VOLUMETRIC VIDEO GUIDELINES

Version 1.0 2022-12-20

VR Industry Forum

5177 Brandin Court

Fremont, CA 94538, USA

www.vr-if.org

1 Introduction

Advances in 3D capturing and rendering technologies have unleashed a new wave of innovation in Virtual/Augmented/Mixed Reality (VR/AR/MR) content creation and communication, of which volumetric video is an integral part. Volumetric video, a sequence of visual volumetric frames, if uncompressed, may be represented by a large amount of data, which can be costly in terms of storage and distribution.

The guidelines presented in this document cover all aspects of the distribution ecosystems, including compression, storage and delivery, in order to ensure high quality, comfortable consumer volumetric video experiences. These guidelines are aimed at addressing best practices for volumetric video content production and distribution as well as advocating interoperability and deployment guidelines based on common technical standards for volumetric video content distribution, including promoting the use of common profiles across the industry.

The targeted audience includes content distributors, service providers, broadcasters, mobile operators, consumer electronics manufacturers, professional equipment manufacturers, software developers and technology companies that aim to enable deployment of volumetric video content distribution services.

The scope of the guidelines presented herein includes:

- Production: Technical aspects of the media formats used in the interface between the content provider and the service provider for compelling and usable volumetric video experiences.
- Compression: Media codecs for volumetric video, i.e. encoding of different production formats and related media profiles for video, audio and possibly also other media types such as text, graphics, etc. This includes decoding and rendering of the media based on an abstracted distribution data model.
- Storage: Media formats for volumetric video content (e.g. file/segment encapsulation) for different distribution means, including but not limited to storage, download, adaptive bitrate streaming and broadcasting.
- Delivery: Interfaces and protocols for Live, Linear and VOD delivery over streaming (unicast), and broadcast applications.
- Security: Security considerations for Volumetric video are highlighted, and the need to design with security in mind, in particular on the client side, flagged.

Contents

1 Introduction 2
2 References 6
3 Conventions and Terminology 7
4 Volumetric Production 8
4.1 Volumetric Video Production Workflow 8
4.2 Volumetric capture/production systems 10
4.2.1 Microsoft Mixed Reality Capture (MS MRC), US 11
4.2.2 4DViews, France 12
4.2.3 Fraunhofer HHI, Germany 12
4.2.4 Volucap, Germany 13
4.2.5 Volograms 13
4.2.6 Other Studios 14
5 Media Profiles 15
5.1 Mesh 15
5.2 Point Cloud 15
5.3 Selected Media Profiles 16
5.3.1 Video 16
5.3.2 Audio 32
6 Content Security 33
6.1 Security considerations for Volumetric Streaming 33
7 Vertical 1: Volumetric Streaming (one asset) 35
7.1 Point Cloud based streaming of volumetric video 36
7.2 Mesh-based streaming of volumetric video 36
7.3 Reference Architecture 37
7.3.1 Distribution architecture 37
7.4 Technical enablers 37
7.4.1 Suitable Media Profiles 37
7.5 Guidelines for Services Providers 39

Figures

Figure 1: General production workflow for volumetric video 8
Figure 2: Mesh processing 9
Figure 3: Approximate estimate of decreased amount of data in a volumetric mesh workflow 10
Figure 4: Volumetric Capture System overviews (exemplary) 11
Figure 5: Metastage Studio Capture 11
Figure 6: Metastage high level workflow overview 12
Figure 7: Fraunhofer HHI system concept (left), capture studio (right) 13
Figure 8: Volucap capture studio 13
Figure 9: Vologram capture studio 14
Figure 10: 3D Volumetric Studio 3iosk 14
Figure 11: Example of an immersive AR application for point cloud content 15
Figure 12: V-PCC Technology Concept Encoding (simplified) 18
Figure 13: Patch generation process 18
Figure 14. Point cloud projected onto "bounded-box" planes 19
Figure 15. Refine segmentation procedure 20
Figure16: Example of projected geometry (a), texture (b), and occupancy (c) images 21
Figure 17: V-PCC Decoder 21
Figure 18. Point cloud reconstruction process: a – atlas; b – atlas and occupancy map; c – atlas, occupancy map and geometry, d – atlas, occupancy map, geometry and attribute 22
Figure 19: Decoder Conformance Points and Profile Structure 23
Figure 20: Content snapshots 29
Figure 21 Sequence Football 30
Figure 22 Sequence Levi 30
Figure 23 Sequence Mitch 31
Figure 24 Sequence Thomas 31
Figure 25: Distribution Architecture 37
Figure 26: DASH Structure using pre-selections 39

Tables

Table 1: Overview of the proposed media profile 17
Table 2: General video bitstream level limits 25
Table 3: General atlas and tile level limits 26
Table 4: General VC3 related level limits 27
Table 5: Performance Analysis (Huawei experiments) 28
Table 6: V-PCC Basic Rec0 results (Nokia experiments) 28
Table 7: Viewing conditions depending the type of device used 35

2 References

[1] Fraunhofer HHI, "Volumetric video capture and production," *Tdoc S4-190368, 3GPP TSG-SA4 103rd meeting, Newport Beach, CA, USA, April 2019.*

[2] "ISO/IEC 23090-5:2021 Information technology — Coded representation of immersive media — Part 5: Visual volumetric video-based coding (V3C) and video-based point cloud compression (V-PCC)".

[3] "ISO/IEC 23008-2:2020 Information technology — High efficiency coding and media delivery in heterogeneous environments — Part 2: High efficiency video coding".

[4] "ISO/IEC 14496-10:2014 Information technology — Coding of audio-visual objects — Part 10: Advanced Video Coding".

[5] "ISO/IEC JTC 1/SC 29/WG 7, V-PCC Subjective Verification Testing, MDS20992, October 2021".

[6] "ISO/IEC JTC 1/SC 29/AG 5 Guidelines for Verification Testing of Visual Media Specifications (draft 2), Doc. N00030, Online, July 2021".

[7] "ISO/IEC JTC 1/SC 29/WG 11, Open Source Software Framework for Implementation and Evaluation of Point Cloud Compression, Doc. m36527, Geneva, June 2016".

[8] VRIF, "VRIF Guidelines 2.3, https://www.vr-if.org/guidelines/".

[9] "ISO/IEC 23090-10 Information technology — Coded representation of immersive media — Part 10: Carriage of visual volumetric video-based coding data".

[10] Fraunhofer HHI, "Volumetric video player demo," *Tdoc S4-190646, 3GPP TSG-SA4 104th meeting, Cork, Ireland, July 2019..*

[11] "ISO/IEC 23009-1:2019, Information technology — Dynamic adaptive streaming over HTTP (DASH) — Part 1: Media presentation description and segment formats".

[12] DASH-IF, "DASH-IF, Interoperability Points v4.1".

[13] "ISO/IEC 14496-12 Information technology – Coding of audio-visual objects – Part 12: ISO Base Media File Format".

3 Conventions and Terminology

Additional terms are defined in the VRIF Lexicon, available at http://www.vr-if.org/lexicon

AR (Augmented Reality)	augmenting the perception of the real environment with virtual elements by mixing in real-time spatially-registered digital content with the real world
Atlas	Collection of 2D bounding boxes and their associated information placed onto a rectangular frame and corresponding to a volume in 3D space on which volumetric data is rendered
AV (Augmented Virtuality)	consists in augmenting the perception of a virtual environment with real elements. These elements of the real world are generally captured in real-time and injected into the virtual environment. The capture of the user's body that is injected into the virtual environment is a well-known example of AV aimed at improving the feeling of embodiment
DASH	Dynamic Adaptive Streaming over HTTP
EOM	Enhanced Occupancy Mode. A patch coding mode where a patch is associated with enhanced occupancy information
ISOBMFF	ISO/IEC base media file format
Mesh	is a collection of vertices, edges and faces that defines the shape of a polyhedral object
MPEG	Moving Picture Experts Group
MR (Mixed Reality)	It includes both AR and AV. It blends real and virtual worlds to create complex environments, where physical and digital elements can interact in real-time. It is defined as a continuum between the real and the virtual environments but excludes both of them.
PLR	Point Local Reconstruction. A reconstruction mode where additional 3D points are generated in the 3D reconstruction.
Point Cloud	A set of data points in space, which represent a 3D shape, object or scene. Each point is described with Cartesian coordinates X,Y,Z
RGB-D	Denotes a data stream consisting of RGB and depth (D) information.
V3C	Visual Volumetric Video-based Coding
Volumetric Asset	A 3D video representation of a particular object.
Volumetric video	Denotes a dynamic 3-dimensional object with dynamic video texture usually created from multiple cameras capturing the dynamic object
V-PCC	Video – Point Cloud Compression
VR (Virtual Reality)	with the use headsets the user is fully immersed in a computer-simulated reality. These headsets generate realistic images and sounds, engaging two senses to create an interactive virtual world
XR (eXtended Reality)	an umbrella term for AR, AV, VR and MR technologies

4 Volumetric Production

4.1 Volumetric Video Production Workflow

In Figure 1, the general volumetric video production workflow is presented. This workflow should hold for the most relevant companies and institutions that have volumetric video studios and perform productions. As it can be seen from the diagram, studios that provide volumetric video as meshes perform a different path in the workflow compared to studios that use point clouds as output format.

