is the conventional 8 × 8 non-separable transform matrix size, a 16×64 direct matrix is used. Hence, the 64×16 inverse RST matrix is used at the decoder side to generate core (primary) transform coefficients in 8 × 8 top-left regions. The forward RST 8 × 8 uses 16×64 matrices so that it produces non-zero coefficients only in the top-left 4 × 4 region within the given 8 × 8 region. In other words, if RST is applied then the 8 × 8 region except the top-left 4 × 4 region will have only

zero coefficients. As a result, the RST index is not coded when any non-zero element exists within an 8×8 block region other than its top-left 4×4 region because it implies that RST was not applied. In such a case, the RST index is inferred to be zero. RST is applied for intra CU in both intra and inter slices, and for both luma and chroma. The above described RST provides -2.8% luma BD-rate for the RA configurations as reported in [46].

C. Spatial Varying Transform (SVT)

Spatial varying transform (SVT) or subblock transform uses horizontal or vertical symmetric binary and ternary splits to partition a CU into two or three smaller transform blocks where it is additionally signaled which transform block contains no residual. Furthermore, a fixed mapping between the position of the zero residual block and a transform from the MTS set (DCT-VIII or DST-VII) is introduced. More details on SVT are provided in [4].

D. Dependent Quantization

For further improving coding efficiency, dependent quantization was proposed in [23] to replace the uniform reconstruction quantizer from previous video coding standards with a low-complexity variant of vector quantization, which is known as trellis-coded quantization (TCQ). At the decoder, it introduces a second scalar quantizer where a state-machine with 4 states determines the quantizer based on the parity of the previous coefficient level. TCQ achieves -2.8% luma BD-rate for the RA configuration with a more detailed description provided in [23].

E. Adaptive Scaling

Adaptive quantization step size scaling scales the quantization step size by a factor which is decided by the contrast and average values of the reconstructed luma sample around the CU. It is described in greater detail in [4].

F. Extended Step Size

In order to achieve lower bit rates, it was proposed in [14] to extend the range of the quantization parameter (QP), which controls the step size, by increasing the maximum allowed QP value from 51 to 57.

VII. Entropy Coding

A. Transform Coefficient Coding

In transform coefficient coding entropy coding, different coefficient scan patterns have been proposed such as block size dependent coefficient scanning in [48] and coefficient scan regions in [4]. In addition to that, dependent quantization as reviewed in Section VI-D requires different ordering of level syntax elements and introduce a new syntax element for the parity as well as new contexts as further detailed in [23]. For the sign of the quantized transform coefficient, various sign prediction techniques have been proposed [4], [16], [33].

B. CABAC Engine

The CABAC entropy coding engine from HEVC had already been extended by a multi-window/multi-hypothesis probability estimator in the JEM, which most proposals were based on. More details on the JEM multi-hypothesis estimator can be found in [1].

VIII. In-Loop Filters

Compared to HEVC, which includes a deblocking and a sample adaptive offset in-loop filter, the JEM introduced a third and a fourth in-loop filter, i.e. the adaptive loop filter (ALF) and the bilateral filter. Both are shortly discussed in the following.

In addition to that, CfP responses proposed a noise suppressor filter, which is further detailed in [4], non-local loop filters [14], [21], [49], and convolutional neural network (CNN) loop filters. The CNN filters are reviewed in the following.

Besides additional in-loop filters, various CfP responses proposed modifications to the well-known deblocking filter, namely stronger deblocking and luma-adaptive deblocking, both discussed at the end of this section.

A. Adaptive Loop Filter (ALF)

A Wiener-based adaptive loop filter (ALF), which performs classification of 4×4 blocks into different classes of filters, based on directionality and 2D-Laplacian activity, was already investigated during the standardization of HEVC [47]. In the JEM, ALF was further extended by geometry transform-based classifiers on 2×2 blocks, coefficient prediction from fixed filters and temporal prediction from collocated pictures [1]. On top of that, various CfP responses have proposed ALF modifications, e.g. by jointly controlling luma and chroma filtering [4], signaling filters on a CTU instead of picture level [14], replacing the Laplacian classifier with a rank-based approach [23], reducing the complexity in terms of filter size [32], and Sobel-based classification [33].

