THE VIDEO CODEC LANDSCAPE IN 2020

Michel Kerdranvat, Ya Chen, Rémi Jullian, Franck Galpin, Edouard François
InterDigital R&D, Rennes, France

Abstract *– Video compression is a key technology for new immersive media experiences, as the percentage of video data in global Internet traffic (80% in 2019 according to the 2018 Cisco Visual Networking Index report) is steadily increasing. The requirement for higher video compression efficiency is crucial in this context. For several years intense activity has been observed in standards organizations such as ITU-T VCEG and ISO/IEC MPEG developing Versatile Video Coding (VVC) and Essential Video Coding (EVC), but also in the ICT industry with AV1. This paper provides an analysis of the coding tools of VVC and EVC, stable since January 2020, and of AV1 stable since 2018. The quality and benefits of each solution are discussed from an analysis of their respective coding tools, measured compression efficiency, complexity, and market deployment perspectives. This analysis places VVC ahead of its competitors. As a matter of fact, VVC has been designed by the largest community of video compression experts, that is JVET (Joint Video Experts Team between ITU-T and ISO/IEC). It has been built on the basis of High Efficiency Video Coding (H.265/HEVC) and Advanced Video Coding (H.264/AVC) also developed by joint teams, respectively JCT-VC and JVT, and issued in 2013 and 2003 respectively.*

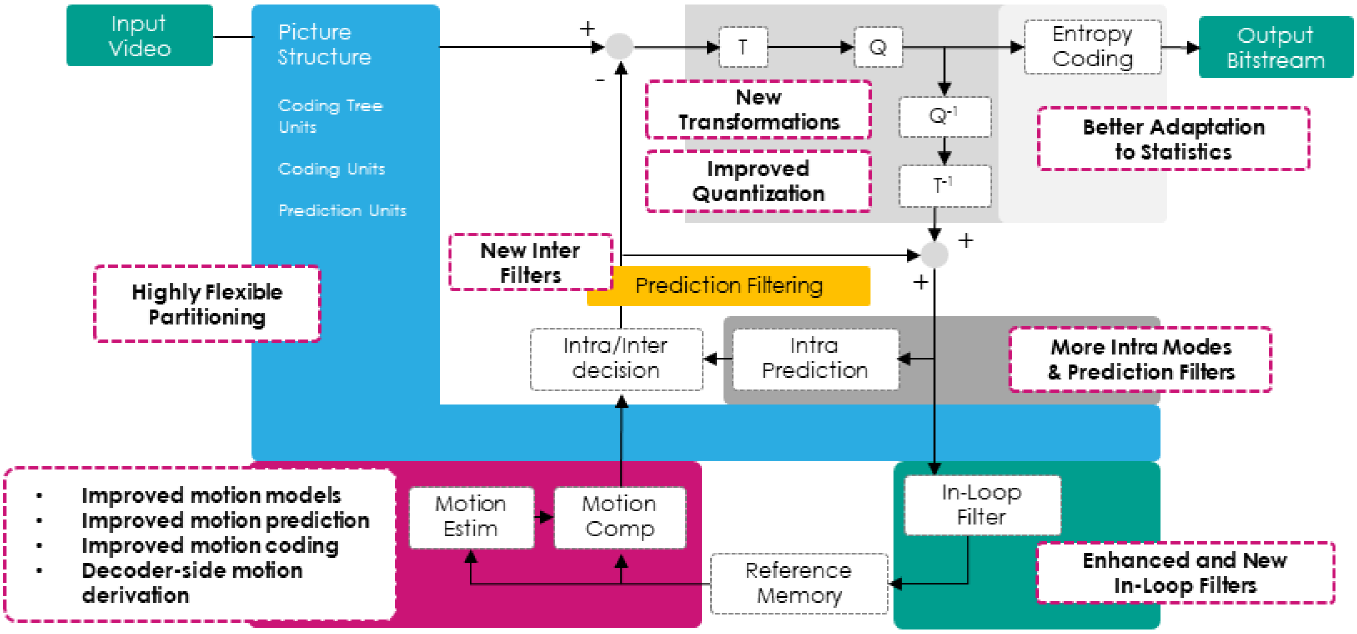
Keywords – AV1, EVC, HEVC, Video Coding, VVC.

1. INTRODUCTION

The landscape of video coding is evolving rapidly. New solutions are developed in the standards organizations ISO/IEC MPEG (Motion Picture Experts Group) and ITU-T VCEG (Video Coding Experts Group), but also in the industry consortium Alliance for Open Media (AOM) founded in 2015. Even though Advanced Video Coding (AVC) [1], which was completed in 2003, is the dominant standard nowadays for video distribution, new services (such as UltraHD 4K format) are deploying High Efficiency Video Coding (HEVC) [2] designed jointly by MPEG and VCEG, and issued in 2013. This move to HEVC is justified by a compression factor of around 2 with regards to AVC leading to half the transmission or storage bit rate for the same subjective quality [3]. To face the emergence of increasing picture resolution (e.g., UltraHD 8K format) and pixel definition (bit depth, color gamut, dynamic range), higher frame rates, and new uses (Video on mobile, Virtual Reality with 360° video...) impacting production, network distribution, and receivers (TVs, mobiles…), MPEG and VCEG have again joined their resources in 2015 to start an exploratory phase, resulting in 2018 in the launching of a joint team, Joint Video Experts Team (JVET), tasked with designing a new video coding standard, Versatile Video Coding (VVC), targeting a 50% compression gain over HEVC. The specification of VVC [4] has been stable since January 2020 and completion is foreseen for July 2020. As with previous ITU-T and ISO/IEC standards, VVC will be royalty-bearing, but unlike HEVC, some industry players are anticipating the publication of suitable licensing terms for a successful deployment. Besides this effort, at the initiative of a few companies, a parallel project for another video coding standard has been worked out in MPEG with the same schedule. The name of this new coding standard is Essential Video Coding (EVC) [5], with the objective of being royalty-free for the baseline profile and royalty-bearing for the main profile with expected timely publication of the licensing terms. The landscape is completed by AOM which released the initial version of AV1 [6] in 2018. AV1 was developed with the goal of being royalty-free, and hence its development process took into account not only technical factors, but also patent rights factors. However, licensing terms of video coding standards have always been defined by patents holders once their technical specifications have been released.

