SMPTE 2020 ATC Manuscript template

V-PCC: performance evaluation of the first MPEG Point Cloud Codec

Céline Guede, Architect, InterDigital**.**

Pierre Andrivon, Principal Scientist, InterDigital.

Jean-Eudes Marvie, Principal Scientist, InterDigital.

Julien Ricard, Architect, InterDigital.

Bill Redmann, Director of Standards, InterDigital.

Jean-Claude Chevet, Engineer, InterDigital.

Written for presentation at the

SMPTE 2020 Annual Technical Conference & Exhibition

Abstract. *Representation of 3D scenes is increasingly important in several industries by usingVirtual and Augmented Reality technologies. The point cloud format is well suited for such representations. Indeed, point clouds can be created with a simple capture process and modest processing, enabling a real-time, end-to-end point cloud distribution chain. However, point cloud compression is required to obtain data rates and files sizes that could be economically viable for industry. Standardization is required to ensure interoperability. In 2020, the International Organization for Standardization (ISO) Moving Picture Experts Group (MPEG) will publish a standard for its first point cloud codec, MPEG-I* Part 5: Visual Volumetric Video-based Coding (V3C) and Video-based Point Cloud Compression (V-PCC)*. This standard enables a world of new services and applications, including cultural heritage, telepresence, and new forms of entertainment.*

In this paper we review the principal use cases targeted by the V-PCC standard, we present the architecture of the V-PCC codec and describe its main tools, by giving insight on complexity at the encoder and decoder level and explaining profiles and conformance points in V-PCC. We then

present the methodology established, as a collaboration between industry and academics, for the evaluation of the V-PCC codec performance and the methodology's origins. This methodology was applied to the MPEG point cloud compression test model software (named TMC2) to consistently evaluate technologies proposed during the standardization process. Finally, we compare the performance of the main V-PCC tools available for lossy and lossless compression. Finally the conclusion provides elements in favor of a near-term deployment of V-PCC in the media industry.

Keywords. point cloud compression, immersive video coding, performance evaluation.

Figure 1. Combination of V-PCC tools in lossy configurations. From left to right: Original uncompressed point cloud, no enhancement (B0_MC1_RDP0_GS0), 2-Map with Geometry Smoothing (B1_MC2_RDP1_GS1), 1-Map with Single Pixel deInterleaving and Geometry Smoothing (B1_ MC1_SPI_RDP1_GS1), 1-Map with Point Local Reconstruction and Patch Border filtering (E2_ MC1_PLR_RDP1_PBF). Magnify frames to see the associated artifacts, especially on hands, shoes and faces. Basketball test content frame #1, see [1].

Introduction

Advanced 3D representations of the world are enabling more immersive forms of interaction and communication, and allow machines to understand, interpret, and navigate our world. 3D point clouds have emerged as an enabling representation for such information.

A point cloud is a set of points in a 3D space, each with associated attributes, e.g. color, material properties, etc. Point clouds can be used to reconstruct an object or a scene as a composition of such points. Point clouds can be captured using multiple cameras and depth sensors in various setups and can be made up of thousands up to billions of points to realistically represent reconstructed scenes.

Compression technologies are needed to reduce the amount of data required to represent a point cloud. As such, technologies supplying lossy but efficient compression of point clouds are needed for use in real-time communications. In addition, technology is sought for lossless point cloud compression in the context of dynamic mapping for autonomous driving, cultural heritage applications, etc.

Several use cases associated with point cloud data have been identified and corresponding requirements for point cloud representation and compression have been developed inside the MPEG community. These will be described in the section “Presentation of the V-PCC codec”.

The International Organization for Standardization (ISO) Moving Picture Experts Group (MPEG) standards address the compression of geometry and attributes such as colors and reflectance, scalable/progressive coding, coding of sequences of point clouds captured over time, and random access to subsets of the point cloud. The acquisition of point clouds is outside of the scope of this standard.

In 2020, ISO/MPEG will publish two standards for point cloud compression: V-PCC (Video-based Point Cloud Compression) and G-PCC (Geometry-based Point Cloud Compression) documented respectively in MPEG-I part 5 Video-based Point Cloud Compression and MPEG-I part 9 Geometry-based Point Cloud Compression of ISO/IEC JTC1/SC29 WG11.

This paper focuses on the V-PCC standard in both lossless and lossy configurations. An initial V-PCC codec architecture was submitted in response to the call for proposals in October 2017. This architecture was enriched by tools improving coding performance or minimizing 3D object reconstruction artifacts, thereby improving visual quality.

We describe the V-PCC codec architecture and provide an overview of its main coding tools in the Section “Presentation of the V-PCC codec”. We then present their performance and explain the methodology used for their comparison in the Section “Experimental results”. Finally, short-term deployment recommendations for the media industry are given in the Section “Presentation of the V-PCC codec”.

Presentation of the V-PCC codec

The democratization of 3D sensors, capable of capturing animated high definition representations of the world, induces strong needs with respect to applications providing real-time visualization of immersive videos.

For V-PCC, applications have been identified by the MPEG community in [2], including real-time immersive content viewing with interactive parallax for telepresence, Augmented Reality (AR) and Virtual Reality (VR), and 3D free viewpoint video, both shown in Figure 2.

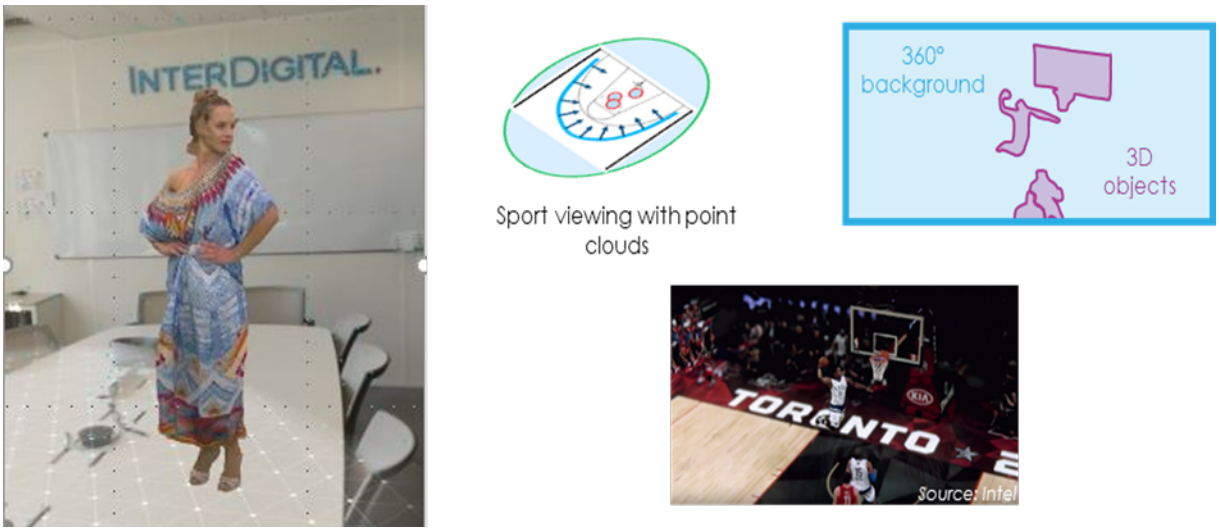


Figure 2. Point cloud use cases: (on the left), VR/AR telepresence, InterDigital source; (on the right) 3D free viewpoint sport replay broadcasting, Intel source.

3D immersive telepresence implies real-time communication and preserving as much realism as possible. Such an application requires a bitrate in the range of medium to low, real-time processing, with low latency (encoding, decoding, and rendering) and error resilience. 3D broadcasting of sports is another application for point clouds, in which the user can replay a part of the game from a preferred point of view. In this case, interoperability between manufacturers is important and low delay for encoding and decoding must be available.

The MPEG PCC standardization community defined three categories of point cloud test data to address all use cases identified in [3] and [4]: *static*, *dynamic*, and *dynamically acquired*. The V-PCC codec, described in this paper, handles the *dynamic* category, which corresponds to dense point cloud datasets varying in time.

Architecture of the V-PCC codec

The core encoding and decoding processes for video-based point cloud compression (V-PCC) were inherited from the solution that demonstrated the highest compression efficiency among all submissions, as agreed during the 119th MPEG meeting (Macau). The V-PCC solution is agnostic with respect to which 2D video codec is used. Hence, it can naturally leverage video codec evolution to take advantage of improvements in encoding techniques and performance. The test procedure described in this document leverages the state-of-the-art HEVC reference model (HM) [5] implementation for video-based coding.

Figure 3 shows a block diagram of the V-PCC process for encoding dynamic point clouds with color attributes while the decoding one is presented in Figure 4.