B. Bilateral Filter

In the JEM, a bilateral filter is acting on the reconstructed block in order to reduce ringing artifacts from quantization [1]. Hence, subsequent intra blocks predict from bilaterally-filtered blocks. This is reported to give -0.5% luma BD-rate impact for the RA configuration [50], and is included in several CfP responses based on the JEM such as [30]. A modified version based on non-local bilateral filtering is included in [16], where the difference to neighboring samples are averaged before being used by the bilateral filter.

C. Convolutional Neural Network Loop Filters

During the 9$^{\text{th}}$ JVET meeting (held before the CfP responses were received), a CNN-based in-loop filter was proposed to replace the bilateral filter, deblocking filter and sample adaptive offset filter in the JEM for intra frames [51]. This proposal reduced the number of in-loop filtering stages from four to two, and reported -3.57%, -6.17% and -7.06% BD-rate impacts for the luma and the two chroma components,

TABLE II
SMOOTHNESS CONDITIONS FOR LONG DEBLOCKING FILTER

S	Smoothness conditions for each of the 8 lines of a 16 sample segment (line 0, 3, 4, 7, 8, 11, 12, 15)
7	$\left(\lvert p_0 - 2*p_2 + p_4\rvert + \lvert q_0 - 2*q_2 + q_4\rvert < \frac{3*\beta}{32}\right)$ && $\left(\lvert p_0 - p_2 - p_3 + p_5\rvert + \lvert q_0 - q_2 - q_3 + q_5\rvert < \frac{3*\beta}{32}\right)$&& $\left(\lvert p_0 - 2*p_3 + p_6\rvert + \lvert q_0 - 2*q_3 + q_6\rvert < \frac{3*\beta}{32}\right)$ && $\left(\lvert p_0 - p_3 - p_4 + p_7\rvert + \lvert q_0 - q_3 - q_4 + q_7\rvert < \frac{3*\beta}{32}\right)$
5	$\left(\lvert p_0 - p_4\rvert + \lvert q_0 - q_4\rvert < \frac{\beta}{8}\right)$ && $\left(\lvert p_0 - p_5\rvert + \lvert q_0 - q_5\rvert < \frac{\beta}{8}\right)$
3	$\left(\lvert p_0 - p_1 - p_2 + p_3\rvert + \lvert q_0 - q_1 - q_2 + q_3\rvert < \frac{3*\beta}{32}\right)$

TABLE III
FILTER KERNELS FOR LONG DEBLOCKING FILTER

S	ref_P	ref_Q	ref_M	f_i, i=0..S−1
7	$(p_7 + p_6 + 1) \gg 1$	$(q_7 + q_6 + 1) \gg 1$	$\left(\sum_{k=1}^{6}(p_k + q_k) + 2 \cdot p_0 + 2 \cdot q_0 + 8\right) \gg 4$	{ 59,50, 41,32,23,14, 5 }
5	$(p_5 + p_4 + 1) \gg 1$	$(q_5 + q_4 + 1) \gg 1$	$\left(\sum_{k=0}^{4}(p_k + q_k) + \sum_{k=0}^{2}(p_k + q_k) + 8\right) \gg 4$	{ 58,45, 32,19, 6 }
3	$(p_3 + p_2 + 1) \gg 1$	$(q_3 + q_2 + 1) \gg 1$	$\left(\sum_{k=0}^{2}(3 \cdot (p_k + q_k)) - p_2 - q_2 + 8\right) \gg 4$	{ 53,32, 11 }

respectively for the AI configuration. Later on, the network structure of this filter was simplified, and the usage of this CNN-based filter was extended to inter-picture prediction as well. There are a few other CNN-based in-loop filters proposed either as independent coding tools or as parts of CfP responses such as [10] and [17]. A broader overview of CNN-based coding tools is provided in [10].