This paper presents results of a comparison between the three solutions VVC, EVC and AV1 in terms of compression efficiency, complexity and features relative to HEVC. Careful attention was paid to an as-fair-as-possible comparison between the encoding softwares, considering that it is not possible to get perfectly aligned comparisons, as the encoder algorithms may significantly differ. The first section presents an overview of the underlying main coding tools from respective standards. The following sections describe the test conditions and associated configuration parameters used in the comparisons.

© International Telecommunication Union, 2020
Some rights reserved. This work is available under the CC BY-NC-ND 3.0 IGO license: https://creativecommons.org/licenses/by-nc-nd/3.0/igo/.
More information regarding the license and suggested citation, additional permissions and disclaimers is available at:
https://www.itu.int/en/journal/2020/001/Pages/default.aspx



T: Transform; Q: Quantization; T^{-1} : Inverse Transform ; Q^{-1} Inverse Quantization

Fig. 1 – Video coding improvements on HEVC encoding scheme

Then, the performances of the solutions relative to four objective metrics are presented, but also the estimation of their complexity . Some observations are also given on the subjective quality. Finally some concluding remarks are provided on measured results and market perspectives.

2. OVERVIEW OF THE VIDEO CODING DESIGNS

HEVC, VVC, EVC and AV1 are all based on the well-known hybrid block-based coding architecture combining inter-frame and intra-frame predictions, transformation of the prediction residual, quantization, and entropy coding. The major tools of each new video coding with regards to HEVC are described from a high-level perspective in the following paragraphs. Fig. 1 depicts a block diagram of an HEVC encoder as reference design, highlighting where the improvements occur (dashed purple boxes). In the following, more details are provided on these improvements. The analysis is not intended to be exhaustive but to underline some important design differences between these standards.

2.1 Partitioning

A significant improvement brought by recent video coding standards as compared to HEVC lies in the picture partitioning, as illustrated in Fig. 2. First, the size of the base processing structure, known as a Coding Tree Unit (CTU) in HEVC and VVC, and partition tree in AV1, is increased from a maximum block size of 64×64 luma samples in HEVC to 128×128 samples. Large uniform areas, as well as large pictures, can now be more efficiently handled. A CTU can be further split into Coding Units (CUs). The CTUs and CUs can also be split into PUs that share common parameters such as the coding mode.

In HEVC the CU partitioning (intra/inter partition) uses quadtree only. However, a CU can be further split into PUs, in intra or interprediction, into six other possible sub-partitions (Fig. 2).

VVC and EVC support six common partitioning types for CU and PU: no partition, quaternary partition, two binary partitions, and two ternary partitions (1/4, 2/4, 1/4 horizontal or vertical partitioning of the CU). Morever, 64 geometric PUs are introduced in VVC to allow for a non-horizontal or non-vertical split in two parts of a rectangular or square CU. Each of the 64 geometic partitions is signalled by an index value pointing to its parameters (angle, distance). This mode cannot be applied to CUs with width or height greater than 64, or with width or height less than 8. VVC also includes a specific partitioning mode called Intra Sub-Partitioning (ISP).

AV1 also supports a larger choice of CU and PU partitioning than HEVC with ten partitioning shapes, including no-split, binary and quarterly split as EVC or VVC. Four ternary split shapes (see Fig. 2) and two split shapes into four equally sized partitions. However, all these partitions are a subset of what can be done in VVC and EVC by cascading their splits, leading to a higher partitioning flexibility. In addition, 16 wedge prediction types, similar to geometric PUs of VVC, are available.

One specific feature of VVC is the support in intra slices of separate luma and chroma partitioning, which allows co-located luma and chroma Coding Tree Blocks to be independtly partitioned.