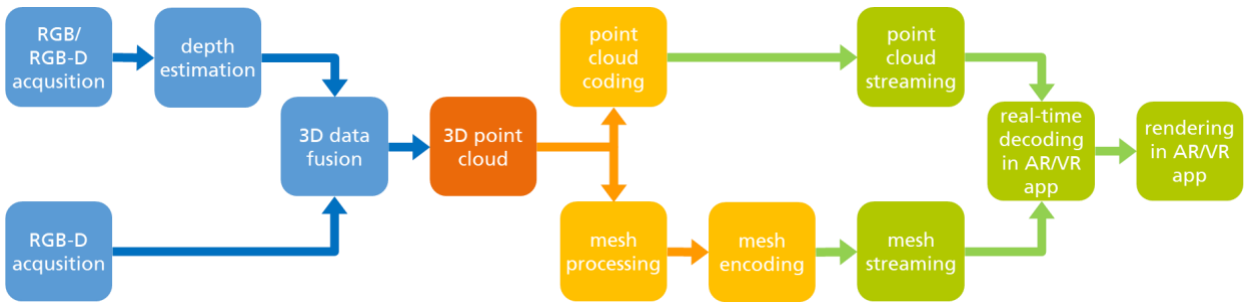


Figure 1: General production workflow for volumetric video

The individual boxes are described as follows:

Acquisition: A number of sensors capture the dynamic scene. This can be either pure video sensors, pure 3D sensors such as time-of-flight sensors or hybrid capture setups. This is reflected in the first two blue boxes.

Depth estimation: If video cameras or hybrid RGB-D sensors are used, then depth estimation is performed on the multiple video streams. The output of this block are basically depth maps per sensor or stereo video pair.

3D data fusion: The depth information from different perspectives is fused into one single point cloud. This holds for depth maps resulting from depth estimation or depth maps form RGB-D acquisition.

3D point cloud: This processing step performs some cleaning and post-processing of the point cloud and is used in both volumetric video formats, either meshes or point clouds.

Point cloud based path

Point cloud coding: The 3D point cloud then encoded in a single stream.

Point cloud streaming: The encoded point cloud stream can then be streamed from anywhere into the target AR/VR application.

Mesh-based path

Mesh processing: In the case of mesh-based rendering, the point cloud is converted into a single mesh. This processing step is a sophisticated and complex step as it involves also mesh cleaning and mesh reduction. The aim of mesh reduction is to decrease the number of vertices. Depending on the target device, either a desktop based

rendering engine or a mobile device the final mesh complexity can range from 60k-100k vertices down to 10k-20k vertices for low capacity devices. The result of this step so far is a sequence of 3D formats (obj-files, ply, etc.) including the related texture provided as a sequence of texture formats (png-files, jpeg, etc.). It is possible to integrate this sequence of meshes in current standard real-time-render engines by creating special asset bundles that contain the volumetric asset.

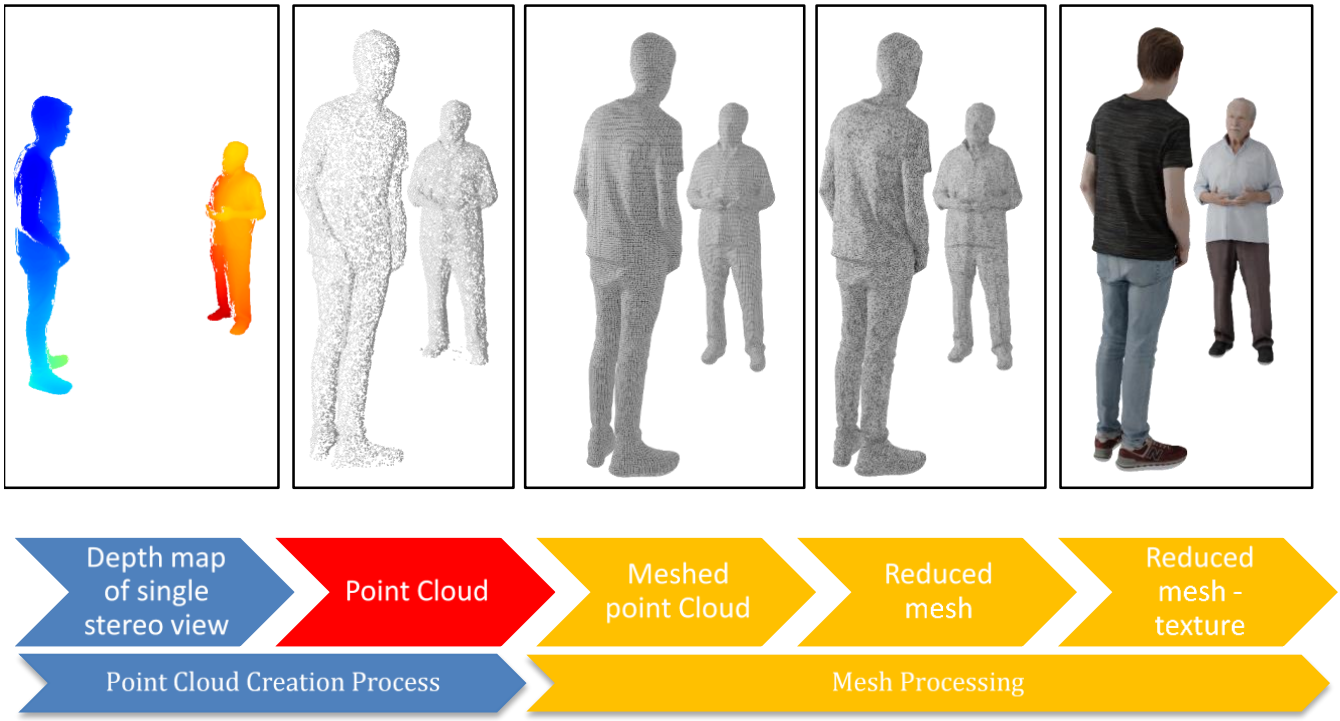


Figure 2: Mesh processing

The result of the 3D video processing workflow so far are independent meshes per frame that consist of individual topology and texture atlas. To create sequences of meshes with identical topology and to improve temporal stability of the related texture atlas, a mesh registration is performed. After definition or automatic selection of a key mesh, succeeding meshes are computed by reshaping the key frame to the geometry of neighbouring frames while preserving topology and local structures. Bidirectional processing is performed to better deal with sudden topology changes of the mesh sequence. Once the deviations of the registered mesh reach a defined threshold, a new key mesh is set. This mesh registration has two advantages. Only for key frames, a new mesh has to be created, while for the registered meshes, only 3D vertex positions have to be adapted. Secondly, the texture atlas arrangement remains the same due to the same topology of the mesh. This has significant advantages for the encoding of the texture atlas and additionally allows easier grading of the texture for the registered period of meshes.

Mesh encoding: The registered mesh sequence is then encoded in a single MP4-file. Until today, several major studios and production companies or research institutes offer their own approach such as Microsoft, 4DViews, Volograms and Fraunhofer HHI. At the moment of writing this document, MPEG is currently working on defining a mesh encoding standard. A very common approach for encoding the meshes is using the Google Draco encoding library[1]. Draco is a library for compressing and decompressing 3D geometric meshes and point clouds. It is intended to improve the storage and transmission of 3D graphics.

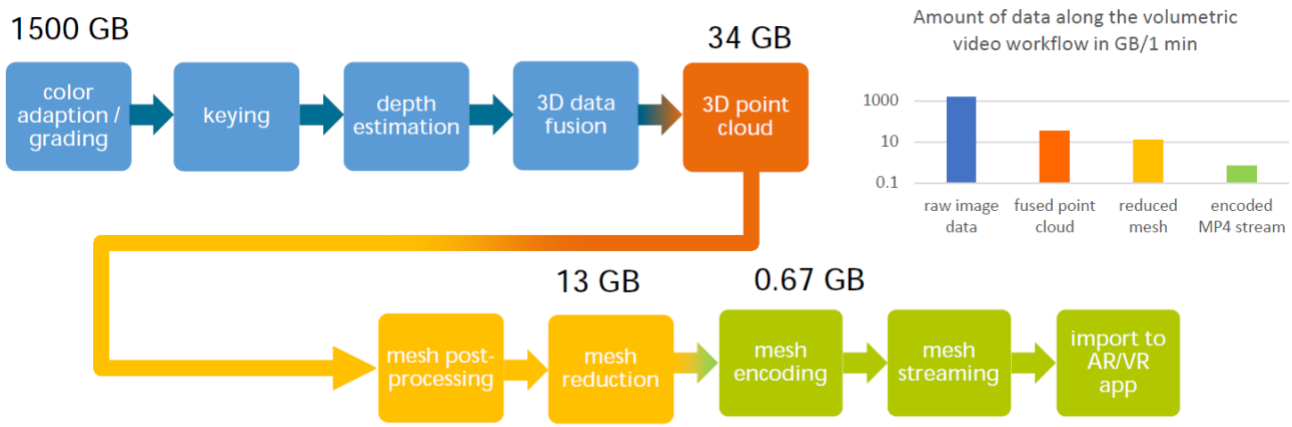


Figure 3: Approximate estimate of decreased amount of data in a volumetric mesh workflow

Mesh streaming: The encoded MP4-file or other proprietary file format can then be streamed from anywhere into the target AR/VR application.