D. Stronger Deblocking Filter

Several responses to the CfP proposed using longer tap deblocking filters to introduce stronger deblocking [4], [14], [16], [30], [33], [48]. The motivation for introducing longer deblocking filters is suppression of blocking artifacts mainly originating from coding of smooth areas with large blocks at relatively coarse quantization. A common element of many responses is to modify more than three luma samples on each side of a luma boundary when the block size orthogonal to the boundary (width for a vertical boundary, height for a horizontal boundary) on both sides of the boundary is equal to or larger than 16 luma samples and the signal contains smooth samples. One proposal also enables longer deblocking filters for chroma, modifying three chroma samples on each side of a chroma boundary when the block size on both sides is equal to or larger than 8 chroma samples and the signal contains smooth samples [33].

The longer deblocking filters in [30] are applied for luma and were designed to not modify ramps (linearly increasing or decreasing sample values) as in HEVC. A long deblocking filter can be applied when the HEVC strong/weak filter decision indicates strong filter. The number of samples S to modify on each side of the boundary are based on the smoothness of the samples orthogonal to the boundary as well as on the block sizes of the neighboring block P and the current block Q. The smoothness conditions to be true for all allowed numbers of samples S are defined in TABLE II. They need to be true for every second line of a 16 lines boundary segment where β is a QP-dependent activity threshold and p_i is the sample value in block P at a distance of i samples from boundary and q_i is the sample value at distance i from the boundary in block Q. The number of samples S to be filtered S are selected based on the block sizes of the neighboring and the current block as follows:

- for block sizes $\geq$ 32, select the largest S (7, 5 or 3) for which the smoothness condition is true,
- otherwise, for block sizes $\geq$ 16, select the largest S (5, 3) for which the smoothness condition is true,
- otherwise, for block sizes $\geq$ 8 check the smoothness condition for S=3.

If none of the conditions is fulfilled, the HEVC luma deblocking filters are used.

The long deblocking filtering modifies S samples on each side of the boundary by interpolation from a virtual sample ref_M centered on the boundary towards a virtual sample ref_P centered in-between samples in block P and also towards a virtual sample ref_Q centered in-between samples in block Q:

$$p_i' = Clip3(p_i - t_C, p_i + tc, (f_i * ref_M + (64 - f_i) * ref_P + 32) >> 6)$$
$$q_i' = Clip3(q_i - t_C, q_i + tc, (f_i * ref_M + (64 - f_i) * ref_Q + 32) >> 6)$$

where i ranges from 0 to S−1, t_C is a QP dependent clipping value, Clip3(l, h, x) clips x to be equal to or larger than l and to be equal to or smaller than h, the virtual samples ref_M, ref_P, ref_Q and the coefficients for the interpolation filter f_i are listed in TABLE III.

E. Luma-Adaptive Deblocking Filter

High dynamic range transfer functions could increase the quantization error in the video signal domain so that blocking artifacts become more visible in bright areas. A luma-adaptive deblocking filter that controls the deblocking strength based on the average luma level of reconstructed samples was proposed in [19] to attenuate these artifacts with stronger deblocking for brighter areas. The average luma level LL across a 4-sample deblocking edge is calculated as follows where p and q denote the samples on each side of the edge:

$$LL = \frac{(p_{0,0} + p_{0,3} + q_{0,0} + q_{0,3}) \gg 2}{1 \ll BitDepth}$$

TABLE IV
CODING EFFICIENCY OF INDIVIDUAL TOOLS IN BD-RATE Y [%]

	RA	AI	Ref
Partitioning			
Multi-type tree	−3.3		[12]
Asymmetric binary trees	−3.2		[16]
Geometric partitioning	−0.8		[28]
Intra			
Wide-angle intra prediction	−0.2		[31]
Line-based intra prediction	−1.2		[23]
Intra block copy	−0.3		[20]
Region-based template matching	−1.1		[23]
Decoder-side intra mode derivation	−0.4		[12]
Neural network-based intra	−1.8		[23]
Inter			
Affine flexing	−0.5		[38]
Combined intra-inter	−0.7		[40]
Triangle merge mode	−0.3		[15]
Multi-hypothesis (up to 3)	−1.1		[23]
Bi-Prediction with CU weights	−0.9		[41]
Merge with MVD	−0.9		[4]
Planar MV field prediction	−0.4		[19]
Non-adjacent MVP	−0.5		[44]
Prediction filtering			
Diffusion filter	−0.7		[23]
DCT Thresholding	−0.5		[23]
Transforms and quantization			
Restricted MTS	−2.0		[45]
Reduced secondary transform	−2.8		[45]
Dependent quantization	−2.8		[23]
In-loop filters			
Bilateral Filter	−0.5	−0.4	[50]
CNN-based loop filter		−3.6	[51]

IX. CODING EFFICIENCY OF INDIVIDUAL TOOLS

The coding efficiency of tools presented in the previous sections is summarized in TABLE IV in terms of BD-rate for the random-access (RA) and all-intra (AI) configurations.