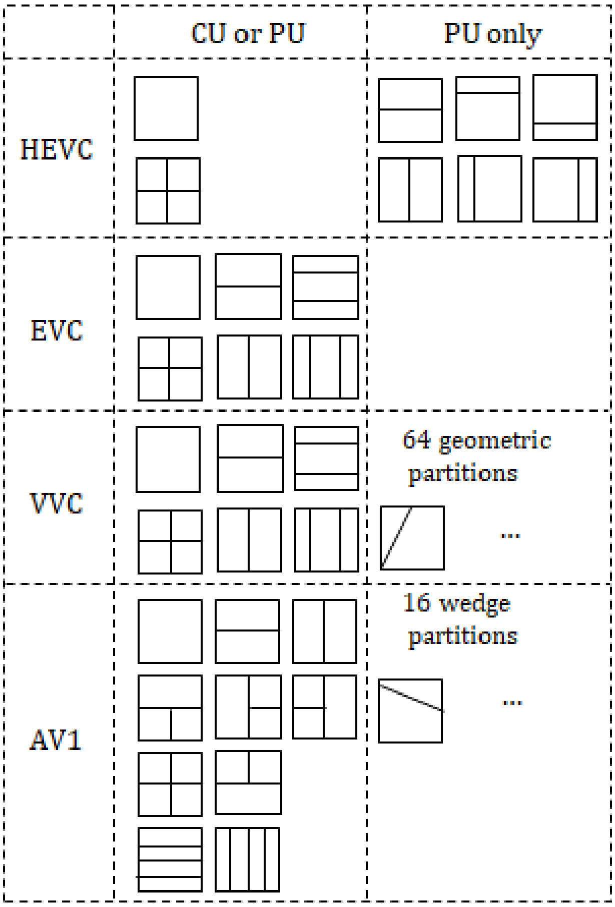


Fig. 2 – Partitions

2.2 Intra coding

Intra prediction in HEVC is based on 33 angular predictors plus planar and DC modes, all predicted from reference samples in the causal spatial neighbourhood of the coded block (top line, left column). In VVC, 93 angular predictors, plus planar and DC modes are specified. Furthermore, the predictors can be computed from an extended neighbourhood. New matrix-based prediction modes are also inserted for luma, leading to 30 additional modes. EVC is close to HEVC with 30 angular predictors, plus planar and DC modes. AV1 specifies 56 angular predictors plus DC and 4 additional modes.

For chroma samples, in addition to the conventional directional, planar and DC prediction modes commonly supported by the four standards, VVC and AV1 support a cross-component prediction mode. In this mode, the chroma samples are predicted from co-located reconstructed luma samples.

The reference and prediction filtering processes used in HEVC are also more elaborated in VVC and AV1. In particular VVC introduces a position-dependent prediction combination, while AV1 defines a recursive filtering-based intra predictor.

2.3 Inter coding

Inter prediction has been significantly enhanced in the recent video coding solutions by introducing the affine motion model on top of the regular translational model, along with more efficient coding of motion vectors and higher motion accuracy.

HEVC specifies three inter prediction types which can be unidirectional (one motion vector) or bidirectional (two motion vectors) using the pictures available in the Decoded Pictures Buffer (DPB): Advanced Motion Vector Prediction (AMVP), Merge, and Skip. In AMVP, both motion information (motion vectors, reference pictures) and the prediction residual are signalled. In merge mode, only the residual is signalled and motion information is derived from a list of most probable candidates. The Skip mode is similar to the Merge mode for the motion, but no residual is transmitted. Fig. 3 and Fig. 4 illustrate the increased number of coding modes supported by VVC compared to HEVC, as well as the impact of the new partitioning. The same color code is used on both figures with different color tints for VVC new inter-coding modes: orange for AMVP, green for Merge and blue for Skip.

VVC adds affine motion prediction to AMVP, Merge, and Skip modes. The regular Merge and Skip modes are enhanced with MMVD (Merge with Motion Vector Difference), ATMVP (Advanced Temporal Motion Vector Prediction) and GPM (Geometric Partition Merge). CIIP (Combined Inter Intra Prediction) is added to Merge mode. Moreover the motion information can be refined at the decoder to enhance the prediction per pixel at constant bit rate of the motion information. It can be noticed on the figures that the amount of Intra modes (red color) for VVC has decreased compared to the HEVC case as a result of improved Inter coding (Intra has higher bit rate).

VVC and EVC inter-coding tools are very close, one important difference being the VVC support of the geometric partitioning mode.

Like HEVC, AV1 has three inter prediction types AMVP, Merge and Skip. The motion model can be translational, affine, or global affine, with new predictions such as Wedge predictions, similar to Geometric Partition Merge (GPM), and compound inter-intra prediction similar to CIIP. Overlapped Block Motion Compensation (OBMC) is used which is not the case for the other solutions.

VVC and EVC include a motion refinement tool named Decoder-side MV Refinement (DMVR). In addition, VVC defines two new prediction refinement modes based on the optical flow, named Bidirectional Optical Flow (BDOF) and Prediction Refinement with Optical Flow (PROF).

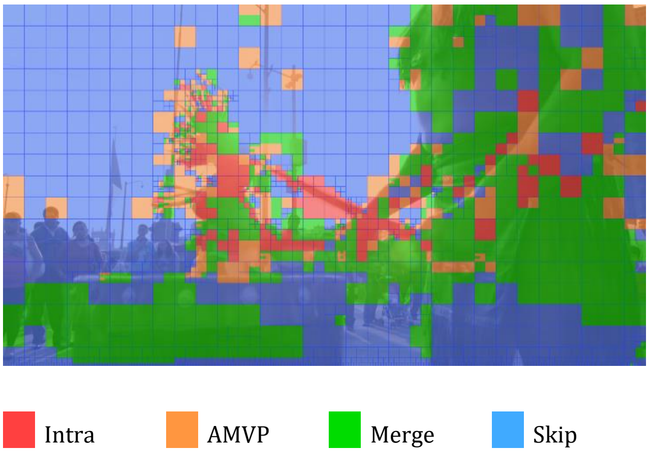


Fig. 3 – HEVC coding modes per coding unit

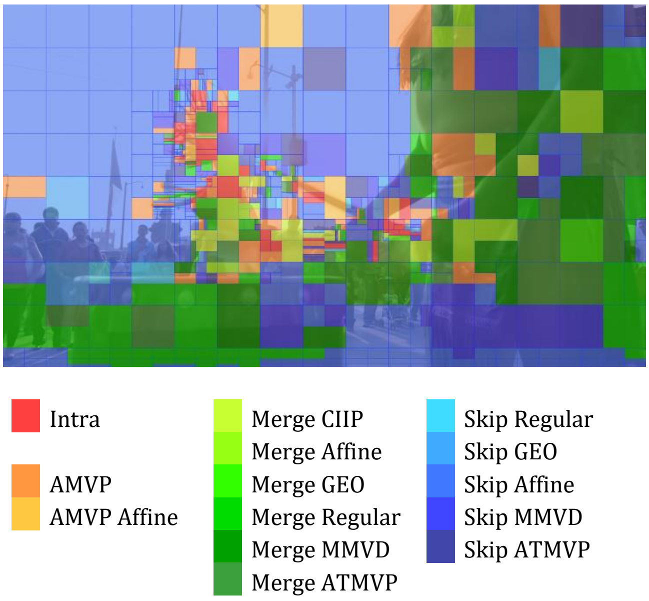


Fig. 4 – VVC coding modes per coding unit

2.4 Transforms

HEVC transforms are square separable NxN DCT-2 (Discrete Cosine Transform) for 4x4 to 32x32 block sizes, plus DST-7 (Discrete Sine Transform) for the intra 4x4 block size. The recent coding schemes introduce more variety with the support of multiple separable transform types for square and rectangular blocks, and for larger sizes up to 64x64.