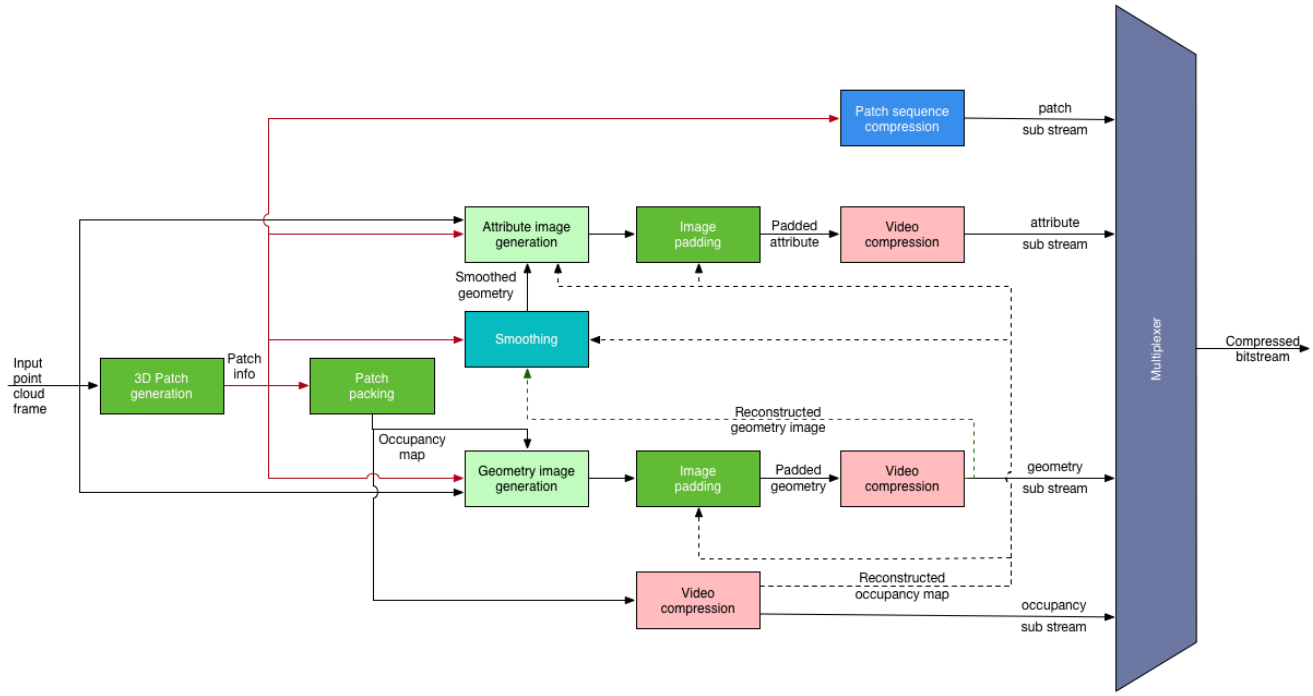


Figure 3. V-PCC TMC2 encoding structure from [6].

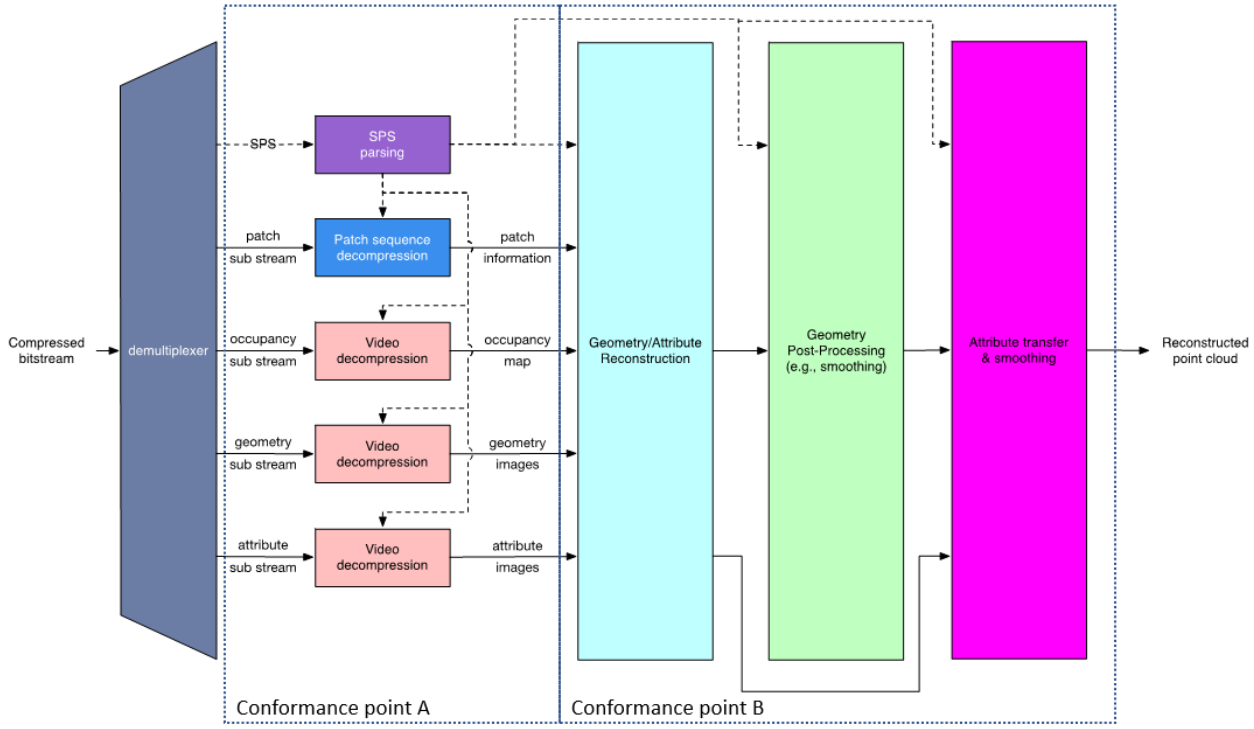


Figure 4. V-PCC TMC2 decoding structure from [6]. Conformance points A and B from [8].

Encoding

In this section we describe each step of the encoding process as depicted by Figure 3. At the encoding input point, each point cloud frame is processed as described hereafter.

3D patch generation is performed first (see Figure 3), in which the input point cloud is represented as a set of orthogonal 3D projections onto the six faces of a rectangular parallelepiped (an axis-aligned bounding box of the 3D object). These projections are made of connected components (CC), where each CC is a set of neighboring points having similar normals, as shown in Figure 5 (a).

(a) Clustering of 3D points

(b) Projection of each cluster in a 2D patch

(c) atlas showing packing of geometry (depth), color (attribute) and occupancy map onto 2D frames, with frame padding for color frame

Figure 5. V-PCC encoder main steps.

Each CC is projected onto one of the six bounding box faces, parallel to the main planes XY, XZ or YZ, by choosing which has a normal closer to the average normal of the points in the CC. The orthographic projection of the geometric information permits storing the distance of each point relative to the selected face. For instance, a point p = (x,y,z) of a CC that is projected to the XY plane would result in its value of z being projected to and stored in the (x,y) pixel of that XY plane.

Optionally, 12 additional projection planes with different orientations (see [7]) can improve the visual quality of the reconstructed point cloud by capturing more points during the projection phase. These additional planes are at 45-degrees along each edge of the bounding box and bring the total number of projections to 18.

Depending on the distribution of points in the CC, more than one value may be projected onto the same coordinate of the projection plane. A trivial approach would simply keep the value corresponding to the closest point (i.e. the smallest depth value), but this may not allow capturing more complex 3D point distributions (like folds in clothing). This trivial approach, called the “1-Map configuration”, supplies one map for the geometry and one for the attribute. A more sophisticated approach is to project several times and generate several maps for the geometry and corresponding maps for the attributes. In order to constrain the decoder

complexity for applications with low computation capabilities, V-PCC Basic profile limits the number of maps to at most two. However, V-PCC Extended profile permits to use a higher number of maps, for instance for applications requiring a higher fidelity. Keeping two depth values per coordinate of the projection plane produces what is called the “2-Map mode”. Whereas in the 1-Map case only the nearest (smallest) depth value is retained, in the 2-Map case, the farthest (largest) depth value is also kept. The 2-Map mode better captures the distribution of points in 3D space but at the expense of increasing the amount of projected data to be encoded. The projection step is shown in Figure 5 (b).

The “patch packing” process (see Figure 3) groups all the previously identified projected areas, also called patches, onto a 2D frame and fills this 2D frame in an optimal way (e.g. by minimizing the empty part and ensuring a temporal consistency) making sure that each patch packing block (e.g. 16 × 16 block) of the frame contains, at most, a single patch.

The result is the so-called atlas (see Figure 5 (c)), a trio of 2D frames constructed according to this projection scheme:

1. a geometry frame, which stores the CC depth values,
2. an attribute frame, which stores the color components (i.e. the texture),
3. plus an occupancy map, a binary image that indicates which parts of the geometry and texture frames are to be used for reconstruction stage.

A “frame padding” process (see Figure 3) fills the empty space between patches with the goal to make the frame better suited for video coding (see Figure 5 (c)). It is used on the geometry and attribute frames.

For lossy configurations, the occupancy map may be downscaled to reduce the number of bits needed to encode the occupancy map. For lossless encoding, the occupancy map must be kept at full resolution for the regeneration of the final point cloud (see Figure 5 (c)).

The 2D frames of each atlas obtained through this process, geometry, attributes and occupancy are finally video encoded using traditional video compression solutions in a separate way (see Figure 3). Some auxiliary data used for the reconstruction is also entropy coded.

At the end of the process, the separate bit streams are multiplexed into the output compressed binary V-PCC bitstream.

Decoding

The decoding process (see Figure 4) starts from demultiplexing the input compressed binary bitstream into geometry, attribute, occupancy map and auxiliary information streams.

The auxiliary information stream, containing information for reconstruction, is decoded. The occupancy map may be upscaled to its the nominal resolution if it was downscaled on the encoder side. The geometry stream is decoded and, in combination with the occupancy map and the auxiliary information, the point cloud geometry information is reconstructed and optionally smoothed along geometric patch boundaries.

Finally, the attributes of the point cloud are reconstructed based on the decoded attribute video stream, the reconstructed information for the geometry (smoothed geometry if present), the

occupancy map, and the auxiliary information. Afterwards, an additional attribute smoothing method may be used as a final point cloud refinement.

Presentation of V-PCC main coding tools with complexity considerations

An array of coding tools is described in detail in [6]. For this paper, the focus has been made on tools that visually and objectively impact the decoding and reconstruction of the point cloud in terms of objective performance metrics (refer to conformance point B described in Figure 4). The following is a list of these tools with a brief description.

Multiple maps (named MC for Map Count)

As mentioned in the section “Architecture of the V-PCC codec”, there is a 2-Map mode defined in V-PCC. In this configuration, maps keep both the nearest and farthest depth, spaced by at most a surface thickness. This surface thickness is defined on the encoder side and can be adjusted according to the reference source content (i.e., the original, uncompressed point cloud). It aids capture of opposed surfaces, such as the screen and the back of a tablet, without mixing them into the same connected component. This multiple-maps concept allows storage of overlapped points and better preserves the point geometries and occlusions in the 3D space.

When the 2-Map mode is set, the input point cloud frame is encoded with two maps for the geometry and two corresponding maps for texture. This implies a doubled memory consumption and requires synchronization since occupancy has one frame for two frames in geometry and texture. This could represent a limitation in some low-end devices.