Real-time decoding in AR/VR app: A dedicated plug-in of the AR/VR app performs real-time decoding of either point clouds or meshes.

Rendering in AR/VR app: A dedicated plug-in of the AR/VR app performs the rendering of the volumetric asset (either point clouds or meshes) in the 3D scene. If stereoscopic or multiview, e.g. holographic, display is desired, the client can easily render additional camera views from the volumetric asset.

4.2 Volumetric capture/production systems

There are several studios that today offer services for volumetric capture. This section presents a list of the most common capture setup around the world:

- Microsoft Mixed Reality Capture Studio (MS MRC)
- 4DViews – Holosys, France
- Fraunhofer HHI, Germany
- Volucap, Germany
- Volograms, Ireland

[1] https://github.com/google/draco

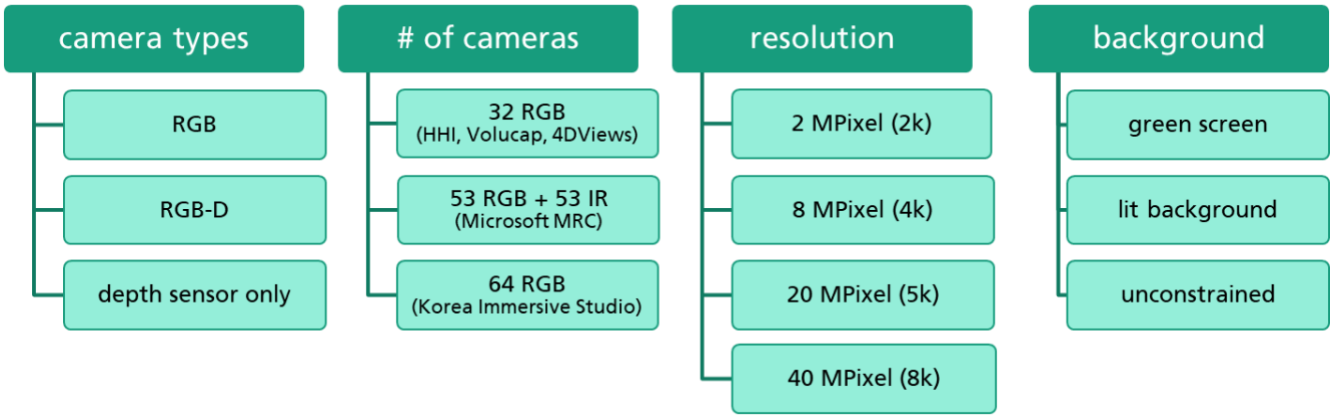


Figure 4: Volumetric Capture System overviews (exemplary)

4.2.1 Microsoft Mixed Reality Capture (MS MRC), US

- Mixed reality capture studios in San Francisco, London (operated by Dimension Studios), Los Angeles (operated by Metastage) and in South Korea (Seoul) (operated by Jump Studio)
- Details of processing technology can be found in [1].
- 106 cameras (12 Mpixel): 53 RGB, 53 infrared
- RGB cameras record the colour required for texture maps & stereo matches.
- IR cameras record the depth and position in the space.
- Default 30 fps; can do up to 60 fps.

Figure 5: Metastage Studio Capture

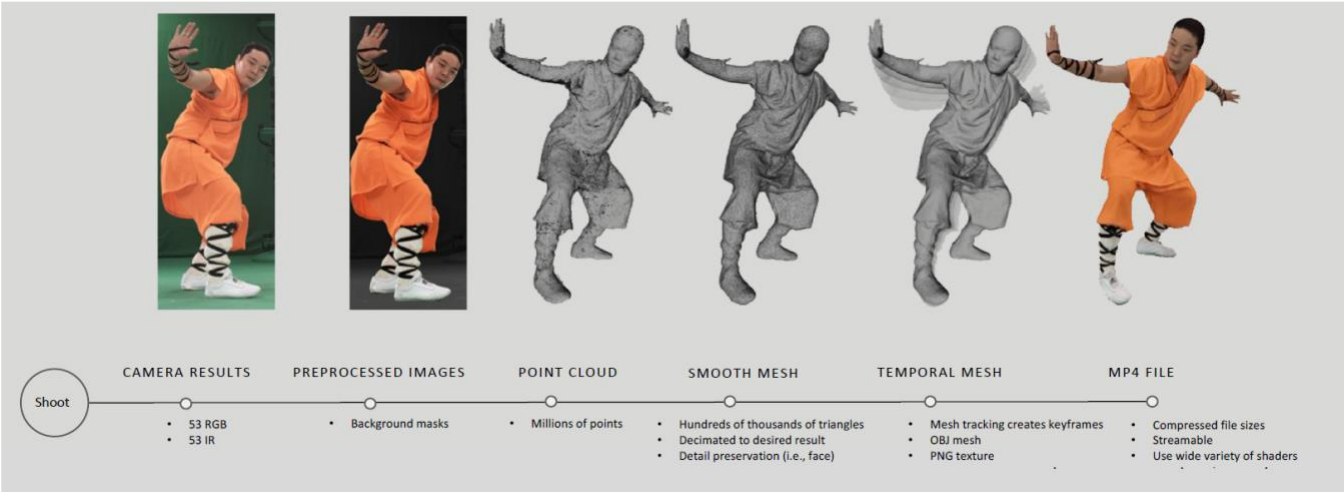


Figure 6: Metastage high level workflow overview

4.2.2 4DViews, France

4DViews Studios is located in Grenoble France and their Volumetric capture system is Holosys. Other studios who use Holosys are: 4D Fun (LA, USA), 4DR Studios (The Netherlands), Crescent Inc. (Japan), Georgia State University (Atlanta, USA), HTC and IP Lab TAICCA (Taiwan)

Holosys spec

Capture volume:	up-to D5m x H2.4m
Capture frame rate:	up-to 60 FPS
Output format:	4DRAW(1), .4DS & .ABC
Texture resolution:	up-to 2880p
Volumetric file size:	down-to 5 MB/s or 40 Mbits/s (2)
Recording capacity:	110 minutes (3)
Storage capacity:	up-to 30 hours of volumetric data

(1): 4Draw gives you full control on quality & compression levels.
(2): average file size for 4DS format, Mobile 720p version.
(3): video recording capacity at 30 FPS.

4.2.3 Fraunhofer HHI, Germany

Research prototype system finalized in 2017 at Fraunhofer HHI labs.

- Capture volume: 2m diameter to 2.4m height
- Frame rate up to 30 fps, recording capacity 90 min (@30 fps)
- 32 high-res cameras (20 MPixel) arranged in stereo pairs
- 120 LED panels (KinoFlo) for arbitrary lit background

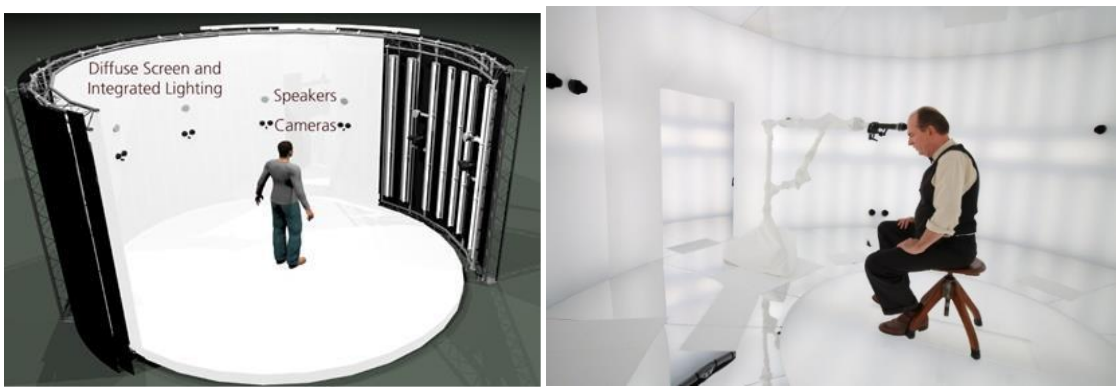


Figure 7: Fraunhofer HHI system concept (left), capture studio (right)

4.2.4 Volucap, Germany

The Volucap system is based on the research prototype by Fraunhofer HHI with further technological improvements.