In VVC and EVC, the concept of Multiple Transform Selection (MTS) is specified for residual coding of both inter and intra-coded blocks, using DCT-2, DCT-8 or DST-7 for square or rectangular blocks. In addition, VVC inserts a set of Low Frequency Non-Separable Transforms (LFNST) implemented at the encoder between the primary separable transforms and the quantization, while at the decoder between the inverse quantization and the inverse primary transform.

AV1 has also a richer set of square or rectangular transforms: DCT-2, ADST (Asymmetric Discrete Sine Transform), flipped ADST (applying ADST in reverse order) and identity transform which is equivalent to transform skip of HEVC or VVC.

The concept of RQT (Residual QuadTree) supported in HEVC is not specified anymore in the three more recent standards. Nevertheless, in VVC, a CU can be split in smaller TUs using the Sub-Block Transform (SBT).

2.5 In-loop filters

New in-loop filters have improved the objective and subjective performance of the new video coding specifications. In VVC and EVC a new in-loop filter called ALF (Adaptive Loop Filter) is inserted on top of the Deblocking Filter (DF) and of the Sample Adaptive Offset (SAO) filter used in HEVC. ALF is a block-based adaption filter. For the luma component, one filter among 25 is selected for each 4×4 block, based on the direction and activity of local gradients. For chroma the choice is among 8 filters. Furthermore, a Cross-Component ALF is introduced to refine chroma details lost in the coding loop, by using co-located luma samples. “Adaptive” means the filters can vary in a video stream according to statistics of the content, and also according to the blocks gradients-based classification.

In VVC, a specific coding tool called luma mapping with chroma scaling (LMCS) is added as a new processing block prior to the loop filters. The luma mapping is based on a piecewise linear model which adjusts the dynamic range of the input signal by redistributing the codewords across the signal range to improve compression efficiency. For the chroma the scaling applies to the prediction residual and depends on the average value of top and/or left reconstructed neighbouring luma samples. As a consequence, with LMCS all the reconstruction processing in luma (inverse quantization, inverse transformation and prediction for inter and intra modes) is made in the mapped domain. Like ALF, the mapping function can vary in a video stream according to content statistics.

AV1 also combines different in-loop filters in addition to a regular deblocking filter. A Constrained Directional Enhancement Filter (CDEF), which de-rings contours and preserves the details to be applied after deblocking, works by estimating edge directions. A loop restoration filter can be applied selectively according to noise level.

2.6 Entropy coding

Arithmetic coding is used in all the video coding solutions addressed in this paper. HEVC entropy coding was based on the so-called CABAC (Context-based Adaptive Binary Arithmetic Coding) initially introduced in AVC [1]. CABAC is composed of four main steps: the binarization of each syntax element in a binary string, followed by the choice of a probability model for each bit (or “bin”) to be “0” or “1” based on the context, then the binary arithmetic coder encodes each bin with less than a bit, and finally the probability of the selected model is updated. VVC has continued to improve CABAC with a higher accuracy on the probabilities derivation, by increasing the number of probability models, and by updating the probabilities through an adaptive double-window. EVC entropy coding is also a CABAC engine, but does not benefit from all the enhancements brought in VVC. AV1 entropy coding is different as it uses a multi-symbol entropy coding per syntax element without binarization step.

2.7 Screen Content Coding

In its Screen Content Coding (SCC) extensions issued in 2016 [7], HEVC specifies four new coding tools adapted to “Screen Content” which are pictures partially or totally composed of computer graphic objects. These tools are Intra Block Copy (IBC), Palette, Block-wise Differential Pulse Code Modulation (BPDCM) when Transform Skip is allowed, and Adaptive Color Transform (ACT). IBC is equivalent to a motion compensation but within the same picture. In VVC the motion search area is limited to the current CTU and part of the left CTU to control the complexity. Palette reduces the number of codewords to encode to a limited number of triplets (one luma and two chroma values). AV1 has equivalent modes to IBC and Palette, but EVC has only IBC.

BDPCM and ACT have been reintroduced in VVC with some adaptations. BDPCM allows us to encode prediction residuals without transform (Transform Skip) by predicting them from the previously coded residuals. BDPCM is relevant for Screen Content but also for lossless compression. ACT reduces the redundancy between color components in typically RGB coding through a color space conversion.

2.8 Other tools

A new feature called Reference Picture Resampling (RPR), which allows the picture size to vary from picture to picture, is specified in VVC. This feature is also available in AV1 but in the horizontal direction only. RPR offers an additional level of flexibility for bit-rate control to adapt to network bandwidth variation.

In HEVC, annexes were produced after the publication of the initial version of the standard to include spatial and temporal scalability, as well as the coding of multiview (3DTV) content. These extensions were based on the layer concept, which is available in the core specification of VVC.

Furthermore, VVC offers high-level syntax (HLS) features such as self-decodable sub-pictures for bit stream extraction and merge applications, or view-dependent streaming. Gradual Decoding Refresh (GDR) is another HLS feature adapted to low delay applications to avoid the burst of complete Intra pictures.