The main syntax element for this tool into the standard [8] is vps_map_count_minus1[atlas_idx].

Single Pixel deInterleaving (SPI)

Single Pixel deInterleaving, detailed in [9], represents geometry frames for two maps in a single geometry frame (and likewise for texture) by interleaving values (in a quincunx shape) of the near and far depth maps. It is represented in Figure 6 as the “interleaved depth frame”.

Missing values can be deduced from the neighborhood on the decoder side. The process is performed in 2D and is only used for lossy configurations. For the decoder, this implies parsing the asps_pixel_deinterleaving_enabled_flag (see [8]), de-interleaving, and then upscaling the geometry and attribute information for each point cloud frame. Compared to a pure single-map representation, pixel interleaving enhances quality while preserving the performance advantage of decoding only one map.

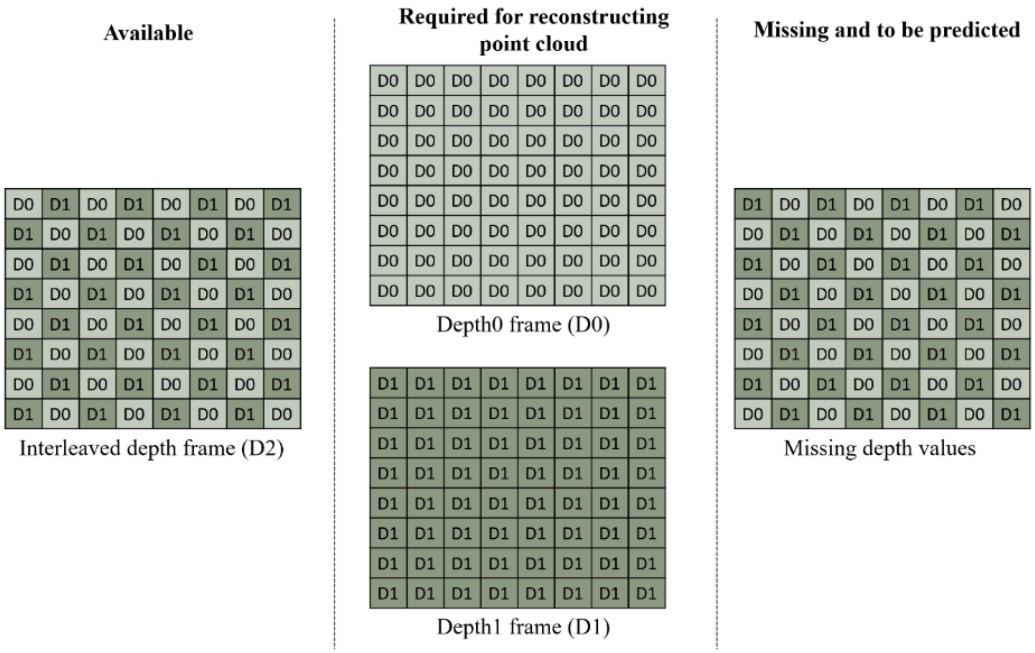


Figure 6. Example of geometry de-interleaving process and prediction.

Point Local Reconstruction (PLR)

Point Local Reconstruction tool, detailed in [10], improves the geometry of one map by adding points during the reconstruction. Compared to SPI, which builds interleaved maps, PLR is based directly on the near map, as described in the section “Encoding”. In addition, a prediction mode is defined as either block-by-block or patch-by-patch (with the best mode determinable by a rate distortion optimization method in the encoder) and is signaled in the stream to allow the decoder to generate points from the single map received. Figure 7 shows an input point cloud (in purple) that is represented in 2D with three different approaches: 2-Map mode, 1-Map mode, and 1-Map augmented with PLR. The 2-Map mode reconstructs points in 3D that correspond to the far (blue points) and near (yellow points) map. The 1-Map approach reconstructs only points of the near map (yellow points). PLR advantageously reconstructs the points of the near map (yellow points) but also points that could have been encoded for the far map (orange points) so as to be as faithful as possible to the reference source point cloud. The PLR process is performed in 2D and is only used in lossy configurations. The attributes for the new points are derived from the attributes of neighboring points. As with SPI, PLR enhances quality while preserving the performance advantage of decoding only one map. It is noted that, alike SPI, PLR may be used on several maps.

The main syntax element for PLR is asps_plr_enabled_flag and a small amount of data parsing is needed on the decoder side to know which mode has to be applied by the geometry reconstruction block, among those described in [10].

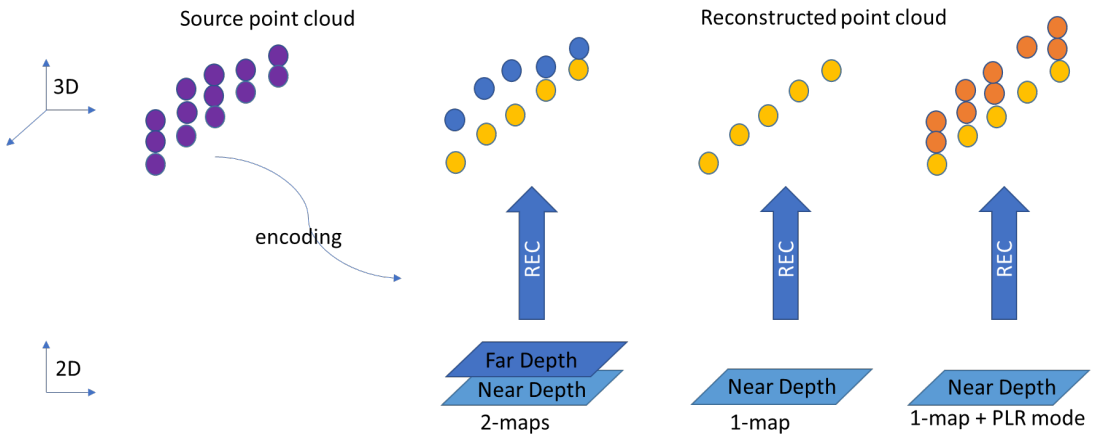


Figure 7. Point cloud reconstruction from left to right: 2-Map mode, 1-Map, and Point Local Reconstruction approaches.

Geometry Smoothing (GS)

Geometry Smoothing, detailed in [11], filters the points at the boundary of patches. For each point being identified as belonging to the patch boundaries, highlighted in red on Figure 8, it computes a centroid of the decoded points in a small 3D grid (see Figure 8, right) . After the centroid and the number of points in the 2 × 2 × 2 grid is derived, a commonly used trilinear filter is applied. This tool is only used for lossy configurations, since its purpose is to correct the artifacts introduced by down-sampling the occupancy map described in the section “Encoding”. The section “Comparisons regarding the patch filtering process” (below) shows the visual quality improvement provided by this tool. However, this is a multi-pass process operating in 3D space and the multiple passes over the reconstructed points may not be appropriate for devices having limited processing capabilities.

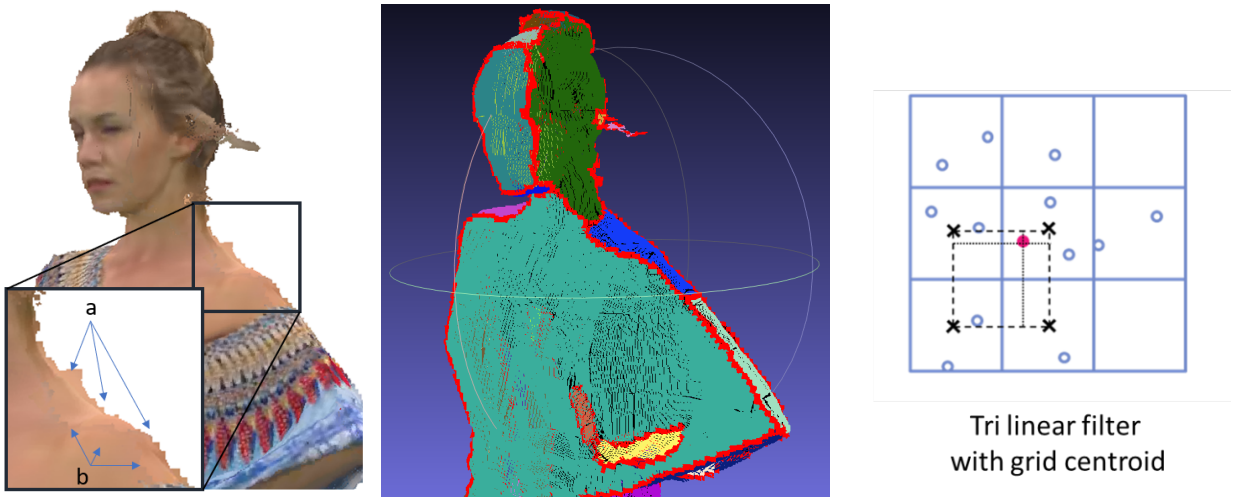


Figure 8. Points for geometry smoothing and trilinear filter from [6]. Left: geometric (a) and photometric (b) artifacts due to occupancy down-sampling. Center: patch boundaries are set in red where the geometry smoothing is applied. Right: the computation of the smoothed point, called centroid (in red), computed from nearest neighbors.

To be applied on the decoder side, the corresponding SEI message, geometry_smoothing(payloadSize), must be supplied with gs_method_type[k] set to 1.

Color Grid Smoothing (CGS)

The Color Grid Smoothing, detailed in [12], aims at averaging potential artifacts in the color values near patch boundaries. The process is done in 3D space so that the correct neighborhood is employed when producing the smoothed attribute value. This tool is only for lossy configurations and may be complementary to the geometry or ocupancy map filtering techniques.

Information on attribute smoothing is transmitted into an SEI message, attribute_smoothing(payloadSize).

Patch Border Filtering (PBF)

As explained in the two previous sections, smoothing patch borders removes some artifacts, due to the occupancy map up-sampling and video compression, by moving the reconstructed points. The patch border filtering process, detailed in [13], proposes an alternative way to correct the occupancy map before the reconstruction step. It has the advantage of being performed in 2D. For each patch, the borders of all adjacent patches are projected in the 2D space to adjust the frontier of the patch (see Figure 9). This process removes the border artifacts, ensures the alignment of two-adjacent patch borders, and avoids holes and any overlapping of points. As with geometry smoothing, this is used only in lossy configurations by design.