- Capture volume: 3m diameter to 4m height
- Frame rate up to 30 fps, recording capacity 90 min (@30 fps)
- 32 high-res cameras (20 MPixel) arranged in stereo pairs
- 220 ARRI Sky Panels for arbitrary lit background
- http://www.volucap.de/

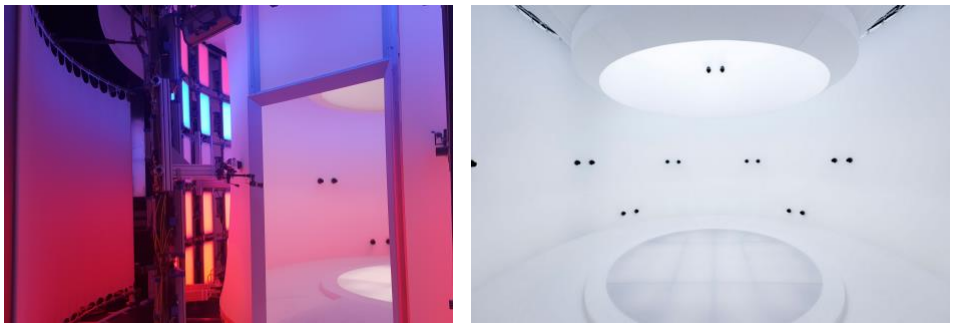


Figure 8: Volucap capture studio

4.2.5 Volograms

The Vologram studio is located in Dublin, Ireland

- Capture Volume: cubic room 2m radius and length cylinder.
- Frame rate up to 30 fps, recording capacity 60 min (@30 fps)

- 12 cameras per aluminium frame (6 HD + 6 4k)
- 10 LED industrial light panels (0.3 x 1.2m)

Figure 9: Vologram capture studio

4.2.6 Other Studios

4.2.6.1 8i

- Synchronized 24 to 60 cameras
- Available in 2k and 4k 30 – 60 fps

4.2.6.2 Mantis Vision

Provides portable 3D Studio 3iosk Volumetric video capture specifications:

	Studio 3iosk	Studio 3iosk XT
No. of Volumetric cameras	15	16
Scanning scene height	1950mm (76.7”)	2050mm (80.7”)
Scanning scene diameter	Ø850mm (33.5”) … (maximum scene diameter 3.5m)	
3iosk footprint	Ø2500mm (98”)	
3iosk height	2477mm (97.5”)	
Color Camera resolution	5MP	
Snapshot scanning time	5 msec	
Snapshot processing time	15 sec	
Video scanning speed	25 fps	

Figure 10: 3D Volumetric Studio 3iosk

5 Media Profiles

5.1 Mesh

As mentioned in section 4.1, studios used their own approach to encode meshes.

At the moment of writing this document, MPEG is currently working on defining a mesh encoding standard. VRIF will include a mesh profile as soon as the standard is ready for publication.

5.2 Point Cloud

While so far, the most common way of representing the visual component of the world has been to take the output of a camera, compress it for transmission and storage using video coding standards and eventually decode it and present it on 2D displays, there are now more and more devices that capture and present 3D representations of the world.

Advances in 3D capture and reconstruction enable real-time generation of highly realistic 3D representations for 3D tele-presence. 3D point clouds are an efficient representation as they can be seamlessly integrated and rendered in 3D virtual worlds enabling a convergence between real and virtual realities. As point cloud capture and reconstruction from single or multiple calibrated cameras are simpler compared to 3D mesh reconstruction, it makes the representation particularly suitable for real-time applications. Some examples use cases are listed below.

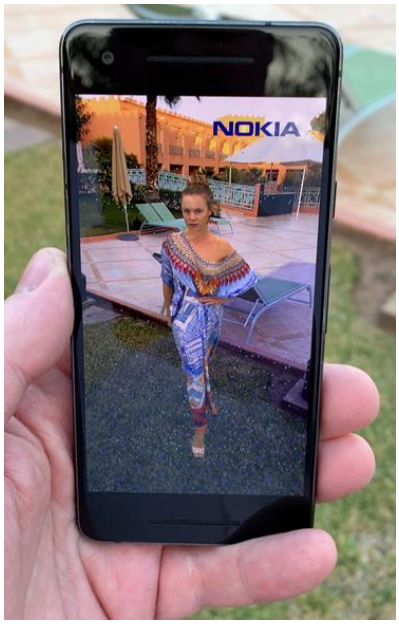


Figure 11: Example of an immersive AR application for point cloud content

Consumption of immersive media content in AR scenarios

Advances in 3D capture and reconstruction enable real-time generation of highly realistic 3D representations for 3D tele-presence. 3D Point clouds are an efficient representation as they can be seamlessly integrated and rendered in AR real-world applications, as demonstrated in Figure 11, enabling a convergence between real and virtual realities.

Real-Time 3D immersive telepresence

As point cloud capture and reconstruction from single or multiple calibrated cameras is simpler compared to 3D mesh reconstruction, it makes the representation particularly suitable for real-time applications. In this case, point clouds are reconstructed, compressed, transmitted, and rendered in real-time as in video conferencing systems, enabling conversational style communication.

Content VR/AR viewing with Interactive Parallax

New VR/AR generation now provides to end-users an immersive experience for viewing specific contents like large field of view movies, e.g., equipped with Head Mounted Display (HMD) or on handheld 2D displays. One major improvement for improving the visual comfort relates to the viewing of content with interactive parallax, where the rendering viewport is updated for each new position of the end-user viewing position (6DOF). This use case is very much related to the previous one, except that this does not require a real time preparation of the source but is rather typically based on a long and sophisticated production workflow mixing shooting and computer graphics.

3D Free viewpoint Sport Replays Broadcasting

Point cloud capture of sports events like basketball, baseball for free viewpoint playback and interaction on mobile devices and TV. This requires a compression and file format standard such that the point cloud data can be streamed or stored in interoperable manner. Desired features are good compression efficiency, progressive download for different device capabilities. The industry has already moved in this direction, examples are Replay Technology. A full scene can also contain textures and/or video. Composition and synchronization can be achieved using MPEG Systems standards.

5.3 Selected Media Profiles

5.3.1 Video

5.3.1.1 Overview

This section describes the selected media profile for video, namely:

1. HEVC Main10 V-PCC Basic Unconstrained media profile

The selected HEVC Main10 V-PCC Basic Unconstrained media profile can encode Point Clouds up to 1 million points at 60 frames per second at level 1.5. Higher levels allow for up to 64M points at 60 frames per second (see Table 4).

5.3.1.2 HEVC Main10 V-PCC Basic Unconstrained media profile

ISO/IEC 23090-5 [2] specifies a generic mechanism for visual volumetric video coding, i.e., visual volumetric video-based coding (V3C). The generic mechanism may be used by applications targeting volumetric content, such as point clouds, immersive video with depth, mesh representations of visual volumetric frames, etc.

In addition to the generic mechanism of coding volumetric content, ISO/IEC 23090-5 [2] specifies one of the applications of visual volumetric video-based coding targeting point cloud representations of visual volumetric frames (V-PCC). In a point cloud sequence, each point cloud frame contains a collection of points. Each point has a 3D position, i.e., geometry information, and each point may also be associated with a number of attributes, such as colour, reflectance, surface normal, etc.

This clause defines a media profile for video-based point cloud coding. Unlike 2-dimensional image or video coding standards, V3C specifies its profiles as a combination of CodecGroup, Toolset, and Reconstruction profile components.

The CodecGroup component indicates video decoding specifications and their profiles, e.g., Main 10 as specified in ISO/IEC 23008-2:2020 [3]:Annex A. The Toolset profile component, which indicates the supported coding tools that could be used to generate the conformant bitstream and shall be supported by a decoder, is defined in subsection H.15.4.1 of the ISO/IEC 23090-5 [2] specification. The Reconstruction profile component, which indicates certain reconstruction operations to be carried out by the decoder, is defined in subsection H.15.5.1 of ISO/IEC 23090-5 [2] specification.

Table 1 provides a high level overview on the supported features in the proposed profile. The detailed specification of each component of the proposed profile is provided in the referred clause.