X. CONCLUSION

Compared to the HEVC video coding standard, the JEM software platform of JVET introduced advances in compression over the HEVC reference model (HM) software with up to 30% bit rate reduction in BD-rate terms [1], and the measured benefit in terms of subjective video quality generally exceeded what was measured by BD-rate [7]. On top of that, responses to the joint call for proposals for a successor to HEVC presented technology that can achieve even higher compression gain. This gain comes mainly from more flexible block structures, new technologies in intra- and inter-picture prediction, prediction signal enhancement filtering, advances in quantization, and improved loop filtering. Other techniques present in the JEM, such as multiple transform selection and secondary transforms, were simplified without sacrificing a significant part of their coding efficiency benefit.

REFERENCES

[1] J. Chen, M. Karczewicz, Y.-W. Huang, K. Choi, J.-R. Ohm, and G. J. Sullivan, "The joint exploration model (JEM) for video compression with capability beyond HEVC," *IEEE Trans. Circuits Syst. Video Technol.*, to be published.

[2] A. Segall, V. Baroncini, J. Boyce, J. Chen, and T. Suzuki, *Joint Call for Proposals on Video Compression With Capability Beyond HEVC*, document JVET-H1002, Macao, China, Oct. 2017.

[3] A. Wieckowski *et al.*, *NextSoftware: An Alternative Implementation of the Joint Exploration Model (JEM)*, document JVET-H0084, Macao, China, Oct. 2017.

[4] K. Choi *et al.*, "Video codec using flexible block partitioning and advanced prediction, transform and loop filtering technologies," *IEEE Trans. Circuits Syst. Video Technol.*, to be published.

[5] *Advanced Video Coding for Generic Audiovisual Services, Version 1*, document Rec. ITU-T H.264 ISO/IEC 14496-10, ITU-T and ISO/IEC, 2003.

[6] *High Efficiency Video Coding, Version 1*, document Rec. ITU-T H.265 ISO/IEC 23008-2, ITU-T and ISO/IEC, 2013.

[7] V. Baroncini, J.-R. Ohm, and G. J. Sullivan, *Report of Results From the Call for Proposals on Video Compression With Capability Beyond HEVC*, document JVET-J0029, San Diego, CA, USA, Apr. 2018.

[8] E. François, D. Rusanovskyy, A. Segall, A. Tourapis, and P. Yin, "High dynamic range video coding technology in responses to the joint call for proposals on video compression with capability beyond HEVC," *IEEE Trans. Circuits Syst. Video Technol.*, to be published.

[9] Y. Ye, J. Boyce, and P. Hanhart, "Omnidirectional 360° video coding technology in responses to the joint call for proposals on video compression with capability beyond HEVC," *IEEE Trans. Circuits Syst. Video Technol.*, to be published.

[10] D. Liu *et al.*, "Deep learning-based technology in responses to the joint call for proposals on video compression with capability beyond HEVC," *IEEE Trans. Circuits Syst. Video Technol.*, to be published.

[11] G. Bjøntegaard, *Improvement of BD-PSNR Model*, document VCEG-AI11, Berlin, Germany, Jul. 2008.

[12] X. Xiu *et al.*, "A unified video codec for SDR, HDR, and 360° video applications," *IEEE Trans. Circuits Syst. Video Technol.*, to be published.

[13] M. Koo *et al.*, *Description of SDR Video Coding Technology Proposal by LG Electronics*, document JVET-J0017, San Diego, CA, USA, Apr. 2018.

[14] Y.-W. Huang *et al.*, "A VVC proposal with quaternary tree plus binary-ternary tree coding block structure and advanced coding techniques," *IEEE Trans. Circuits Syst. Video Technol.*, to be published.