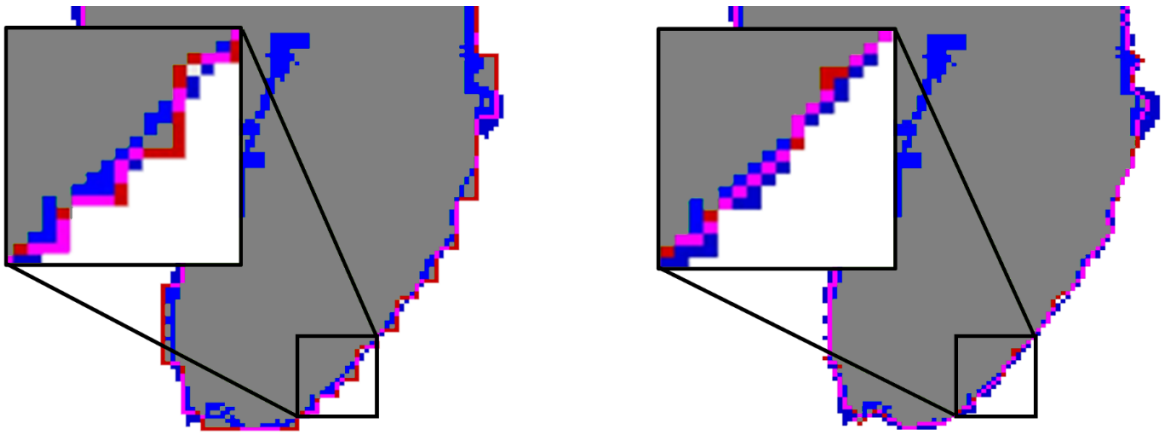


Figure 9. Example of patch pictures. The figures show how points are moved: patch points (grey), adjacent patch points (blue) and current patch border (red or purple (when above blue)) before patch filtering (left) and after (right).

To be applied on the decoder side, the corresponding SEI message, occupancy_synthesis(payloadSize) must be set with os_method_type[k] set to 1.

Remove Duplicate Points (RDP)

By default, the reconstruction process creates one 3D point per map for each non-zero value of the occupancy map, without looking at the coordinates of the points created. If the number of maps per atlas is greater than one, then wherever two depth values stored at same coordinates in two frames are equal, the reconstruction process will create two 3D points having the same coordinates. The process of removing duplicate points, detailed in [14], compares the pixel values in depth maps during the reconstruction process and does not create a second point if a pixel value in the far map is equal to the pixel value of the nearer map.

By reducing the number of points that must be treated, this filtering has a positive impact on the complexity and memory consumption subsequent processes.

Multiple Streams (MS)

The Multiple Stream tool indicates the number of video streams used to encode each of the geometry and the attribute frames. When this tool is activated, the geometry is composed of as many streams as the number of maps (typically two), likewise for attribute. Each supplementary stream can be, in that case, coded as a delta value compared with the first stream (i.e., the difference between the far and near depth) or as an absolute value. This is detailed in [10] and in [12].

When the Multiple Stream tool is active, it is better to use the delta coding mode for coding geometry and attribute information, as shown in section “Lossy 2-Map results” below. However, when the Multiple Stream tool is deactivated, the behavior is that of absolute coding. Indeed, in that case, the far depth frame has the same characteristics as the nearest depth frame and the coding becomes more suitable to be compressed as it uses inter-frame prediction supported by 2D video codecs.

This tool is signaled in the bitstream with vps_multiple_map_streams_present_flag. Some comparison results are given in Section “Lossy 2-Map results”.

Raw

The Raw tool, detailed in [15], allows coding of any remaining points that have not been gathered into a CC and projected in a specific patch. Instead, these points are aggregated into an identified “raw patch” for which the flag asps_raw_patch_enabled_flag signals the presence. A Separate Video (SV) flag, vps_auxiliary_video_present_flag[atlasIdx], when set to 1, indicates that raw coded geometry and attribute information for the atlas may be stored in a Separate Video stream (referenced as auxiliary video in the standard). When this flag is set to 0, then raw point patches are present in the same geometry and attribute video streams as the other patch types, as shown in Figure 10. This tool can be used in lossy configurations, but it is mandatory for lossless operations, so as to reflect all the information from the reference source point cloud concerning the otherwise unallied points. The quality gain provided by this Raw information comes at the expense of additional coding.

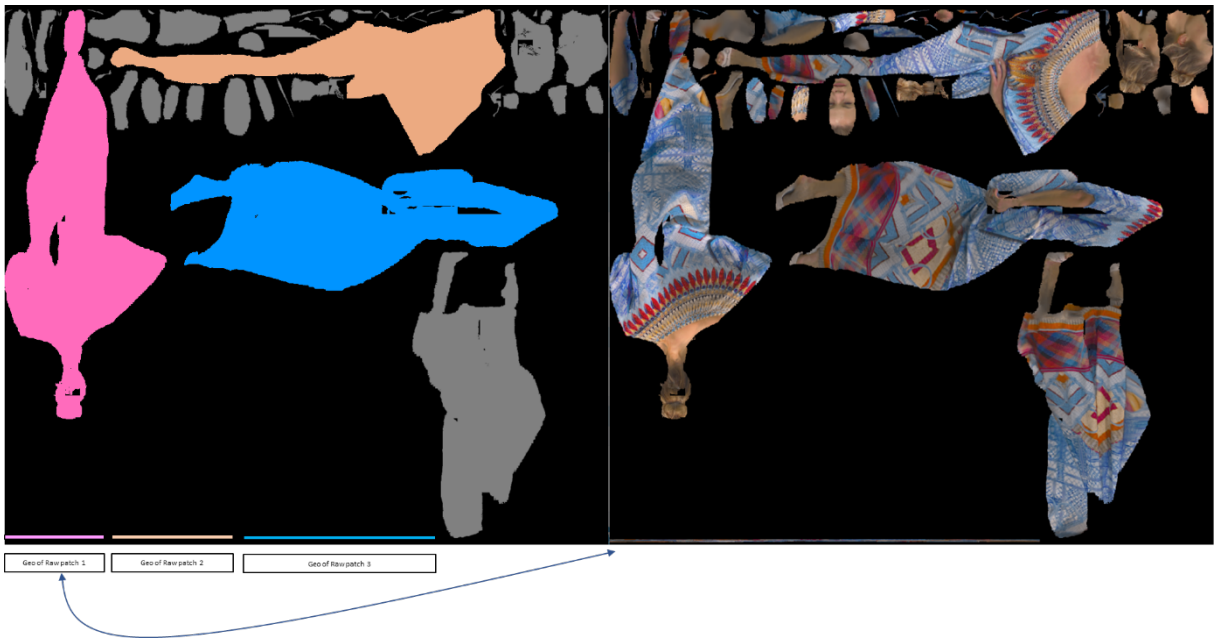


Figure 10. Raw points are typically coded below other projected patches. Example for a geometry map using false colors for illustrative purpose (on the left) and for an attribute map with frame padding disabled (on the right).

Enhanced Occupancy Map (EOM)

This tool, detailed in [16], proposes to handle “in-between points” that lie from the near and possibly up to the far maps (when it exists) by using a coded bit sequence (or codeword) in which each bit represents the projected occupation of points from immediately past the near map to the far map, as shown in Figure 11. The geometry of such intermediate depth positions is stored in the occupancy frame. The EOM can be associated with both the 1-Map and 2-Map configurations. In the case of a 1-Map configuration, the EOM code can define up to 15 positions from the projected depth. In the case of a 2-Map configuration, the number of EOM codes is limited to the surface thickness between the near and the far depth (e.g., a thickness of 4 is shown in Figure 11). The attributes associated to those EOM points are stored in the attribute frame. As with Raw patches, EOM attribute patches can be stored in a SV stream.

The usage of the Enhanced Occupancy Map tool is signaled with the flag asps_eom_patch_enabled_flag.

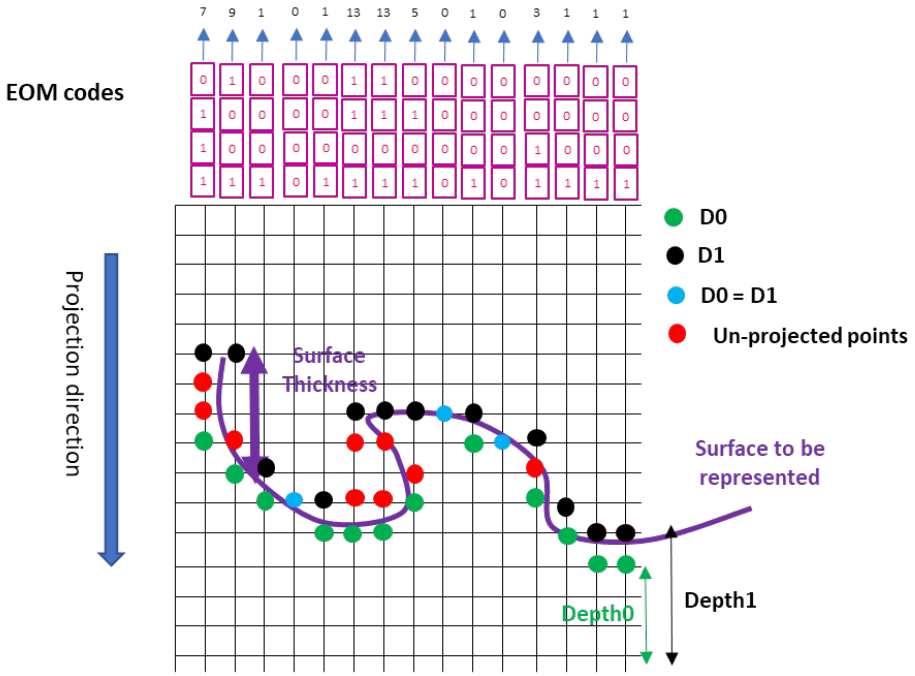


Figure 11. Enhanced Occupancy Map, 2-Map case, the near map (green), the far map (black). If near and far map is the same point (blue), EOM code is set 0; otherwise, each bit of the EOM code corresponds to the consecutively further in-between points (in red).