Media Profile	**Codec**	**CodecGroup**	**Toolset**	**Reconstr.**	**Level**	**Brand**
HEVC Main10 V-PCC Basic Unconstrained	V3C V-PCC ISO/IEC 23090-5 [2]	HEVC Main10	V-PCC Basic	Unconstrained	1.5	v3mt
		ISO/IEC 23090-5 clause A.3	*ISO/IEC 23090-5 clause H.15.4.1*	*ISO/IEC 23090-5 clause H.15.5.1*	*ISO/IEC 23090-5 clause A.6.2*	*ISO/IEC 23090-10 clause A.3*

Table 1: Overview of the proposed media profile

5.3.1.2.1 General (informative)

This media profile fulfils requirements to support widespread dynamic point cloud content delivery for augmented reality applications. The selected *HEVC Main10* CodecGroup profile component ensures maximum distribution, benefiting from a large HEVC deployment base. Compared to other V3C Toolset, and Level requirements, the selected *HEVC Main10 V-PCC Basic Unconstrained* profile provides full support on current mobile devices. The *Unconstrained* reconstruction profile component allows for maximum manufacturer flexibility, balancing decoding performance and reconstruction quality.

5.3.1.2.2 Introduction (tutorial)

The main idea of V-PCC is to reuse existing video codecs for compressing the geometry and texture information of dynamic point clouds. This is done by converting the point clouds into a set of three different video sequences: (1) geometry information, (2) texture information and (3) occupancy in 3D space. These video sequences are then compressed using existing video codecs, such as MPEG-4 AVC, HEVC or similar.

In order to interpret the three video sequences, additional metadata (a relatively small amount of data compared to the video sequences) is generated (i.e. auxiliary patch information) and compressed separately. The three video bitstreams and the metadata are then multiplexed together as to generate the final V-PCC bitstream.

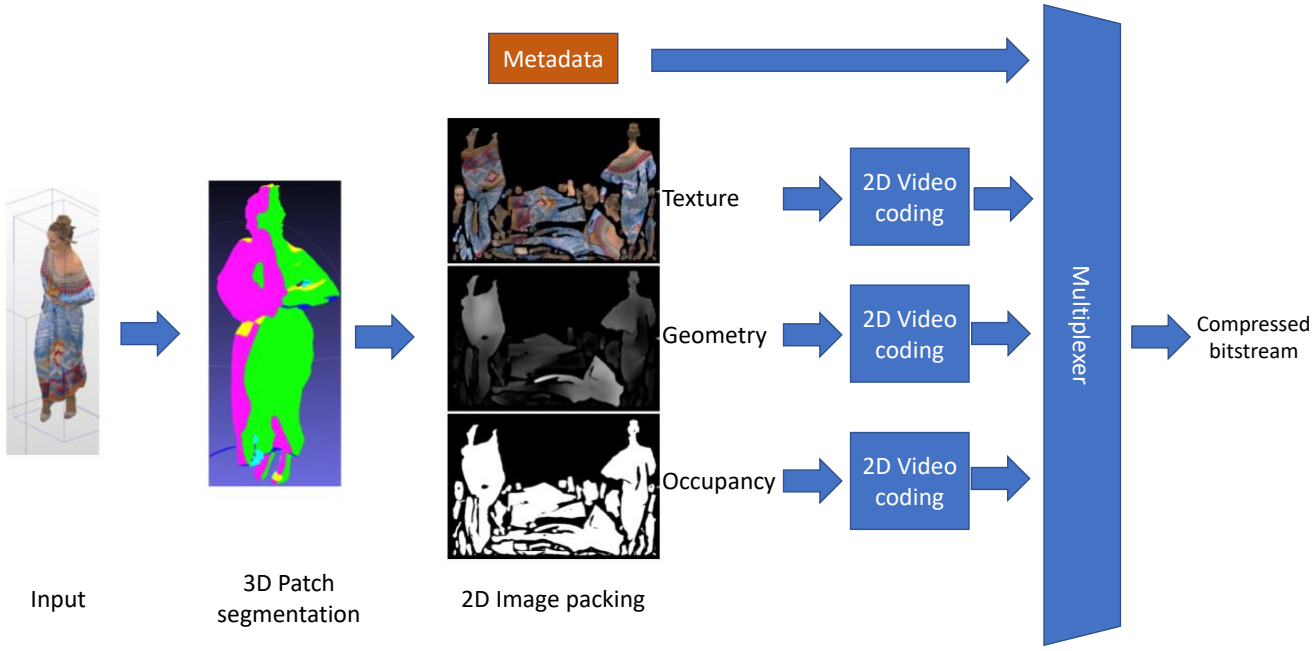


Figure 12: V-PCC Technology Concept Encoding (simplified)

Patch creation[2]

The patch generation process decomposes the point cloud frame by converting 3d samples to 2d samples on a given projection plane using a strategy that provides the best compression. The patch generation process aims at decomposing the point cloud into a minimum number of patches with smooth boundaries, while also minimizing reconstruction error.

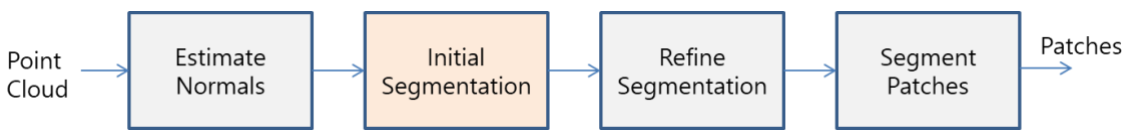


Figure 13: Patch generation process

At the initial stage, a surface normal per each 3D point is estimated. The tangent plane and it's corresponding normal are defined per each point, based on the point's nearest neighbours m within a predefined search distance. A K-D tree is used to separate the data and find neighbours in the vicinity of a point p_i and a barycenter $c = \bar{p}$ of that set of points is used to define the normal. The barycenter c is computed as follows:

$$c = \bar{p} = \frac{1}{m}\sum_{i=1}^{m} p_i$$

The normal is estimated from eigen decomposition for the defined point cloud as:

$$\sum_{i=1}^{m} (p_i - \bar{p})(p_i - \bar{p})^T$$

[2] Reference implementation can be found at https://github.com/nokiatech/vpcc

Based on this information each point is associated with a corresponding plane of a point cloud bounding box. Each plane is defined by a corresponding normal $\vec{\boldsymbol{n}}_{p_{idx}}$ with values:

- (1.0, 0.0, 0.0),
- (0.0, 1.0, 0.0),
- (0.0, 0.0, 1.0),
- (-1.0, 0.0, 0.0),
- (0.0, -1.0, 0.0),
- (0.0, 0.0, -1.0).

More precisely, each point is associated with the plane that has the closest normal (i.e., maximizes the dot product of the point normal $\vec{\boldsymbol{n}}_{p_i}$ and the plane normal $\vec{\boldsymbol{n}}_{p_{idx}}$).

$$\max_{p_{idx}}\left\{\vec{\boldsymbol{n}}_{p_i} \cdot \vec{\boldsymbol{n}}_{p_{idx}}\right\}$$

The sign of the normal is defined depending on the point's position in relation to the "center".

The projection estimation description is demonstrated in Figure 14.

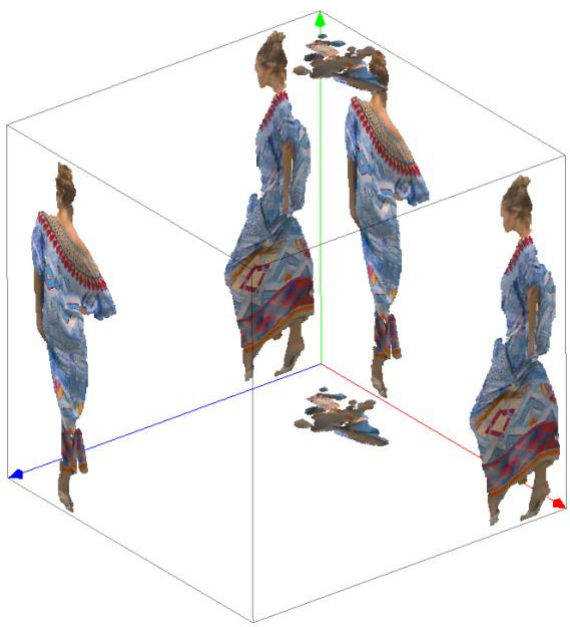


Figure 14. Point cloud projected onto "bounded-box" planes

The initial clustering is then refined by iteratively updating the clustered index associated with each point based on the point's normal and the cluster indices of the point's nearest neighbours.

At the following step the points are clustered based on the closeness of the normals and the distance between points in Euclidian space. Final Patches are created from the clusters by grouping similar clusters. By adding the weight to each plane the patches are refined when the Initial Segmentation process decides the projection plane, in order to increase the size of the patch in the front or back. The weight values are calculated in the first frame per GOF. It is determined according to the ratio of projected points when projecting all points to the three planes (XY,YZ, ZX).

The refine segmentation process provides a minimum number of connected components (patches) for a given number of points in the point cloud frame.