3. TEST CONDITIONS

The test conditions for the video coding evaluations reported in this paper are derived from the JVET Common Test Conditions (CTC) [8]. Two scenarios are considered:

- Broadcast scenario with one Intra picture approximately every second,
- Streaming scenario with one Intra picture approximately every two seconds.

The Broadcast scenario is the same as the “Random Access” case in JVET Common Test Conditions (CTC) [8]. It simulates a real scenario allowing a TV user to zap from a channel to another at an acceptable delay by inserting an intra-coded picture approximatively every second. The Streaming scenario is not part of JVET CTC, but the only difference with the "Random Access" case is the intra refresh period (around two seconds instead of one). This Streaming scenario simulates the video on-demand case for which AV1 has been designed and optimized. This point to point scenario requires adaptive bit rate (ABR) streaming to adapt to network bandwidth variation. The most deployed ABR protocol is MPEG-DASH (Dynamic Adaptive Streaming over HTTP) [9] which recommends segments of two seconds approximatively for switching between segments. Each segment is encoded at several bit rates or picture resolutions. Indeed, each encoded segment starts with an intra-coded picture for switching from a segment to another to adapt the bit rate.

The AV1 reference software (libaom) does not have the same configuration parameters as those used by JVET, but the settings chosen in the current evaluation were defined to ensure an as-similar-as-possible behavior. The reported results must anyway be interpreted with care as the reference encoders used for the evaluation may noticeably differ.

The nineteen video clips of JVET CTC referenced in [8], comprising six UHD (3840×2160), five HD (1920×1080), four WVGA (800×480) and four WQVGA (400×240) have been processed.

The following reference encoder software versions were used:

- HM-16.18 (HEVC Test Model), 2/1/2018,
- VTM8.0 (VVC Test Model), 2/24/2020,
- ETM4.1 (EVC Test Model), 12/20/2019,
- libaom (AV1 commit aa595dc), 09/19/2019.

The first three are up-to-date reference softwares representative of HEVC, VVC, and EVC respectively. The reference AV1 software (libaom) is stable in compression performance since this release. Complexity measures are the runtimes of encoder and decoder softwares executed in a single thread, on the same computer platform, to get comparable figures. Dedicated hardwares would indeed give different results.

4. VIDEO CODING CONFIGURATIONS

4.1 HEVC, VVC, and EVC

The Broadcast is based on the "Random Access" case with 10-bits sample representation, as specified in the JVET CTC [8]. The encoder configuration requires not more than 16 frames of structural delay which means a Group Of Picture (GOP) of 16 pictures. To accomodate to the different frame rates of each video clip, the intra period must be below 1.1 seconds for the Broadcast senario, and below 2.2 seconds for the Streaming scenario. The only difference between the two scenarios is the intra period.

For HEVC, VVC and EVC a hierachical GOP structure is used, with a constant quantization parameter per picture, increasing with the picture hierachical level. AV1 is configured to reproduce as far as possible similar settings, as described in the next paragraph.

4.2 AV1

4.2.1 Two-pass encoding

AOM recommends for the libaom software to run two-encoding passes to reach the best performance. The first pass is used to derive statistics on the full sequence that are further used to guide the second-pass encoding. It has been observed that the one-pass encoding in recent libaom software versions is less efficient than in past versions. The evaluation made on all the test sequences leads to the following results: the PSNR gain versus HM was −14.7% with two-pass encoding but 1.2% (small loss) with one-pass encoding. The encoder runtime versus the HM reference for two-pass encoding is 497%, while for one-pass encoding is 455%, meaning the first pass is light in processing compared to the second pass. The two-pass configuration provides a look-ahead to derive some encoding parameters that the VVC, EVC and HEVC encoding softwares have not. The impact of two-pass is analyzed in the section 5. It was noticed that the GOP structure when using one-pass encoding is very different from the hierarchical GOP structure used for VVC, EVC and HEVC. In libaom two-pass encoding, the GOP structure is hierarchical similar to the one used in the HM, ETM and VTM settings. Based on those observations, it was decided to use libaom with two-pass encoding at constant quality without rate control.

4.2.2 Quantization control

The libaom "End-usage" parameter defining the quantization control is set to "q", meaning a constant quality is achieved without rate control. Then "cq-level" fixes a base quantizer value on the full clip allowing to compute the BD-rate curves with a constant quantizer value per picture ("deltaq" parameter equal to 0) like in JVET CTC. The "aq" parameter (adaptive quantization for rate control) is not activated.

The meaning of “group of frames” (GOF) in AV1 can be compared to “group of pictures” in the other coding solutions. The GOF was fixed to 16, like in JVET CTC. The verification was done by instrumenting the libaom decoder to output the quantization parameter per picture. A similar type of quantization parameter offset appears per type of picture in a hierarchical GOF structure of 16 in libaom, like in the JVET CTC configurations. The results of this instrumentation are reported in [10].

In order to compute the gain (BD-rate [11]) of libaom with regards to the HM, the “cq-level” was fixed at 32, 40, 48 and 56, providing the range of bit rates to compute VTM and ETM gains.

4.2.3 Encoding parameters

The difference between the two test scenarios lies on parameters kf-min-dist and kf-dist-max which are the minimum and maximum distances between key frames (intra frames). For the Broadcast scenario both values are set equal to the number of frames in an integer number of GOPs of length below 1.1 second, while for the Streaming scenario, both values are set equal to the number of frames in an integer number of GOPs of length below 2.2 seconds.

The libaom software has a parameter to set the encoding speed, which is the inverse of the encoding algorithm quality. This parameter called “cpu-used” is fixed to 0 meaning the best encoding quality but also the slowest encoding time.