V-PCC profiles and conformance points

Essentially, a profile of a codec corresponds to a selection of tools that can be used by an encoder and must be implemented by a conforming decoder. A profile is expected to represent a category of industrial applications of the codec. The selection of tools in a profile is a trade-off between quality and bearable complexity for either the encoder or the decoder. This can be important where, for example, a low-end smartphone with capture and coding capabilities may not have the same coding or decoding performance as a high-end production workstation. Profiles can address diverse needs by emphasizing, for example, low complexity decoding, low latency distribution, or studio-grade quality.

A V-PCC profile is composed of a codec group, a toolset profile component (directed to decoding the atlas), and a reconstruction toolset profile component, see annex A of [8] for a comprehensive description. During the development of the V-PCC profile requirements, the MPEG 3DG group agreed that profiles need to cover at least the decoding of atlases with associated information but making the reconstruction process optional. Indeed, some companies were supportive of using proprietary tools for reconstruction to allow for differentiation and leveraging of algorithms or architecture of existing resources on the rendering device. Others wanted guarantees about the visual quality of the reconstructed point cloud. Thus, a peculiarity of V-PCC is that two conformance points are defined and are illustrated in Figure 4.

- The first, conformance point A, covers the decoded video sub-bitstreams and atlas metadata sub-bitstream. It also covers the patch sequence decompression. It does not,

however, cover the 2D to 3D reconstruction process. Conformance point A is bit-accurate, i.e., hard conformance.

- The second, conformance point B, covers the reconstruction process and is not bit-accurate: conformance point B is soft conformance.

In this paper, profiles covering conformance points A and B are considered (profile descriptions are given in [8]), and especially those enlisted in Table 1.

Acronyms	Profile Names
B0	HEVC Main10 V-PCC *Basic Rec0*
B1	HEVC Main10 V-PCC *Basic Rec1*
B2	HEVC Main10 V-PCC *Basic Rec2*
E0	HEVC Main10 V-PCC *Extended Rec0*
E1	HEVC Main10 V-PCC *Extended Rec1*
E2	HEVC Main10 V-PCC *Extended Rec2*

Table 1. Profiles naming convention. Acronyms are not part of the standard and are used in the "Experiments" section.

In the current version of the V-PCC standard, two toolset components (Basic and Extended) and three main reconstruction toolset components (Rec0, Rec1, Rec2) are defined. Readers should note that the numbers X suffixing the reconstruction profile "RecX" are merely enumerators and are not indicative of complexity, quality, or any such ordering.

The Basic profile component essentially does not enable EOM, PLR, extended projection or eight orientations. The Extended profile component removes these restrictions.

The Rec0 reconstruction profile component was designed so that the normative but optional reconstruction tools are ignored during the reconstruction process (i.e., the reconstruction process is performed with tools outside the scope of the standard). Rec0 may be adapted to either low-end devices or ecosystems with devices using proprietary reconstruction tools.

Rec1 and Rec2 reconstruction profiles enable most of standardized reconstruction tools, with the difference being that Rec1 comprises geometry grid-based smoothing while Rec2 comprises PBF (occupancy map pruning). Rec1 and Rec2 may be better suited to operators who prefer keeping control of the reconstruction visual quality or preserving artistic intent. Indeed, Rec1 and Rec2 tools output information computed from the original point cloud at the encoder side. It is noted that a decoder implementing Rec1 or Rec2 might not use signaled reconstruction tools.

Additionally, a fourth reconstruction toolset component (RecUnconstrained) allows any reconstruction tool and associated processing to be used, or not, for the reconstruction stage, whether or not it is signaled in the bitstream – user's choice.

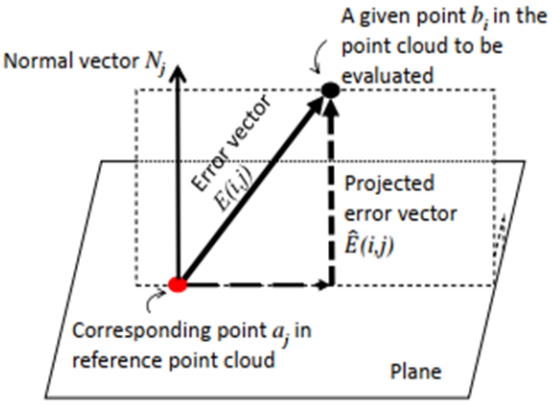
Experimental results

In this section we present some benchmark results for different coding tools. The tools are evaluated using a set of objective metrics described in [17] and [18], which include Peak Signal-

to-Noise-Ratios (PSNR) of a point-to-point error (D1 or point metrics) and a point-to-plane error (D2 or plane metrics) for geometry, as well as PSNR of color and reflectance attributes. These distortion metrics compare the original data with the reconstructed data and provide numerical values.

Geometric Distortions

For D1 and D2, both Mean Square Error (MSE) and PSNR are reported. Figure 12 shows how D1 and D2 are computed. For D1, the comparison is such that the Mean Square Error (MSE) between the reconstructed point b_i and the closest corresponding point a_j in the reference point cloud is calculated. For D2, the MSE is calculated between the reconstructed point b_i and the surface plane in the given reference test data. The reference test data provides surface normal information to facilitate the computation of the surface planes, avoiding potential variations in the choice of which face is used to project each CC. In the case of dynamic content, which corresponds to test configurations discussed below, the reported distortion measures are averaged over all coded frames and further reported by type (i.e., average over I-frames, average over P-frames, and average over all frames).



$$D1 = \frac{1}{N_A} \sum_{\forall a_j \in A} \|E(i,j)\|_2^2$$

$$D2 = \frac{1}{N_A} \sum_{\forall a_j \in A} \left(E(i,j) \cdot N_j\right)^2$$

Figure 12. Point-to-point distance (D1) versus point-to-plane (D2) distance from [17].

Attributes Distortion

Color distortion is measured in "YUV" space as 3 separate MSE distortions, which are reported as PSNRs for each channel: Y, U, and V. The reflectance distortion is likewise computed using MSE but only for a single component with the corresponding PSNR values being reported.

Subjective observations

Although subjective evaluation could not be conducted by the time of writing this manuscript, some frame captures are provided to show the effects of the coding tools. To render point clouds for subjective evaluation, the reference point cloud renderer, chosen by MPEG 3DG group [19], is used. The "cube" rendering method is used here. This method was chosen by the MPEG community to best assess the impact of compression without introducing additional filtering at the rendering stage.

Evaluation Methodology

Naming conventions

The tests are named in accordance with the following naming convention.

First, test names are prefixed with B[0,2] or E[0,2], representing the acronyms from Table 1, thereby specifying the targeted profile, as introduced in the section “V-PCC profiles and conformance points”.

Following the prefix, test names present a sequence of acronyms that identifies which coding tools are enabled. Note that when a tool is not mentioned in the test definition, a default value for the corresponding tool is induced, as listed in Table 2.

Acronyms	Coding tool	Default Value
MC1/MC2	The number of maps, where MC1 means only one map, whereas MC2 means two maps.	-
RAW	When mentioned, Raw tool is activated. Required for Lossless test.	Off
EOM	When mentioned, Enhanced Occupancy Map tool is activated.	Off
SV	When mentioned, Separate Video tool for coding RAW and EOM geometry and attribute information for the atlas in a separate video. It is only applicable if RAW or EOM tools are activated.	Off
GS0/GS1	Geometry Smoothing (GS0 means deactivated, GS1 means activated). For Lossless tests, GS is deactivated (GS0).	GS1
RDP0	RDP0 means Remove Duplicate Points tool is deactivated. RDP1 means RemoveDuplicate Points tool is activated	RDP1
CGS	When mentioned, Color Grid Smoothing is activated.	Off
SPI	When mentioned, Single Pixel deInterleaving is activated. When SPI is activated, MC1 is activated.	Off
PBF	When mentioned, Patch Border Filtering is activated. If PBF is activated, Geometry Smoothing is set to GS0	Off
PLR	When mentioned, Point Local Reconstruction is activated. With PLR activated, MC1 is activated.	Off
MS_AT11_AD11	Multiple Streams tool is activated for coding geometry and attribute video streams. In this case, geometry and attribute of supplementary streams are coded as absolute values.	Off
MS_AT10_AD10	Multiple Streams tool is activated for coding geometry and texture video streams. In this case, geometry and texture of supplementary streams are coded as delta coding.	Off

Table 2. Coding tools naming convention.

Test configurations

The software used to conduct the experiments is the test model developed within the MPEG community, called TMC2 (Test Model for Category 2 see [20]). Revision 10.0 is available in [21] and corresponding results are shown in [22].

Tests are run on each of seven 32-frame sequences, in both lossless and lossy configurations, to evaluate the performance of different coding tools using the objective metrics.

Table 3 and Table 4 show for these seven sequences a summary of compression ratios and bitrates, respectively for lossless and lossy configurations, relative to the MPEG "anchor" (i.e., reference) configuration (which is Basic.Rec1 with Geometry Smoothing and Remove Duplicate Points tools enabled, see [23]). The right-hand portion of each table shows the average,

minimum, and maximum bits per input point (bpip) of all sequences for the corresponding coding structure.

	basketball	dancer	longdress	loot	queen	redandblack	soldier	Average bpip	Min bpip	Max bpip
Lossless AI	16,34	16,66	27,76	16,51	15,13	23,76	18,55	10,67	8,17	14,99
Lossless LD	16,33	16,65	27,59	16,38	13,58	23,64	17,99	10,48	7,33	14,90
Lossy AI	0,34	0,39	1,46	0,59	0,63	0,85	0,92	0,41	0,12	0,92
Lossy RA	0,20	0,26	0,70	0,22	0,23	0,47	0,27	0,18	0,06	0,45

Table 3. Compression ratio per sequence in percentage with respect to original uncompressed point clouds (i.e. basketball lossless AI represents 16% of original test model size). Followed by bits per input point (bpip) average, min and max among all sequences – lossless (E1_MC2_RAW_EOM) lossy (B1_MC2_RDP1_GS1) 32 frames TMC2 R10.0. AI stands for All-Intra, LD stands for Low-Delay and RA stands for Random-Access.