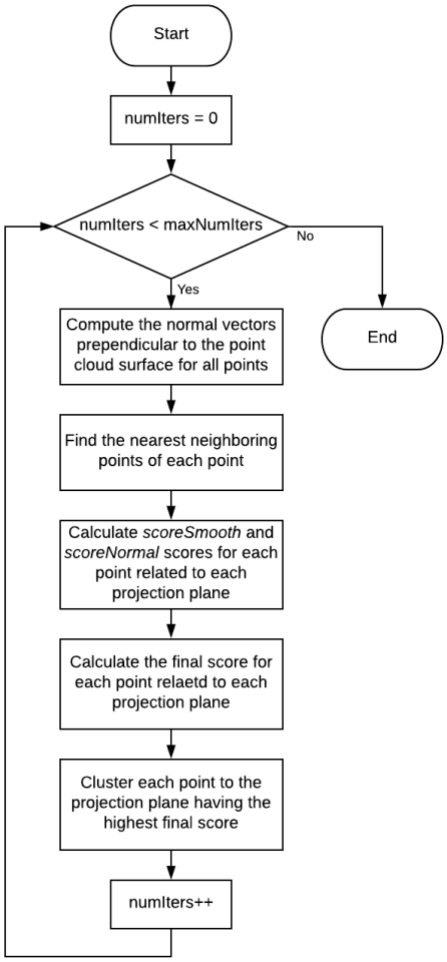


Figure 15. Refine segmentation procedure

The projected patches are then packed using a certain packing strategy. A corresponding set of images for occupancy map, geometry, and attribute(s) is grouped into a video sequences and encoded using a 2-dimensional video coding method. The occupancy map is utilized to signal if 2D pixel in the geometry image shall be reconstructed into 3D space at the decoder.

Additional atlas information carries a description of patches in each point cloud frame, required to reconstruct a 3-dimensional path from a corresponding set of projected patches. Such atlas information is encoded in a lossless binary form.

(a) (b) (c)

Figure16: Example of projected geometry (a), texture (b), and occupancy (c) images

At the decoder, the received V-PCC bitstream is demultiplexed into the separate video bitstreams. Based in the decoded auxiliary patch information, the pixels in each 2D patch are remapped into 3D space based on their occupancy information, i.e., only points with a valid occupancy are mapped into 3D space. After the 3D reconstruction, geometry smoothing may be applied to alleviate potential discontinuities that may arise at the patch boundaries due to compression artefacts in the video coding process.

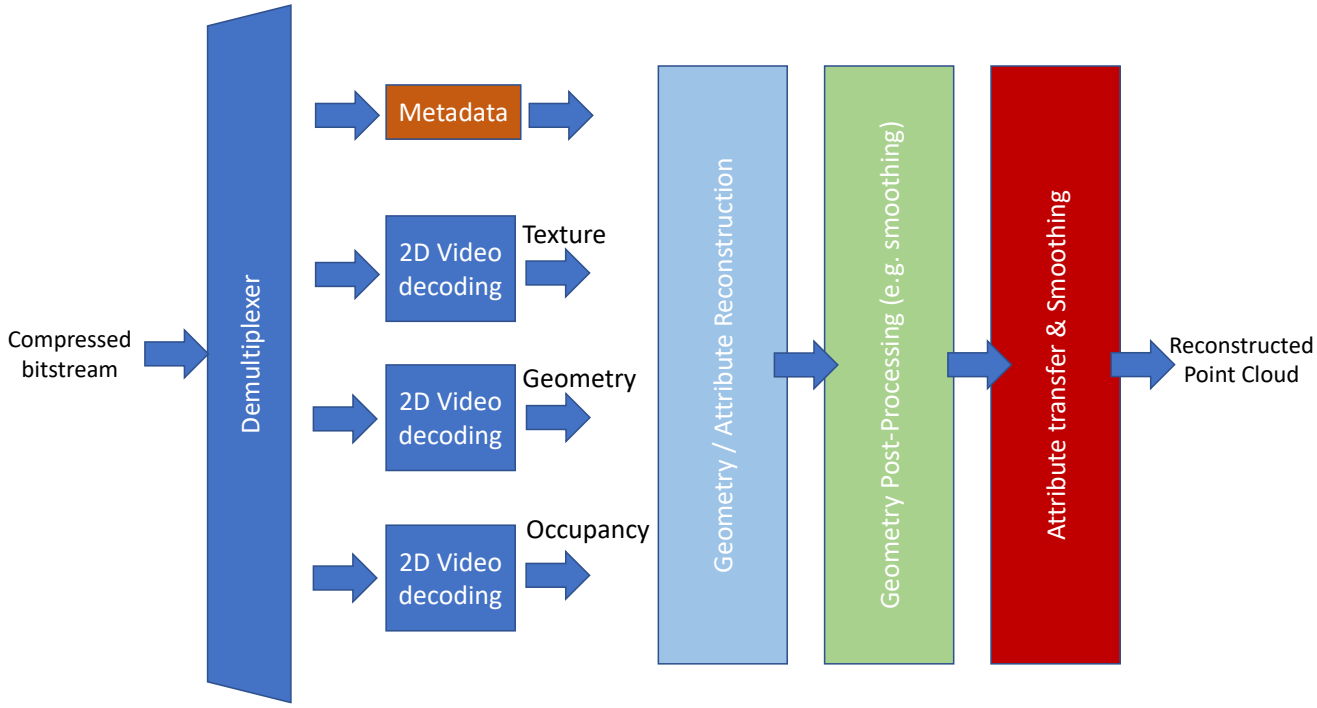


Figure 17: V-PCC Decoder

Point Cloud Reconstruction Process

The reconstruction process for V3C coded point cloud data is illustrated in Figure 18, starting form the atlas information forming planes in 3D space for each patch (a). Then applying occupancy information on these patches to get the correct outlines (b), followed by applying geometry information to form the actual 3D shape (c), and finally mapping the (texture) attributes to the 3D shape to form the final point cloud (d).

a) b) c) d)

Figure 18. Point cloud reconstruction process: a – atlas; b – atlas and occupancy map; c – atlas, occupancy map and geometry, d – atlas, occupancy map, geometry and attribute

5.3.1.2.3 Tiers, Profiles and Levels

V-PCC defines conformance points in terms of profiles and tiers, i.e., combinations of decoding tools and associated bitstream syntax that is expected to be interpreted by a decoder and levels (typically maximum sizes of atlases, pictures, and frame rates, maximum bit rate, buffer capacity, etc.). The decoder conformance points, and profile structure is depicted in Figure 19.

Profile components

V-PCC specifies three profile components:

1. **CodecGroup** profile component (ISO/IEC 23090-5 A.3 [2]), indicating 2D video decoding specifications and their profiles,
2. **Toolset** profile component (ISO/IEC 23090-5 A.4 & H.15.4.1 [2]), indicating the supported coding tools, and
3. **Reconstruction** profile component (ISO/IEC 23090-5 A.5 & H.15.5.1 [2]), indicating certain reconstruction operations
   to be carried out by the decoder.

Conformance Points

This structure mandates two Decoder Conformance Points:

1. One after 2D decoding (Point A) - *Mandatory*
2. One after 3D reconstruction (Point B) - *Optional*

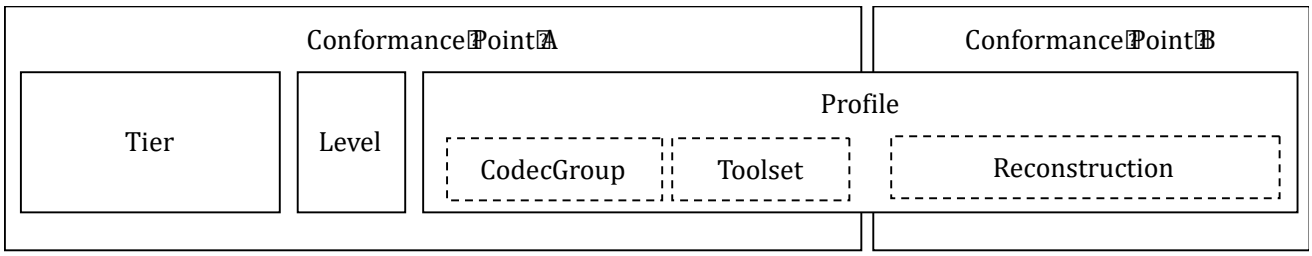


Figure 19: Decoder Conformance Points and Profile Structure

Any V3C bitstream or V3C decoder can claim support of both conformance point A and conformance point B. However, indicating conformance Point A is mandatory while conformance Point B is optional

5.3.1.2.3.1 V3C V-PCC profiles

5.3.1.2.3.1.1 CodecGroup Profile Component

CodecGroup profile component indicates video decoding specifications and their profiles (e.g., Progressive High as specified in ISO/IEC 14496-10:2014 [4] Annex A, Main or Main 10 as specified in ISO/IEC 23008-2:2020 [3] Annex A).