[15] T. Toma *et al.*, *Description of SDR Video Coding Technology Proposal by Panasonic*, document JVET-J0020, San Diego, CA, USA, Apr. 2018.

[16] W.-J. Chien *et al.*, "Hybrid video codec based on flexible block partitioning with extensions to the joint exploration model," *IEEE Trans. Circuits Syst. Video Technol.*, to be published.

[17] P. Bordes *et al.*, *Description of SDR, HDR and 360° Video Coding Technology Proposal by Qualcomm and Technicolor—Medium Complexity Version*, document JVET-J0022, San Diego, CA, USA, Apr. 2018.

[18] K. Misra, C. A. Segall, and F. Bossen, "Tools for video coding beyond HEVC: Flexible partitioning, motion vector coding, luma adaptive quantization and improved deblocking," *IEEE Trans. Circuits Syst. Video Technol.*, to be published.

[19] S. Iwamura *et al.*, *Description of SDR and HDR video coding technology proposal by NHK and Sharp*, document JVET-J0027, San Diego, CA, USA, Apr. 2018.

[20] X. Li, X. Xu, X. Zhao, J. Ye, L. Zhao, and S. Liu, *Description of SDR Video Coding Technology Proposal by Tencent*, document JVET-J0029, San Diego, CA, USA, Apr. 2018.

[21] F. Wu *et al.*, *Description of SDR Video Coding Technology Proposal by University of Science and Technology of China, Peking University, Harbin Institute of Technology, and Wuhan University (IEEE 1857.10 Study Group)*, document JVET-J0032, San Diego, CA, USA, 10th Meeting, Apr. 2018.

[22] J. An, H. Huang, K. Zhang, Y. W. Huang, and S. Lei, *Quadtree Plus Binary Tree Structure Integration With JEM Tools*, document JVET-B0023, San Diego, CA, USA, 2nd Meeting, Feb. 2016.

[23] J. Pfaff *et al.*, "Video compression using generalized binary partitioning and advanced techniques for prediction and transform coding," *IEEE Trans. Circuits Syst. Video Technol.*, to be published.

[24] M. Bläser, J. Sauer, and M. Wien, *Description of SDR and 360° Video Coding Technology Proposal by RWTH Aachen University*, document JVET-J0023, San Diego, CA, USA, 10th Meeting, Apr. 2018.

[25] O. D. Escoda, P. Yin, C. Dai, and X. Li, "Geometry-adaptive block partitioning for video coding," in *Proc. Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, Honolulu, HI, USA, 2007, pp. I-657–I-660.

[26] E. Francois, X. Zheng, and P. Chen, *CE2: Summary of Core Experiment 2 on Flexible Motion Partitioning*, document JCTVC D229, Daegu, South Korea, 4th Meeting, Jan. 2011.