	R01	R02	R03	R04	R05	Average bpip	Min bpip	Max bpip
Lossy AI	4,44	6,72	10,84	18,81	34,00	0,41	0,12	0,92
Lossy RA	2,38	3,24	4,80	8,30	17,34	0,18	0,06	0,45
Average	3,41	4,98	7,82	13,55	25,67	0,30	0,09	0,69

Table 4. Average Lossy Compression Bitrate (Mbps) on all sequences for each pre-defined rate (R01..5 from Common Test Conditions, see [23]) followed by bits per input point (bpip) average, min and max among all rates – B1_MC2_RDP1_GS1 32 frames TMC2 R10.0. AI stands for All-Intra and RA stands for Random-Access.

Table 4 reveals two key pieces of information: the first is the bitrates achieved by the V-PCC test model with respect to coding structure (All-Intra, AI and Random-Access, RA); the second is the average, minimum, and maximum bpip for all rates by coding structure. The takeaway from Table 4 is that bitrates achieved by V-PCC (using HEVC as an underlying video codec) are compatible with typical distribution networks deployed today.

Objective performances

For the experiments presented in this section, four test families are identified: lossless, the lossy 1-Map tool (MC1), the lossy 2-Map tool (MC2), and the lossy 1-Map versus 2-Map tool. The last family preserves remarkable configurations from the two previous families.

For each test, a table presents the gain (negative value in green) or loss (positive value in red) as a percentage relative to the anchor (the first line on each table) for the different metrics.

For the lossless family, Table 5 presents four data points concerning the bit per input point ratio (*bpip*) metrics: the whole bitstream (*bpip total*), geometry plus data (*bpip data+geometry*), geometry only information (*bpip geometry*), and attributes only information (*bpip color*).

For the three lossy families, the corresponding tables (Table 6-Table 8) present the ratios of the point metrics (*point/D1*), plane metrics (*plane/D2*), luminance (*Y*), and chrominance (*U and V columns*). These metrics are introduced in sections Geometric Distortions and Attributes Distortion.

For all of the families, the time ratios, in percentage gain, consumed inside the encoder (*self time enc*) and decoder (*self time dec*) are shown, as well as the memory consumption in each of the encoder (*gain mem enc*) and decoder (*gain mem dec*).

A discussion follows each test series. In the last section, some screen captures are presented, highlighting some visual effects corresponding to the coding tools.

Lossless results

Here the anchor is the B1_MC2_RAW, which is a 2-Map configuration with the Raw tool activated.

The lossless results in Table 5 show that the Separate Video tool (SV) does not affect the "bpip" much, only a marginal gain of 0.08% for the whole bitstream (*bpip total*), and modest gains for both time and memory usage, with a gain of 0.46% on decoding time and 3.38% on the decoder memory time (*self time dec* and *gain mem dec*, respectively) achieved. When SV is activated, it implies the use of several (real or virtual) encoders/decoders. Although this can be a burden on today's lower-end devices, SV has the advantage of allowing an independent configuration of the encoders for geometry and RAW / EOM information, as these tools could also be used for lossy configurations.

Experiments	bpip total	bpip data + geometry	bpip geometry	bpip color	self time enc	self time dec	gain mem enc	gain mem dec
B1_MC2_RAW	0	0	0	0	0	0	0	0
B1_MC2_RAW_SV	-0,08	0,29	0,31	-0,15	-2,1	-0,46	-1,83	-3,38
E1_MC2_RAW_EOM	-2,71	-18,05	-20,08	-0,44	-3,42	-7,23	1,22	-1,01
E1_MC2_RAW_EOM_SV	-2,76	-17,93	-19,95	-0,51	-5,11	-8,28	0,71	-2,13
E1_MC1_RAW_EOM	1,34	-3,81	-99,55	2,18	-6,06	-21,49	-8,09	-13,44

Table 5. Metric and performance ratios of lossless tools compared to the anchor B1_MC2_RAW (in percentage).

The Enhanced Occupancy Map tool (E1_MC2_RAW_EOM) performs better with a gain of 2.71% for the whole bitstream (*bpip total*). The benefit occurs essentially because of the geometry coding gain of 20.08% (*bpip geometry*). This tool is less time consuming, showing a gain of 7% on the decoder decoding time (*self time dec*) and has no big impact on memory consumption.

When combined with the Separate Video tool, EOM is the tool that performed the best of all tested lossless configurations in terms of performance (E1_MC2_RAW_EOM_SV).

Comparing two maps configurations against one (E1_MC2_RAW_EOM and E1_MC1_RAW_EOM), the 1-Map configuration shows a loss in lossless configuration, since it does not benefit from the inter prediction of the 2D video encoder. Indeed, as attribute information (*bpip color*) represents most of the data size to encode, the advantage of the gains in geometry is not impacting *bpip total*. The 1-Map coding tool (MC1) has a positive impact on computation performance, showing gains of 21% for the decoding time (*self time dec*) and 13% for the decoder memory consumption (*gain mem dec*).

Lossy 1-Map results

Experiments	point/D1	plane/D2	Y	U	V	self time enc	self time dec	gain mem enc	gain mem dec
B0_MC1_RDP0_GS0	0	0	0	0	0	0	0	0	0
B1_MC1_RDP0_GS1	-1,14	-4,81	0,89	0,34	0,68	0,33	73,02	0	0,61
B1_MC1_SPI_RDP1_GS0	-36,02	5,19	-0,74	3,36	2,73	40,93	66,38	37,86	55,96
B1_MC1_SPI_RDP1_GS1	-47,6	-21,3	-0,22	3,78	3,33	45,91	177,73	37,84	56,77
B2_MC1_RDP0_PBF	2,71	-3,02	0,61	-0,04	0	0,28	-3,24	2,5	-6,18
E1_MC1_PLR_RDP1_GS0	-34,65	3,16	2,33	7,05	6,13	46,91	67,99	37,87	55,97
E1_MC1_PLR_RDP1_GS1	-47,14	-21,08	2,89	7,42	6,81	47,18	175,93	37,86	56,79
E2_MC1_PLR_RDP1_PBF	-46,49	-23,24	2,67	6,86	5,91	45,44	62,02	38,55	45,17

Table 6. Metric and performance ratios of 1-Map tools compared to the anchor BO_MC1_GS0_RDP0 (in percentage).

Here the anchor is the B0_MC1_RDP0_GS0, which is a 1-Map configuration with no other tool.

The Geometry Smoothing tool (B1_MC1_RDP0_GS1) slightly improves point and plane metrics, at the expense of increased complexity of about 73% at the decoding level (*self time dec*). When combined with SPI or PLR, the impact of Geometry Smoothing on the decoder's performance is very strong, multiplying by more than 2.6 the impact of the decoder's performance, e.g., 177% instead of 66% for the SPI tool (*self time dec)*. This impact on SPI and PLR are comparable to the one observed when the 2-Map tool is activated (see section "Lossy 2-Map results"). The major interest of this tool is to significantly reduce aliasing effects at patch borders (see Comparisons regarding the patch filtering process and Geometry Smoothing (GS) sections).

The Patch Border Filtering process (B2_MC1_RDP0_PBF) has results similar to the anchor, in term of metrics and complexity. Like geometry smoothing, this tool has the strong benefit of reducing aliasing effects along patch borders. One notable advantage, compared to the Geometry Smoothing, is its low complexity (*self time dec*) and low memory consumption (*gain mem dec*), demonstrating performance even better than the anchor.

Single Pixel deInterleaving (B1_MC1_SPI_RDP1_GS0) and Point Local Reconstruction (E1_MC1_PLR_RDP1_GS0) are tools that enhance the reconstructed point cloud with a remarkable gain of 35% in the point metric at a cost of around 70% in complexity on the decoder side. However, SPI and PLR show a loss in the *Y, U, V* metrics, with the PLR tool showing a little bit greater loss.

Geometry results are even better when map-enhancement and filtering tools are combined (B1_MC1_SPI_RDP1_GS1, E1_ MC1_PLR_ RDP1_GS1, E2_ MC1_PLR_ RDP1_PBF). Geometry Smoothing and Patch Border Filtering improve point metrics by about 12% and plane metrics by about 15%. Compared with the anchor, one advantage of the Patch Border Filtering tool is that it adds neither complexity (*self time enc/dec*) nor memory consumption overhead (*gain mem enc/dec*) when combined with PLR (E2_ MC1_PLR_ RDP1_PBF versus E2_ MC1_PLR_ RDP1_GS0).

Lossy 2-Map results

Here the anchor is the B0_MC2_RDP0_GS0, which is a 2-Map configuration with no enhancement tool. As seen from Table 7, the Remove Duplicate Point tool (RDP) does not bring significant metrics improvements when only the 2-Map configuration is activated (B1_MC2_RDP1_GS0) and has a modest gain of 5% in *self time dec*. However, when

combined with other tools (e.g. Geometry Smoothing: B1_MC2_RDP0_GS1 versus B1_MC2_RDP1_GS1), it decreases the decoding complexity by about 34% (*self time dec*) as fewer points are handled downstream in the reconstruction process.

As with the 1-Map configurations, Geometry Smoothing (B1_MC2_RDP0_GS1) improves both point and plane metrics, here by 19.85% and 24.77% respectively, at the expense of increasing complexity by 89% on the decoder (*self time dec*). The major gain from this tool is to significantly reduce aliasing effects at patch borders (see Comparisons regarding the patch filtering process section).