The current supported options are:

- AVC Progressive High
- HEVC Main10
- HEVC444
- VVC Main10
- MP4RA (provided by component codec mapping SEI message)

5.3.1.2.3.1.2 Toolset Profile Component

Toolset Profile Component (ISO/IEC 23090-5 A.4 & H.15.4.1 [2]) indicates V3C specific tools and describes bitstream syntax structure. It currently supports four options:

- V-PCC Basic profile
- V-PCC Basic Still profile
- V-PCC Extended profile
- V-PCC Extended Still profile

Basic tools profiles were designed to meet low complexity decoding and reconstruction and focus on performance and throughput. Extended profiles aim for high-performance platforms with focus on quality and compression efficiency. Key differences are:

- Basic disallows the use of more complex tools, such as enhanced occupancy mode (EOM), Point Local reconstruction (PLR), extended projections, and additional projection directions
- Basic allows only for 3 dimensional attributes
- Basic allows only for individual streams per “map”

5.3.1.2.3.1.3 Reconstruction Profile Component

The Reconstruction profile component indicates certain reconstruction operations to be carried out by the decoder. Currently it supports four options:

- Rec0, ignoring any complex reconstruction operations[3]
- Rec1, defining complex reconstruction for many operations, including geometry smoothing and attribute transfer but excluding occupancy synthesis
- Rec2, defining complex reconstruction for many operations, including occupancy synthesis but excluding geometry smoothing and attribute transfer
- Unconstrained, giving the decoder full freedom but limiting the possibility of conformance testing

5.3.1.2.3.1.4 VC3 Naming convention

VC3 profiles follows the naming convention **CodecGroup + Toolset + Reconstruction**, e.g.:

- AVC V-PCC Basic, is a valid profile that clearly indicates its CodecGroup and Toolset profile component while providing no indication of a Reconstruction profile component.
- AVC V-PCC Basic Rec0, is a valid profile that clearly indicates its CodecGroup and Toolset profile components while also indicating support of a Reconstruction profile component that provides for elementary 3D reconstruction.

5.3.1.2.3.2 Tiers & Levels (A.6)

Currently two tiers are supported, main (MT) and high (HT). The tiers relate directly to the video bitstream level limits requirements and have no impact on any other toolsets or level aspects.

There are 8 specified levels, ranging in support for volumetric images sizes as small as 1 000 000 volumetric samples per frame for low end devices and up to as large as 3 200 000 volumetric samples per frame equivalent at the high end.

[3] Decoders are still allowed to claim conformance at point A and perform complex reconstruction operations (e.g., manufacturer specific tools)

Level	Max luma picture size in (samples)	Max aggregate luma sample rate (samples/sec)	Max # 1000 bits/s per video stream		Max aggregate # 1000 bits/s	
			MT	HT	MT	HT
1.0	2 228 224	133 693 440	12 000	30 000	25 000	100 000
1.5	2 228 224	267 386 880	20 000	50 000	40 000	160 000
2.0	8 912 896	534 773 760	25 000	100 000	60 000	240 000
2.5	8 912 896	1 069 547 520	60 000	240 000	120 000	480 000
3.0	35 651 584	2 139 095 040	60 000	240 000	120 000	480 000
3.5	35 651 584	4 278 190 080	240 000	800 000	480 000	1 600 000
4.0	142 606 336	8 556 380 160	240 000	800 000	480 000	1 600 000
4.5	142 606 336	17 112 760 320	720 000	2 400 000	1 440 000	4 800 000

Table 2 specifies the general video bitstream level limits required to decode a V3C V-PCC bitstream.

Table 3 specifies the atlas and tile related level limits. The specified limits apply to all atlas tiles forming the V3C bitstream.

Table 4 specifies the general V3C and VPS related limits for each level of each tier.

The level 1.5 of the profile proposed in these guidelines (section 5.3.1.2) is highlighted in each table.

Level	Max luma picture size in (samples)	Max aggregate luma sample rate (samples/sec)	Max # 1000 bits/s per video stream		Max aggregate # 1000 bits/s	
			MT[4]	HT[5]	MT	HT
1.0	2 228 224	133 693 440	12 000	30 000	25 000	100 000
1.5	2 228 224	267 386 880	20 000	50 000	40 000	160 000
2.0	8 912 896	534 773 760	25 000	100 000	60 000	240 000
2.5	8 912 896	1 069 547 520	60 000	240 000	120 000	480 000
3.0	35 651 584	2 139 095 040	60 000	240 000	120 000	480 000
3.5	35 651 584	4 278 190 080	240 000	800 000	480 000	1 600 000
4.0	142 606 336	8 556 380 160	240 000	800 000	480 000	1 600 000
4.5	142 606 336	17 112 760 320	720 000	2 400 000	1 440 000	4 800 000

Table 2: General video bitstream level limits

[4] Main Tier

[5] High Tier

Level	Max # Patches per atlas	Max coded atlas frame buffer size (CAB) in 1 000 000 bits	Max atlas bit rate in 1 000 000 bits/s	Max # number of tiles per atlas	Max atlas size	Max patch rate (patches/s)
1.0	2 048	15	15	50	2 228 224	65 536
1.5	4 096	30	30	50	2 228 224	131 072
2.0	16 384	120	120	200	8 912 896	524 288
2.5	32 384	240	240	200	8 912 896	1 036 288
3.0	65 536	480	480	500	35 651 584	2 097 152
3.5	65 536	480	480	500	35 651 584	4 194 304
4.0	262 140	1920	1920	2 000	134 217 728	8 388 608
4.5	262 140	1920	1920	2 000	134 217 728	16 777 216

Table 3: General atlas and tile level limits

Level	Max # of points per second	Max # of points per atlas	Max # of maps	Max # of attributes	Max # of attribute dimensions
1.0	30 000 000	1 000 000	2	1	3
1.5	60 000 000	2 000 000	2	3	3
2.0	120 000 000	4 000 000	4	4	3
2.5	240 000 000	8 000 000	4	8	4
3.0	480 000 000	16 000 000	8	16	5
3.5	960 000 000	32 000 000	8	24	6
4.0	1 920 000 000	64 000 000	16	32	7
4.5	3 840 000 000	128 000 000	16	48	8

Table 4: General VC3 related level limits

5.3.1.2.3.3 Device Performance

In this section, it is presented the performance of the V3C V-PCC profiles on various devices.

Huawei experiments (RA, Rec0)			
Test device	Toolset	FPS	
		AR on	AR off
Samsung Note 10+ (Adreno 640)	Basic	42	26
	Extended	38	24
Huawei P30pro (Mali-G76 MP10)	Basic	42	41
	Extended	36	36
Google Pixel 2XL (Adreno 540)	Basic	30	28
	Extended	24	20
One Plus 7 pro (Adreno 640)	Basic	40	40
	Extended	37	37

Table 5: Performance Analysis (Huawei experiments)

DEVICE	GPU / CHIPSET	FPS	Mpoints@60 fps
iPhone 6S	Apple A9 GPU	15.4	
iPhone 8	Apple A11	**30.3**	
iPhone X	Apple A11	**30.3**	
iPhone XS	Apple A12 GPU	**30.3**	
Samsung Galaxy S10	Mali-G76 / Exynos 9820	**25.0**	
Huawei Mate Pro 20	Mali-G76 / Kirin 980	23.3	1.32
Google Pixel	Adreno 530 / Snapdragon 821	**29.4**	**1.60**
Google Pixel 2 XL	Adreno 540 / Snapdragon 835	**30.3**	**1.81**
Nokia 8 Sirocco	Adreno 540 / Snapdragon 835	**30.3**	
Google Pixel 3	Adreno 630 / Snapdragon 845	**25.6**	
OnePlus (A6013)	Adreno 630 / Snapdragon 845	**25.6**	**1.90**

Table 6: V-PCC Basic Rec0 results (Nokia experiments)

Results from Nokia and Huawei indicate that V-PCC Basic has broad support on *legacy mobile clients*, while V-PCC Extended is at the limit of computational capacity. Anything beyond Rec0 may not run in real-time on a broad range of mobile clients.[6]

5.3.1.3 Quality and performance

A detailed subjective verification test was performed by ISO/IEC JTC1/SC29/WG7 to assess the quality and performance of the V3C V-PCC codec. Detailed logistics and results of the V-PCC subjective verification test are available in [5]. This section will provide a brief overview focusing on the proposed *HEVC Main10 V-PCC Basic Unconstrained media profile*.

The test was focused on well distinguishable V-PCC profiles and target lossy applications that are popular for consumer applications. As test content, over time varying dynamic point clouds have been selected. The test was conducted in line with the Guidelines for Verification Testing of Visual Media Specifications documented in AG5 N00030 [6]. As the test anchor, a known state-of-the-art point cloud codec, the CWI-PCL-Codec, has been selected that fits to the test conditions [6].