[27] G. Tech, Y. Chen, K. Müller, J.-R. Ohm, A. Vetro, and Y.-K. Wang, "Overview of the multiview and 3D extensions of High Efficiency Video Coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 26, no. 1, pp. 35–49, Jan. 2016.
[28] M. Bläser, J. Schneider, J. Sauer, and M. Wien, "Geometry-based partitioning for predictive video coding with transform adaptation," in *Proc. Picture Coding Symp. (PCS)*, San Francisco, CA, USA, 2018, pp. 134–138.
[29] A. Wieckowski, J. Ma, H. Schwarz, D. Marpe, and T. Wiegand, "Fast partitioning decision strategies for the upcoming versatile video coding (VVC) standard," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Taipei, Taiwan, Sep. 2019, pp. 4130–4134.
[30] R. Sjöberg *et al.*, *Description of SDR and HDR Video Coding Technology Proposal by Ericsson and Nokia*, document JVET-J0012, San Diego, CA, USA, 10th Meeting, Jul. 2018.
[31] J. Lainema, *CE3: Wide-Angle Intra Prediction (Test 1.3.1)*, document JVET-K0046, Ljubljana, Slovenia, Apr. 2018.
[32] J. W. Kang *et al.*, *Description of SDR Video Coding Technology Proposal by ETRI and Sejong University*, document JVET-J0013, San Diego, CA, USA, Apr. 2018.
[33] T. Suzuki, M. Ikeda, and K. Sharman, *Description of SDR and HDR Video Coding Technology Proposal by Sony*, document JVET-J0028, San Diego, CA, USA, Apr. 2018.
[34] S.-L. Yu and C. Chrysafis, *New Intra Prediction Using Intra-Macroblock Motion Compensation*, document JVT-C151, Fairfax, CA, USA, May 2002.
[35] M. Budagavi and D.-K. Kwon, *AHG8: Video Coding Using Intra Motion Compensation*, document JCTVC-M0350, Apr. 2013.
[36] X. Xu *et al.*, "Intra block copy in HEVC screen content coding extensions," *IEEE J. Emerg. Sel. Topic Circuits Syst.*, vol. 6, no. 4, pp. 409–419, Dec. 2016.
[37] X. Xu, X. Li, G. Li, and S. Liu, *Intra Block Copy Improvement on Top of Tencent's CFP Response*, document JVET-J0050, San Diego, CA, USA, Apr. 2018.
[38] J. Lainema, *CE4: Affine Flexing (Test 1.2)*, document JVET-K0047, Ljubljana, Slovenia, Jul. 2018.
[39] K. Andersson, *Combined Intra Inter Prediction Coding Mode*, document ITU-T SG16/Q6 (VCEG), VCEG-AD11, Hangzhou, China, Oct. 2006.
[40] M.-S. Chiang, C.-W. Hsu, Y.-W. Huang, and S.-M. Lei, *CE10.1: Combined and Multi-Hypothesis Prediction*, document JVET-K0257, Ljubljana, Slovenia, Jul. 2018.
[41] Y.-C. Su, T.-D. Chuang, C.-Y. Chen, Y.-W. Huang, and S.-M. Lei, *CE4.4.1: Generalized Bi-Prediction for Inter Coding* document JVET-K0248, Ljubljana, Slovenia, Jul. 2018.
[42] Y. Ahn, H. Ryu, and D. Sim, *Diagonal Motion Partitions on top of QTBT Block Structure*, document JVET-H0087, 8th Meeting, Oct. 2017.
[43] Y. Chen *et al.*, "An overview of core coding tools in the AV1 video codec," in *Proc. Picture Coding Symp. (PCS)*, San Francisco, CA, USA, 2018, pp. 41–45.
[44] R. Yu, P. Wennersten, and R. Sjöberg, *CE4-2.1: Adding Non-Adjacent Spatial Merge Candidates*, document JVET-K0228, Ljubljana, Slovenia, Jul. 2018.
[45] M. Koo, M. Salehifar, J. Lim, and S. H. Kim, *CE 6-1.11: AMT Replacement and Restriction*, document JVET-K0096, Ljubljana, Slovenia, Jul. 2018.
[46] M. Salehifar, M. Koo, J. Lim, and S. Kim, *CE 6.2.6: Reduced Secondary Transform (RST)*, document JVET-K0099, Ljubljana, Slovenia, Jul. 2018.
[47] C.-Y. Tsai *et al.*, "Adaptive loop filtering for video coding," *IEEE J. Sel. Topics Signal Process.*, vol. 7, no. 6, pp. 934–945, Dec. 2013.
[48] K. Kawamura, Y. Kidani, and S. Naito, *Description of SDR Video Coding Technology Proposal by KDDI*, document JVET-J0016, San Diego, CA, USA, Apr. 2018.
[49] Z. Wang *et al.*, *Description of SDR Video Coding Technology Proposal by DJI and Peking University*, document JVET-J0011, San Diego, CA, USA, Apr. 2018.
[50] P. Wennersten, J. Ström, Y. Wang, K. Andersson, R. Sjöberg, and J. Enhorn, "Bilateral filtering for video coding," in *Proc. IEEE Int. Conf. Vis. Commun. Image Process. (VCIP)*, St. Petersburg, FL, USA, Dec. 2017, pp. 1–4.
[51] L. Zhou *et al.*, "*Convolutional Neural Network Filter (CNNF) for Intra Frame*, document JVET-I0022, Gwangju, South Korea, Jan. 2018.

Benjamin Bross (S'11–M'17) received the Dipl.-Ing. degree in electrical engineering from RWTH Aachen University, Aachen, Germany, in 2008.