5. RESULTS

5.1 Objective quality

In the following tables the results are provided considering four different objective metrics.

PSNR (Peak Signal to Noise Ratio) is calculated as:

$$PSNR = 10log_{10}(Max^2 / MSE) \qquad (1)$$

with MSE being the Mean Square Error between the source and the decoded pictures, and Max the peak sample value of the content. PSNR is computed separately for the three components of each picture, then averaged across all pictures of a sequence ($PSNR_Y$, $PSNR_U$, $PSNR_V$).

In order to get an easier interpretation of the measured performance, a weighted sum of the PSNR on the three components Y (luma), U and V (chroma), for the complete sequence, is used as the first objective metric:

$$PSNR_{YUV} = (6\times PSNR_Y + PSNR_U + PSNR_V) / 8 \qquad (2)$$

This formula is commonly used as a global performance metric by video compression experts (see [3], [12], [14], [17]).

Additional results are provided with $PSNR_Y$ and two other well-known objective metrics with full reference (source pictures) which include some subjective factors:

- Multi-Scale Structural Similarity (*MS-SSIM*) [15],
- Video Multi-method Assessment Fusion (*VMAF*) [16].

As these two metrics are only using the luma component, the comparison has to be made with $PSNR_Y$. For each scenario the results are given with four tables corresponding to four objective metrics: $PSNR_{YUV}$, $PSNR_Y$, VMAF, and MS-SSIM. The tables below show the performance according to the “Bjøntegaard Delta-Rate” (BD-rate) metric (see [11] and [12]) in percentages with regard to the HM. It measures the bit-rate reduction provided by each solution at the same quality, here on four measures: $PSNR_{YUV}$, $PSNR_Y$, VMAF and MS-SSIM. The rate change is computed as the average percentage difference in rate over a range of Quantization Parameters (QP). A negative percentage represents a gain relative to the HM. The results are given as the average value on all the sequences but also split per picture size: UHD, HD, WVGA, WQVGA. The difference of the overall measure of ETM and libaom with VTM is the last line of each table.

5.1.1 Broadcast

Table 1 reports the $PSNR_{YUV}$ BD-rate variations, compared to the HM of the three tested solutions. It is observed that the VTM outperforms the other video coding solutions, achieving nearly 42% gain over the HM in the UHD format. The performance of all three solutions increases as the picture size increases. Over all picture resolutions, the ETM performed roughly 14.4% behind the VTM, and the libaom reference encoder 21.3% behind.

The performance of libaom can be discussed on the software maturity level. Clearly the libaom reference encoder, with two-pass encoding and encoding algorithms improvements brought in since 2018, is the most mature encoding solution. It has been observed that the highest quality configuration (cpu-used=0) had been accelerated significantly in encoding runtime from version to version. In 2017 [14] libaom was tested to lag significantly behind HM in its best two-passes configuration by −9.5%. The AV1 specification was release in March 2018. The libaom software corresponding to this version was used in [12] with two-pass, constant quality (cq) but with cpu-used=1 (not the best quality). The BD-rate gain over HM was 10%. The same year [17] measured 17% gain in the same configuration. In 2019 [18] tested the current libaom software with one-pass only on HD and UHD format resulting in the same performance as HM.

The overall gain reported in Table 1 of 14.7% for libaom over HM is lower than the gain of 17% reported in [17], but in the same order of magnitude, in very similar testing conditions. The difference could come from the disabling of quantization parameter variation within a picture (deltaq=0).

Table 1 – Broadcast:PSNR$_{YUV}$ BD-rate versus HM

Broadcast PSNR$_{YUV}$	VTM8	libaom	ETM4.1
UHD	−41.9%	−18.0%	−28.4%
HD	−39.0%	−16.3%	−21.0%
WVGA	−30.8%	−11.4%	−17.6%
WQVGA	−28.4%	−11.1%	−16.3%
Overall	**−36.0%**	**−14.7%**	**−21.6%**
Diff vs VTM	***0%***	***21.3%***	***14.4%***

Objective metrics in Table 2, Table 3, and Table 4 take only into account the luma component. It can be observed that the results in PSNR$_{Y}$ for libaom and ETM are very close to their PSNR$_{YUV}$, while VTM PSNR$_{YUV}$ is 3.6% above its PSNR$_{Y}$. This explains why the differences of ETM and libaom with VTM are smaller for VMAF and MS-SSIM. The superior performance of VTM chroma tools is not reflected. It should also be noted that only VMAF has a temporal dimension.

Table 2 – Broadcast: PSNR$_{Y}$ BD-rate versus HM

Streaming PSNR$_{Y}$	VTM8	libaom	ETM4
UHD	−38.7%	−18.0%	−28.2%
HD	−31.7%	−15.3%	−17.0%
WVGA	−28.8%	−13.5%	−16.7%
WQVGA	−27.3%	−14.1%	−16.0%
Overall	**−32.4%**	**−15.5%**	**−20.3%**
Diff vs VTM	***0.0%***	***16.9%***	***12.1%***

Table 3 – Broadcast: VMAF BD-rate versus HM

Broadcast VMAF	VTM8	libaom	ETM4.1
UHD	−42.8%	−19.6%	−33.9%
HD	−40.8%	−21.0%	−30.7%
WVGA	−29.2%	−12.5%	−20.3%
WQVGA	−30.7%	−19.5%	−22.1%
Overall	**−36.9%**	**−18.5%**	**−27.7%**
Diff vs VTM	***0.0%***	***18.4%***	***9.2%***