Experiments	point/D1	plane/D2	Y	U	V	self time enc	self time dec	gain mem enc	gain mem dec
B0_MC2_RDP0_GS0	0	0	0	0	0	0	0	0	0
B1_MC2_RDP1_GS0	0,01	-0,04	-0,32	0,13	0,16	-0,4	-5,76	-7,5	-14,06
B1_MC2_RDP0_GS1	-19,85	-24,77	1,09	0,3	0,54	0,21	89,03	-0,02	0,72
B1_MC2_RDP1_GS1	-19,36	-23,56	0,92	0,5	0,78	-0,19	55,24	-7,51	-13,66
B1_MC2_RDP1_GS1_CGS	-19,36	-23,55	92,93	126	139,56	0,16	76,39	-6,61	-7,66
B1_MC2_RDP1_GS1_MS_AT11_AD11	30,83	28,17	64,29	72,33	71,89	79,27	69,02	-6,22	-10,83
B1_MC2_RDP1_GS1_MS_AT10_AD10	30,06	24,62	12,1	15,61	14,91	66,44	77,15	-0,01	-4,93
B2_MC2_RDP0_PBF	-18,79	-27,4	0,46	-0,21	-0,26	-1,99	-4,94	1,24	-5,8
B2_MC2_RDP1_PBF	-18,71	-27,37	0,04	-0,05	-0,12	-0,89	-7,78	-6,36	-19,98

Table 7. Metric and performance ratios of 2-Map tools compared to the BO_MC2_RDP0_GS0 (in percentage).

The Multiple Stream (both MS lines) process produces a loss (30% in point metrics and 25% in plane metrics), but it brings an alternative for V-PCC bitstream representation. Notably, in the case of Multiple Streams, there is a strong benefit to the delta coding option (AT10_AD10), especially on color metrics (*Y, U, V*).

Although, the Color Grid Smoothing (CGS) tool does not bring benefits in term of these color metrics, there are claims that it brings subjective visual improvements.

One remarkable tool is the Patch Border Filtering (B2_MC2_RDP0_PBF), which performs better than the anchor in terms of all metrics and complexity. This tool yields a 18.79% gain in the point metric, 27.4% in plane metric, and though holding roughly the same in *Y, U, V, it* has a gain of 2% on the encoder (*self time enc*), 5% on the decoder (*self time dec*), and 6% for decoder memory usage (*gain mem dec*). Here again, combining the Patch Border Filtering and the Remove Duplicate Point tools reduces the complexity and memory consumption on both the encoder and decoder sides.

Lossy 1-Map vs 2-Map results

In order to be able to compare 1-Map versus 2-Map tool, the best previous tests from the respective families are collected and listed in Table 8.

Experiments	point/D1	plane/D2	Y	U	V	self time enc	self time dec	gain mem enc	gain mem dec
B0_MC1_RDP0_GS0	0	0	0	0	0	0	0	0	0
B0_MC2_RDP0_GS0	-36,42	3,69	0,27	5,95	4,9	53,88	74,13	36,16	90,75
B1_MC1_SPI_RDP1_GS1	-47,6	-21,3	-0,22	3,78	3,33	45,91	177,73	37,84	56,77
B1_MC2_RDP1_GS1	-46,31	-22,02	0,43	6,44	5,64	53,59	170,32	25,93	64,7
B2_MC2_RDP1_PBF	-44,67	-24,98	0,26	5,88	4,73	52,51	60,58	27,5	52,63
E1_MC1_PLR_RDP1_GS1	-47,14	-21,08	2,89	7,42	6,81	47,18	175,93	37,86	56,79
E2_MC1_PLR_RDP1_PBF	-46,49	-23,24	2,67	6,86	5,91	45,44	62,02	38,55	45,17

Table 8. Metric and performance ratios of 1-Map vs 2-Map tools compared to the BO_MC1_GS0_RDP0 (in percentage).

Here the anchor is the B0_MC1_RDP0_GS0, which is a 1-Map configuration with no enhancement tool.

Comparing these two families leads to several observations. First, all selected tools bring a gain in the geometric metrics (*point* and *plane*) with a compromise on the color (around 5%). This net gain in quality comes at the cost of an increase in complexity for both the encoder and decoder (*self time enc*, *self time dec*), including their memory consumption (*gain mem enc*, *gain mem dec*).

The Geometry Smoothing tool (B1_MC2_RDP1_GS1) delivers a gain of 46.31% for the *point* metric and 22.02% for the *plane* but with performance strongly affected on the decoder side: 170% loss in self decoding time. This loss persists even when combined with other tools like SPI or PLR.

The PBF tool (B2_MC2_RDP1_PBF) brings a 44.67% gain in the *point* metric and a 24.98% gain for the *plane,* at an increase of 60% in complexity (*self time enc* and *self time dec*) compared to the anchor. Coupled with the Point Local Reconstruction (E2_PLR_RDP1_PBF), the Patch Border Filtering tool sees PLR's advantages, keeping a low complexity, and benefits from the visual improvement as described in next section.

Some visual comparisons on lossy configurations

In this section we present some screen captures to highlight various tool benefits. Screen captures are taken from the MPEG 3DG test content called "longdress", at the middle rate R03 targeted in the Common Test Conditions (see [23]). The original uncompressed test point cloud (reference source model) was provided by 8i (see [24]) and is shown in Figure 13.

Figure 13. Longdress reference source model frame #1051 overview and closeups.

Comparisons regarding the patch filtering processes

Here, using 2-Map configurations, the comparison is among the tools that impact patch border alignment, namely Geometry Smoothing (GS) and Patch Border Filtering (PBF). Reference source model and 2-Map without smoothing are presented as the basis for comparison. As seen in Figure 14, filtering the patch border, with GS1 or PBF, significantly increases the final point cloud quality as it strongly reduces patch border aliasing (see shoulder areas). The benefit is two-fold. First, the geometry artifacts are reduced, leading to more accurate silhouettes and surfaces. Second, the photometric continuity is better preserved at patch boundaries.

Figure 14. Smoothing tools. From left to right: reference source, no smoothing (B0_MC2_RDP0_GS0), Geometry Smoothing on (B1_MC2_RDP1_GS1), Patch Border Filtering on (B2_MC2_RDP1_PBF).

Comparisons regarding the number of maps

Figure 15. Number of maps tools. From left to right: reference source, 1-Map (B0_MC1_RDP0_GS0), 2-Map (B0_MC2_RDP0_GS0).

For this evaluation, the comparison is between tools that change the number of maps, say B0_MC1_RDP0_GS0 for 1-Map and B0_MC2_RDP0_GS0 for 2-Map. The results, as shown in Figure 15, demonstrate that increasing the number of projected points, thanks to the use of several maps, visually enhances the reconstructed point cloud. A 2-Map configuration captures More accurately the distribution of points in 3D space; fewer holes appear; and surface continuities are better preserved, even if there are still artifacts due to the lossy nature of the compression (see Multiple maps (named MC for Map Count) section).

Comparisons regarding the map enhancement tool

In Figure 16, using the 1-Map tool, the comparison is among the map enhancement tools: Single Pixel deInterleaving (SPI) and Point Local Reconstruction (PLR). The reference and unfiltered 1-Map are given as the basis for comparison. As can be seen, artificially increasing the number of points at the point cloud reconstruction stage by using SPI or PLR leads to significant visual enhancements, especially at the surface continuity level. In this set of experiments, PLR generally led to the highest quality visual results, at the cost of few extra data to be transmitted, though still showing artifacts due to lossy compression.

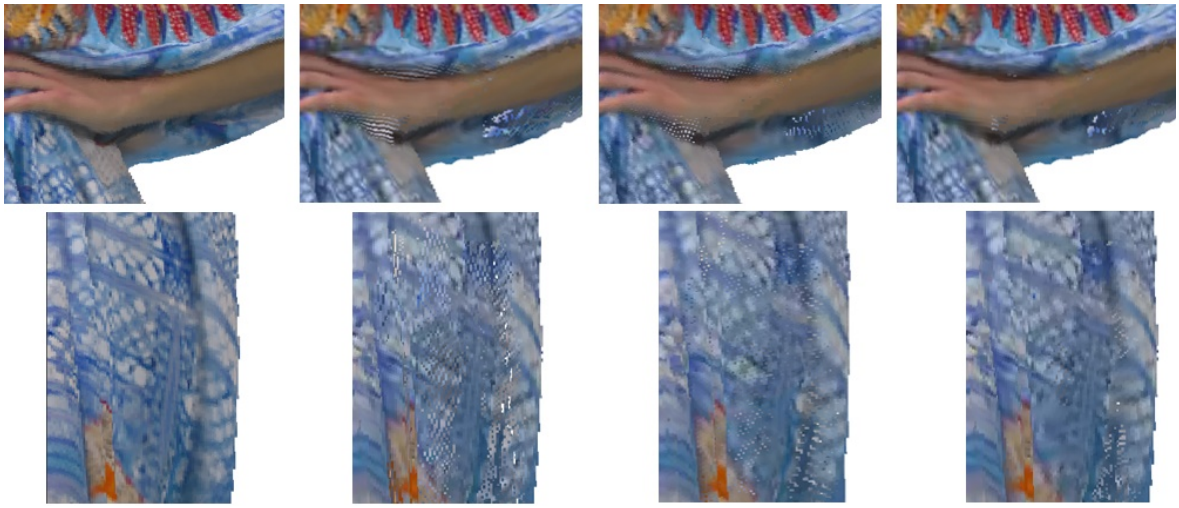


Figure 16. Map enhancement tools. From left to right: reference source, 1-Map no enhancement (B1_MC1_RDP0_GS0), 1-Map and Single Pixel deInterleaving active (B1_ MC1_SPI_RDP1_GS0), 1-Map and Point Local Reconstruction active (E2_ MC1_PLR_ RDP1_GS0).

Comparisons regarding the combination of tools

Finally, Figure 17 illustrates the benefits to the reconstruction of combining the tools together. While some tools focus on patch border filtering and trying to reduce the aliasing effect on geometry and photometry, other tools contribute to the advantage of filling holes introduced by compression, whether resulting from unprojected points or 2D compression errors. In most cases shown, one can see that superior visual results are achieved by using 2-Map or 1-Map using PLR with PBF. The choice of which combination to use would thus be made according to the targeted device performance (see the previous section Lossy 1-Map vs 2-Map results).