In 2009, he joined the Fraunhofer Institute for Telecommunications, Heinrich Hertz Institute, Berlin, Germany, where he is currently a Project Manager with the Video Coding and Analytics Department and a part-time Lecturer with the HTW University of Applied Sciences, Berlin. Since 2010, he has been very actively involved in the ITU-T VCEG | ISO/IEC MPEG video coding standardization processes as a Technical Contributor, a Coordinator of core experiments, and the Chief Editor of the High Efficiency Video Coding (HEVC) standard [ITU-T H.265 | ISO/IEC 23008-2] and the emerging Versatile Video Coding (VVC) standard.

Besides giving talks about recent video coding technologies, he is an author or a coauthor of several fundamental HEVC-related publications, and an author of two book chapters on HEVC and *Inter-Picture Prediction Techniques in HEVC*. He received the IEEE Best Paper Award at the 2013 IEEE International Conference on Consumer Electronics, Berlin, in 2013, the SMPTE Journal Certificate of Merit in 2014 and the Emmy Award at the 69th Engineering Emmy Awards in 2017 as part of the Joint Collaborative Team on Video Coding for its development of HEVC.

Kenneth Andersson received the M.Sc. degree in computer science and engineering from Luleå University in 1995 and the Ph.D. degree from Linköping University in 2003. In 1994, he started to work with Ericsson Research in speech coding. Since 2005, he has been active in video coding standardization in ITU-T and ISO/IEC for development of High Efficiency Video Coding (HEVC) and Versatile Video Coding (VVC). His main research interests include image and video signal processing, and video coding.

Max Bläser received the Dipl.-Ing. degree in electrical engineering from RWTH Aachen University, Aachen, Germany, in 2013, where he is currently pursuing the Dr.-Ing. degree in video coding with the Institute for Communication Engineering (IENT). In 2014, he joined the IENT, RWTH Aachen University, where he is currently a Researcher. His current research interest includes non-rectangular block partitioning for prediction and transform coding.

Virginie Drugeon received the engineering degree from the École Nationale Supérieure des Télécommunications (ENST) in Paris, France, in 2006. She has been an Engineer with Panasonic Business Support Europe, Germany, for 12 years, in the area of video coding and transmission. She has participated to Joint Collaborative Team on Video Coding (JCT-VC) and Joint Video Experts Team (JVET) for the standardization of the High Efficiency Video Coding (HEVC) standard [ITU-T H.265 | ISO/IEC 23008-2] and the emerging Versatile Video Coding (VVC) standard, and to Digital Video Broadcasting (DVB) for the standardization of 3D Television (3DTV) and Ultra-High-Definition Television (UHDTV) broadcast, including high dynamic range and high frame rate. She is currently the Chair of the DVB TM-AVC sub-group (Technical Module sub-group on Audio/Video Coding).

Seung-Hwan Kim received the Ph.D. degree from the Gwangju Institute of Science and Technology (GIST) in 2008. In 2009, he was with the University of Southern California (USC) as a Post-Doctoral Research Fellow. In 2011, he joined Sharp Laboratories of America, where he has been active in video coding standardization in ITU-T and ISO/IEC for development of High Efficiency Video Coding (HEVC) and its extension projects, including Joint Video Experts Team (JVET). Since 2016, he has been with LG Electronics, where he is currently an Assistant Vice President and has lead Versatile Video Coding (VVC) activity.

Jani Lainema received the M.Sc. degree in computer science from the Tampere University of Technology, Finland, in 1996. He joined the Visual Communications Laboratory, Nokia Research Center in 1996. Since 1996, he has been contributing to the designs of ITU-T's and MPEG's video coding standards and to the evolution of different multimedia service standards in 3GPP, DVB, and DLNA. He is currently a Bell Labs Distinguished Member of Technical Staff and also as a Distinguished Scientist with Visual Media, Nokia Technologies, Tampere, Finland. His research interests include video, image, and graphics coding and communications.

Jingya Li received the B.S. degree in engineering and the M.S. degree in science from Nanyang Technological University, Singapore, in 2012 and 2016, respectively.