Table 4 – Broadcast: MS-SSIM BD-rate versus HM

Broadcast MS-SSIM	VTM8	libaom	ETM4.1
UHD	−39.5%	−14.5%	−29.9%
HD	−32.9%	−10.7%	−23.6%
WVGA	−28.1%	−7.2%	−17.2%
WQVGA	−25.0%	−6.1%	−15.5%
Overall	**−34.3%**	**−11.3%**	**−24.5%**
Diff vs VTM	***0.0%***	***23.0%***	***9.8%***

5.1.2 Streaming

In the Streaming scenario, Table 5 shows the VTM still performs better than the other solutions with approximatively the same figure (14.7%) against ETM as in the broadcast scenario. However, the gain of libaom over the HM is higher than in the broadcast scenario (3.6%), which is not observed for the VTM and ETM. One possible reason is that the two-pass encoding can take more benefits from a longer intra refresh period, by a better adaptation of the GOP structure, while for HM, VTM and ETM, the GOP structure remains static. It can be also noted that libaom software has a more consistent performance over the different picture resolutions in the streaming senario.

Table 5 – Streaming: PSNR$_{YUV}$ BD-rate versus HM

Streaming PSNR$_{YUV}$	VTM8	libaom	ETM4.1
UHD	−41.2%	−20.5%	−27.7%
HD	−36.8%	−18.5%	−18.1%
WVGA	−31.0%	−16.1%	−17.2%
WQVGA	−29.1%	−17.2%	−16.4%
Overall	**−35.3%**	**−18.3%**	**−20.6%**
Diff vs VTM	***0.0%***	***17.0%***	***14.7%***

Table 6, Table 7, and Table 8, which are based on metrics computed only on the luma component, show the same tendancy as in the broadcast scenario. The gap of ETM and libaom with VTM is smaller than on Table 5, but in a smaller proportion than in the broadcast scenario on VMAF and MS-SSIM.

Table 6 – Streaming: PSNR$_Y$ BD-rate versus HM

Streaming PSNR$_Y$	VTM8	libaom	ETM4
UHD	−38.7%	−18.0%	−28.2%
HD	−31.7%	−15.3%	−17.0%
WVGA	−28.8%	−13.5%	−16.7%
WQVGA	−27.3%	−14.1%	−16.0%
Overall	**−32.4%**	**−15.5%**	**−20.3%**
Diff vs VTM	***0.0%***	***16.9%***	***12.1%***

Table 7 – Streaming: VMAF BD rate versus HM

Streaming VMAF	VTM8	libaom	ETM4
UHD	−41.6%	−21.5%	−33.3%
HD	−38.5%	−22.6%	−24.6%
WVGA	−29.5%	−16.0%	−20.2%
WQVGA	−32.8%	−24.0%	−23.1%
Overall	**−36.4%**	**−21.2%**	**−26.1%**
Diff vs VTM	***0.0%***	***15.2%***	***10.3%***

Table 8 – Streaming: MS-SSIM BD-rate versus HM

Streaming MS-SSIM	VTM8	libaom	ETM4
UHD	−38.0%	−17.0%	−28.8%
HD	−30.4%	−13.6%	−14.8%
WVGA	−28.2%	−12.1%	−16.5%
WQVGA	−26.2%	−13.6%	−16.1%
Overall	**−32.9%**	**−14.5%**	**−20.9%**
Diff vs VTM	***0.0%***	***18.4%***	***12.0%***

5.2 Subjective quality

Formal subjective tests have not been conducted to compare these compression solutions, but an experts' viewing has been performed to collect some subjective observations. The broadcast case was taken to compare the HM at a given bit rate per sequence with VTM, libaom and ETM at approximatively the bit rate saving reported in Table 1 per resolution. The HM bit rate has been chosen at the bit rate point where artefacts may appear.

Globally the subjective quality of the tested solutions is better than the HM. For VTM and ETM the blocking artefacts visible in the HM on uniform areas with low level variations dissapear. However, on static textured areas some experts notice a loss of details which is not perceived as a degradation by others. For libaom, a smoother definition and loss of sharpness is globally observed.

Formal subjective tests are required to really evaluate the bit rate savings. JVET will conduct a formal evaluation of VVC with regards to HEVC as JCTVC did for HEVC with regards to AVC [3]. HEVC was measured at −44% at PSNR BD-Rate, but at −59% at MOS BD-Rate (MOS: Measure Of Satisfaction score in subjective tests).

5.3 Processing time

All simulations are run in single thread on the same platform in order to get comparable encoder and decoder runtimes.

The platform characteristics are:

Table 9 – Platform characteristics

CPU type	Intel(R) Xeon(R) Gold 6142
Hyper threading	Off
Turbo mode	On
Compiler	gcc 6.3.0
OS	CentOS7
SIMD options	SSE42

Table 10 and Table 11 report the runtime factor for encoding and decoding versus the HM-16.18 for all sequences. The runtime provides an estimate of the complexity. "N%" means "N/100" times the HM-16.18 runtime. The runtimes for the two scenarios, Broadcast and Streaming, are very similar.

Table 10 – Encoding runtime versus HM

Encoding	VTM8	libaom	ETM4.1
Broadcast	1308%	497%	669%
Streaming	1283%	515%	680%

Table 11 – Decoding runtime versus HM

Decoding	VTM8	libaom	ETM4.1
Broadcast	192%	76%	162%
Streaming	201%	77%	167%

The libaom software has the lowest runtime but as mentioned above, the maturity of the software is much higher than that of the others. Moreover, they are not using the same code optimization.

Each version of libaom software since March 2018 has improved in running time at constant quality. The ETM runtimes are lower than the VTM ones, which can be explained by a lower compression performance (less complex algorithms) but also by a different code base. The development process of VVC was driven by compression efficiency but also by considering implementability constraints, mainly on the decoder side. It can be observed that the decoder runtime (Table 11) is limited to 1.9 times the HM decoder. However, no specific effort was put on the reference encoder runtime (Table 10), which is 13 times the HM encoder with the VTM8 software, as encoding algorithms are non-normative.