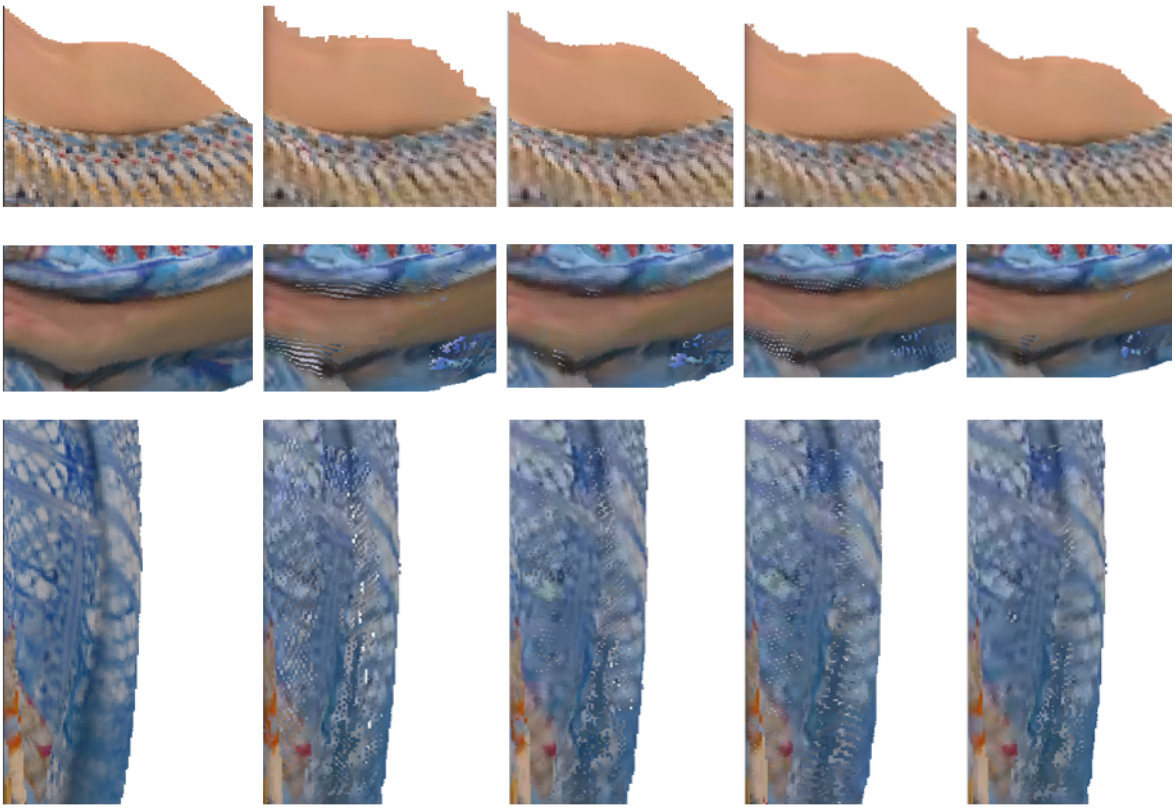


Figure 17. Combination of tools. From left to right: reference source, no enhancement (B0_MC1_RDP0_GS0), 2-Map with Geometry Smoothing (B1_MC2_RDP1_GS1), 1-Map with Single Pixel deInterleaving and Geometry Smoothing (B1_MC1_SPI_ RDP1_GS1), 1-Map with Point Local Reconstruction and Patch Border filtering (E2_ MC1_PLR_ RDP1_PBF).

Conclusion

The V-PCC standard is the first MPEG codec for point cloud compression. Its promotion by ISO to International Standard is planned for the end of 2020. V-PCC was initiated with near-term deployment in mind; therefore, its architecture was designed to rely upon conventional 2D video codecs. It thus leverages the large base of video codecs already deployed in consumer electronics devices (e.g. AVC, HEVC...) and, at the same time, it is expected to directly scale up in performance thanks to coming generation of video codecs (e.g. VVC).

This document has presented the main tools implemented in the V-PCC standard and their performance (i.e., quality for a given bitrate at an acceptable complexity for mass market devices) in the V-PCC MPEG test model. Although performance may differ for an optimized product implementation according to the targeted platform architecture and capabilities, the V-PCC test model is the platform used By the MPEG group to make the decisions to include tools in the V-PCC standard, based upon a balance between complexity and signal distortion. The tools evaluated enable reconstruction that maintains fidelity to the original point cloud or that provides bitrate reduction compatible with typical user network capabilities, at a few megabits per second. It is interesting to note that according to the performance results, the Extended

profile tools may not require additional computational resources as compared to the Basic profile tools, and can even significantly reduce decoder complexity for some implementations.

V-PCC real-time decoding and rendering in AR environment was already demonstrated on mid-range mobile phones in trade fairs as IBC2019, see [25] and [26]. Futurewei is releasing an open source version of a V-PCC decoder, OpenV3C (see [27]). In parallel, MPEG has worked on a convergence between the V-PCC and Metadata for Immersive Video (MIV, see MPEG-I part 12) standards, to provide a partly common specification, named "V3C" for "Visual Volumetric Video-based Coding", to facilitate understanding and better promote the adoption of these 3D immersive standards by industry. Even now, V-PCC performance allows contemplating deployment on the mass market for applications as diverse as AR telepresence, VR for medical applications, free viewpoint video, edge-cloud 3D video gaming, edutainment, and many more immersive, innovative, creative services to be unleashed.

References

[1] Keming Cao, Yi Xu, Yao Lu, and Ziyu Wen, "Owlii Dynamic human mesh sequence dataset" ISO/IEC JTC1/SC29/WG11 m42816, 122th MPEG Meeting, San Diego, April 2018

[2] ISO/IEC JTC1/SC29 WG11, Genova, Switzerland, June 2016, w16331, Use cases for Point Cloud Compression

[3] ISO/IEC JTC1/SC29 WG11, Gwangju, Korea, January 2018, w17353, PCC Requirements

[4] ISO/IEC JTC1/SC29 WG11, Alpbach, Austria, April 2020, m53595, PCC requirements status update

[5] https://hevc.hhi.fraunhofer.de/svn/svn_HEVCSoftware/tags/HM-16.168+SCM-8.7

[6] ISO/IEC JTC1/SC29 WG11, Alpbach, Austria, April 2020, w19332, V-PCC codec description

[7] ISO/IEC JTC1/SC29/WG11, Ljubljana, Slovenia, July 2018, m43494, PCC TMC2 with additional projection plane

[8] ISO/IEC JTC1/SC29 WG11, Alpbach, Austria, April 2020, m53943, Container for V-PCC Editing, http://wg11.sc29.org/doc_end_user/documents/130_Alpbach/comments/comment-566-V-PCC_DIS_d90.zip

[9] ISO/IEC JTC1/SC29 WG11, Ljubljana , Slovenia, July 2018, m43723, [PCC] TMC2 Pixel interleaving in geometry and texture layers

[10] ISO/IEC JTC1/SC29 WG11, Gwangju, Korea, January 2018, m42111, PCC TMC2 Geometry coding improvements

[11] ISO/IEC JTC1/SC29/WG11, Ljubljana, Slovenia, July 2018, m43501, 2018. PCC TMC2 low complexity geometry smoothing

[12] ISO/IEC JTC1/SC29/WG11, Geneva, Switzerland, October 2019, m51003, 2019, [V-PCC] On attribute smoothing

[13] ISO/IEC JTC1/SC29/WG11, Geneva, Switzerland, October 2019, m51501, [V-PCC][specification] Patch border filtering specification

[14] ISO/IEC JTC1/SC29/WG11, Macao, China, October 2018, m44784, [VPCC-TM] New contribution on duplicate point avoidance in TMC2

[15] ISO/IEC JTC1/SC29/WG11, Marrakech, Morocco, January 2017, m45946, [V-PCC] Report on Core Experiment CE 2.18 on lossy additional points patch for improved visual quality

[16] ISO/IEC JTC1/SC29/WG11, Geneva, Switzerland, October 2019, m47895, [V-PCC][New proposal] Generalized Enhanced Occupancy Map for Depth

[17] IEEE International Conference on Image Processing (ICIP), Beijing, 2017, pp. 3460-3464, doi: 10.1109/ICIP.2017.8296925, D. Tian, H. Ochimizu, C. Feng, R. Cohen and A. Vetro, "Geometric distortion metrics for point cloud compression,"

[18] IEEE Visual Communications and Image Processing (VCIP), St. Petersburg, FL, 2017, pp. 1-4, doi: 10.1109/VCIP.2017.8305132, R. Mekuria, S. Laserre and C. Tulvan, "Performance assessment of point cloud compression,".

[19] ISO/IEC JTC1/SC29 WG11, Alpbach, Austria, April 2020, m53592, [VPCC][software] mpeg-pcc-renderer software v5.0

[20] ISO/IEC JTC1/SC29 WG11, Alpbach, Austria, April 2020, w19325, V-PCC Test Model v10

[21] Git reference for test model v10 http://mpegx.int-evry.fr/software/MPEG/PCC/TM/mpeg-pcc-tmc2/commit/1018e5467ea0d18b879272bcba8c060e1cfd97a9

[22] ISO/IEC JTC1/SC29 WG11, Alpbach, Austria, April 2020, w19327, V-PCC performance evaluation and anchor results

[23] ISO/IEC JTC1/SC29 WG11, Alpbach, Austria, April 2020, w19324, Common Test Conditions for PCC

[24] Eugene d'Eon, Bob Harrison, Taos Myers, and Philip A. Chou, "8i Voxelized Full Bodies, version 2 – A Voxelized Point Cloud Dataset," ISO/IEC JTC1/SC29 Joint WG11/WG1 (MPEG/JPEG) input document m40059/M74006, Geneva, January 2017

[25] Céline Guede, Ralf Schaefer, IBC 2019, September 2019, https://www.interdigital.com/videos/v--pcc-the-first-mpeg-codec-for-point-cloud-compression

[26] S. Schwarz, M. Pesonen, IBC 2019, September 2019 Real-time decoding and AR playback of the emerging MPEG video-based point cloud compression standard

[27] ISO/IEC JTC1/SC29 WG11, Alpbach, Austria, April 2020, w19375, OpenV3C - multi platform Open Source implementation of V-PCC