Since 2012, she has been with Panasonic Research and Development Center Singapore involved in the area of video compression SoC development and Versatile Video Coding (VVC) standard as a participant in Joint Video Experts Team (JVET).

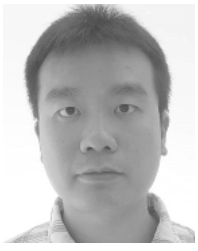
Shan Liu received the B.Eng. degree in electronics engineering from Tsinghua University, Beijing, China, and the M.S. and Ph.D. degrees in electrical engineering from the University of Southern California, Los Angeles, CA, USA. She was the Chief Scientist and the Head of America Media Laboratory, Futurewei Technologies. She was the Director of the Multimedia Technology Division, MediaTek USA. She was also with MERL, Sony, and IBM. She is currently a Distinguished Scientist and a General Manager with Tencent, where she heads the Media Laboratory. She has been actively contributing to international standards, such as SVC, H.265/HEVC, DASH, OMAF, and VVC. She also involved in products such as PlayStation3, and is leading the laboratory to build technology solutions, platforms, services, and products for Tencent internal and external customers. She served on the Industrial Relationship Committee of the IEEE Signal Processing Society from 2014 to 2015 and was the Vice President of industrial relations and development of the Asia–Pacific Signal and Information Processing Association (APSIPA) from 2016 to 2017. She was named as an APSIPA Industrial Distinguished Leader in 2018.

Jens-Rainer Ohm (M'92) has been with the Moving Picture Experts Group (MPEG) since 1998. He has been the Chair of the Institute of Communication Engineering, RWTH Aachen University, Germany, since 2000. He also serves as the Dean of the Faculty of Electrical Engineering and Information Technology, RWTH Aachen University.

He has been chairing/co-chairing various standardization activities in video coding, namely, the MPEG Video Subgroup 2002–2018, the Joint Video Team (JVT) of MPEG and ITU-T SG 16 VCEG 2005–2009, and the Joint Collaborative Team on Video Coding (JCT-VC) since 2010, and the Joint Video Experts Team (JVET) since 2015. His research and teaching activities cover the areas of multimedia signal processing, analysis, compression, transmission, and content description, including 3D and VR video applications, bio signal processing and communication, application of deep learning approaches in the given fields, and fundamental topics of signal processing and digital communication systems. He has authored textbooks on *Multimedia Signal Processing, Analysis and Coding* and *Communication Engineering and Signal Transmission*, and numerous articles from the fields mentioned above.

Prof. Ohm has served on the editorial boards of Journals and TPCs of various conferences.

Gary J. Sullivan (S'83–M'91–SM'01–F'06) is currently a Video and Image Technology Architect with the AI and Research Group, Microsoft Corporation. He has been a longstanding Chairman or Co-Chairman of various video and image coding standardization activities in ITU-T VCEG, ISO/IEC MPEG, ISO/IEC JPEG, and in their joint collaborative teams, since 1996. He is best known for leading the development of the Advanced Video Coding (AVC) standard [ITU-T H.264 | ISO/IEC 14496-10], the High Efficiency Video Coding (HEVC) standard [ITU-T H.265 | ISO/IEC 23008-2], and the various extensions of those standards. Most recently, he has been the Co-Chair of the Joint Video Experts Team (JVET), since 2015, for developing the upcoming Versatile Video Coding (VVC) standard. At Microsoft, he has been the Originator and the Lead Designer of the DirectX Video Acceleration (DXVA) video decoding feature of the Microsoft Windows operating system. He was the recipient of the IEEE Masaru Ibuka Consumer Electronics Award, the IEEE Consumer Electronics Engineering Excellence Award, two IEEE TRANSACTION CSVT Best Paper awards, the INCITS Technical Excellence Award, the IMTC Leadership Award, and the University of Louisville J. B. Speed Professional Award in engineering. The team efforts that he has led have been recognized by three Emmy Awards. He is a fellow of the SPIE.



Ruoyang Yu received the B.S. degree from Zhejiang University, Hangzhou, China, in 2010 and the M.S. degree from Lund University, Lund, Sweden, in 2012. Since 2012, he has been with Ericsson Research, Stockholm, Sweden, where he is currently a Senior Researcher involving with video compression technology.