6. CONCLUSION

The evolution of video compression has until now always been done incrementally, by building on top of the previous generation of video coding standards. The latest video coding solutions designed in ITU-T VCEG, ISO/IEC MPEG, and AOM followed the same path. In the test conditions considered in this paper, and using the reference encoders, VVC, developed jointly by ITU-T and ISO/IEC, provides the best objective measures of compression efficiency. EVC and AV1 are each significantly better than HEVC but also significantly lagging behind VVC. For future immersive services in UHD, the objective gain in bit rate at same quality versus HEVC is 42% for VVC, 28.4% for EVC and 18% for AV1 with the PSNR_{YUV} metric. One can argue that 18% gain for AV1 with lower complexity is interesting, but it must be reminded that the libaom software is suited for production, with tuned encoding algorithms using two passes for optimized subjective quality. VTM, ETM, and HM are reference softwares used in the standardization process but not suited for production, without the optimization of the subjective quality. The results provided by two other objective metrics, VMAF and MS-SSIM, are coherent with the PSNR_{YUV} results. The observed differences are due to the fact that VMAF and MS-SSIM do not take into account the chroma components, while PSNR_{YUV} does. However, the subjective quality is the most important metric which needs to be carefully studied to measure the performance of these video coding solutions.

Other criteria affecting choice will be the applications and services. AV1 is designed for on-demand video streaming types of service but VVC and EVC are more generic to cope with both broadcast and streaming cases. VVC offers versatility by meeting the requirements of higher compression efficiency on any type of content including 360° video for Virtual Reality (VR), High Dynamic Range (HDR), and computer graphics (Screen Content and Gaming). Furthermore scalability and RPR features provide tools for network bandwidth adaptation. The new sub-picture feature also offers support of the region-wise random access feature, which can be of particular interest for viewport dependent streaming of 360° video. These video coding standards have different announced licensing terms that could impact their deployment. AV1 is publicized to be royalty-free. The EVC contributors are claiming licensing terms will be available less than two years after the standard publication. VVC should follow the path of past video coding standards developed jointly by ITU-T, ISO and IEC such as H.264/AVC and H.265/HEVC. These standards have been successfully deployed as reported in [19] for HEVC.

REFERENCES

[1] Recommendation ITU-T H.264 (05/2003), Coding of moving video: Advanced video coding, ITU-T.

[2] Recommendation ITU-T H.265 (04/2015), Coding of moving video: High Efficiency Video Coding, ITU-T.

[3] Tan, Thiow Keng; Weerakkody, Rajitha; Mrak, Marta; Ramzan, Naeem; Baroncini, Vittorio; Ohm, Jens-Rainer; and Sullivan, Gary J. (2016), "Video Quality Evaluation Methodology and Verification Testing of HEVC Compression Performance", *IEEE Trans. Circuits and Systems for Video Technology*, Vol. 26, No. 1, pp. 76–90, January.

[4] JVET-Q2001 (2020), "Draft text of video coding specification (draft 8), text for DIS", 17th Meeting: Brussels, BE, 7–17 January.

[5] ISO/IEC JTC1/SC29/WG11 N18774 (2019), "Text of ISO/IEC DIS 23094-1, Essential Video Coding", Oct. 31st.

[6] AV1 specification (2018): https://aomediacodec.github.io/av1-spec/, March.

[7] Recommendation ITU-T H.265 (12/2016), Coding of moving video: High Efficiency Video Coding, ITU-T.

[8] JVET-M1010 (2019), "JVET common test conditions and software reference configurations for SDR video", 13th Meeting: Marrakech, MA, 9–18 January.

[9] MPEG-DASH ISO/IEC 23009-1 (2019).

[10] Michel Kerdranvat et al (2019), "Extra results to JVET-N605 "Comparative study of video coding solutions VVC, AV1 and EVC versus HEVC", JVET-O0898, 15th meeting: Gothenburg, SE, July 3-12.

[11] Gisle Bjøntegaard (2001), "Calculation of Average PSNR Differences between RD curves", ITU-T SG16/Q6 VCEG 13th meeting, Austin, Texas, USA, April, Doc. VCEG-M33 http://wftp3.itu.int/av-arch/video-site/0104_Aus/.

[12] Gisle Bjøntegaard (2008), "Improvements of the BD-PSNR model", ITU-T SG16/Q6 VCEG 35th meeting, Berlin, Germany, 16–18 July, Doc. VCEG-AI11http://wftp3.itu.int/av-arch/video-site/0807_Ber/.

[13] Julien Le Tanou (2018), "Analysis of Emerging Video Codecs: Coding Tools, Compression Efficiency and Complexity", MediaKind (Ericsson), SMPTE 2018 annual conference.

[14] Dan Grois et al. (2017), "Performance Comparison of AV1, JEM, VP9, and HEVC Encoders", HHI, Proceedings of SPIE Vol. 10396.

[15] Wang, Z.; Simoncelli, E.P.; Bovik, A.C (2004), "Multiscale structural similarity for image quality assessment", ACSSC Conference.

[16] Zhi Li et al (2016), "Toward A Practical Perceptual Video Quality Metric", Netflix TechBlog, June 6.

[17] C. Feldman (2018), "Best Video Codec: An Evaluation of AV1, AVC, HEVC and VP9", BitMovin, March.

[18] James Bruce et al. (2019), "Testing AV1 and VVC", BBC R&D, 22 October.

[19] Gary J. Sullivan (2020), "Deployment status of the HEVC standard", JCTVC-AL0020, Brussels, BE, 10–17 January.