Figure 20 illustrates the used test content and the graphs below summarize the subjective evaluation score for the V-PCC verification testing.

(a) Football (b) Levi (c) Mitch (d) Thomas

Figure 20: Content snapshots

[6] It is expected that in a future version of this guidelines, the device performance section will be updated with newer mobile phone models

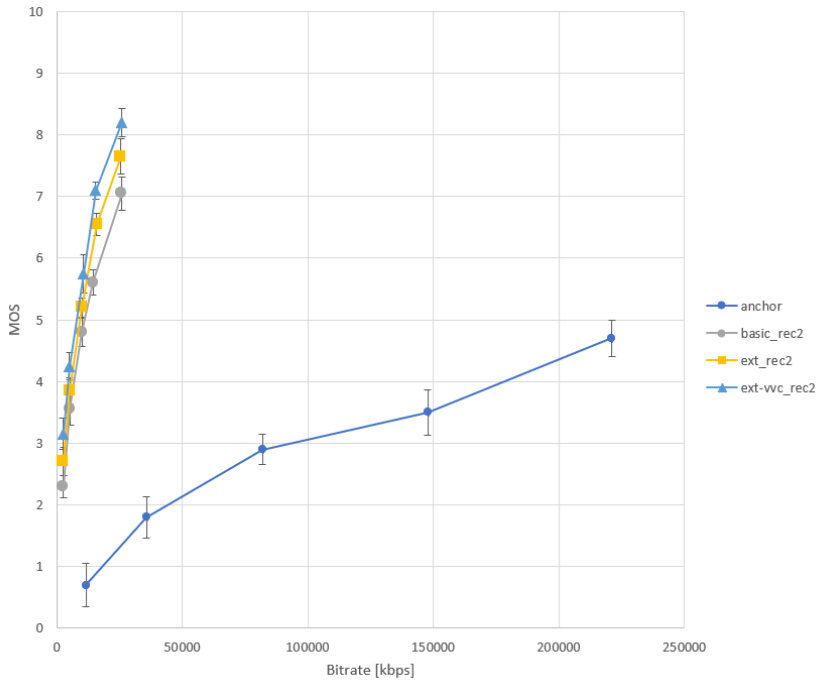


Figure 21 Sequence Football

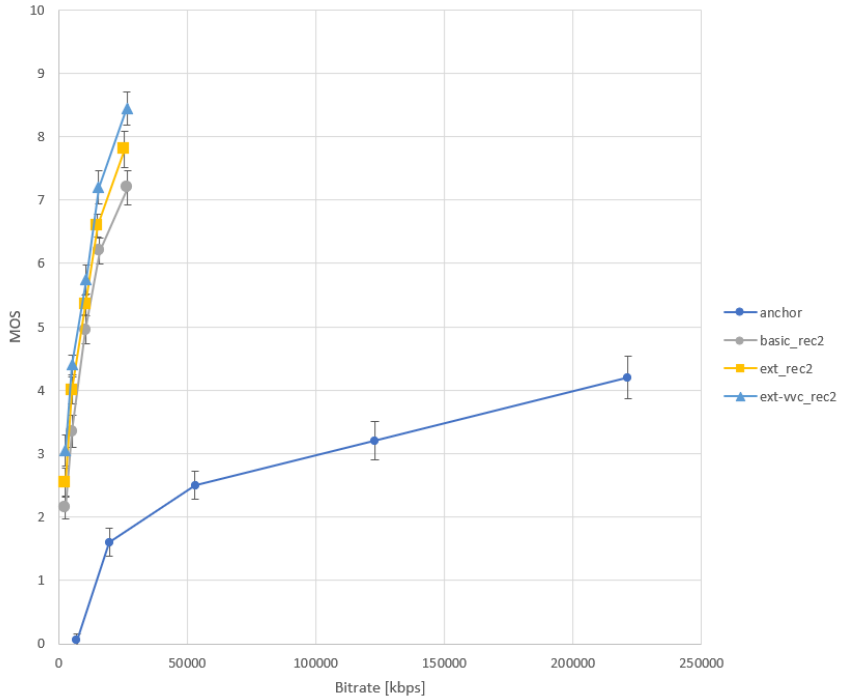


Figure 22 Sequence Levi

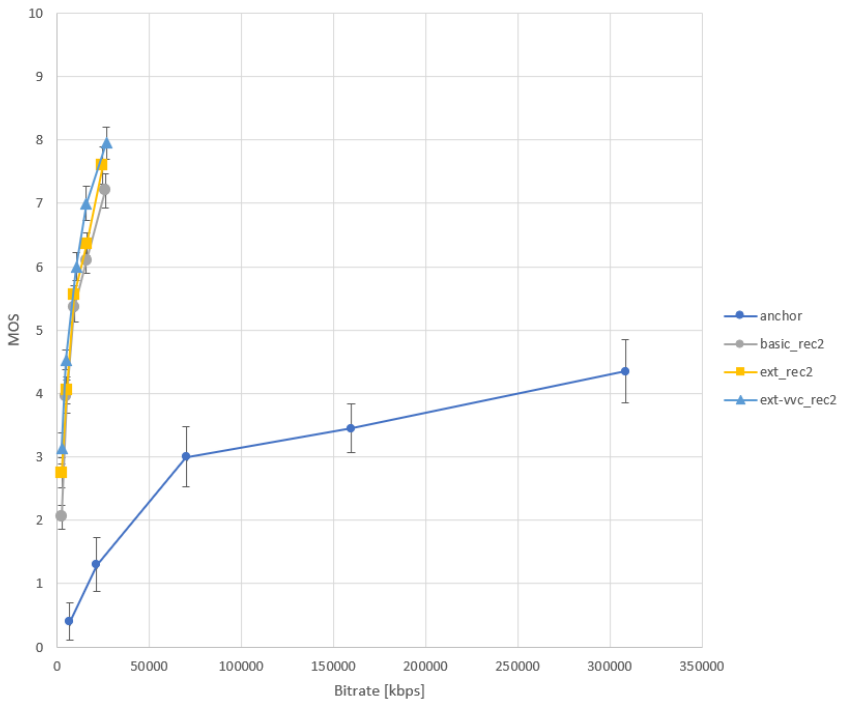


Figure 23 Sequence Mitch

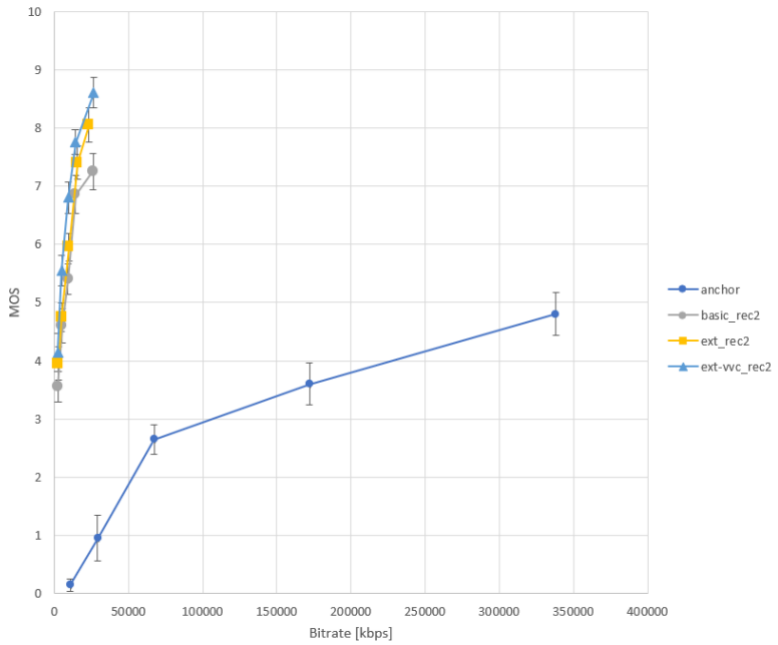


Figure 24 Sequence Thomas

In the presented graphs, the grey line for basic_rec2 corresponds most closely to the proposed *HEVC Main10 V-PCC Basic Unconstrained* media profile. From the subjective verification testing results, it can be concluded that the V3C V-PCC codec outperforms previous state-of-the-art significantly. The performance increase for the extended profile becomes only statistically significant at the highest bit rate test points. The increased complexity introduced by the extended toolset profile component, has not been taken into account in this verification. Thus, the proposed *HEVC Main10 V-PCC Basic Unconstrained* media profile can be safely confirmed as the optimal trade-off between decoding complexity and reconstruction quality.

5.3.2 Audio

No new audio profile has been added to this Guidelines, for previous audio profile please refer to Guidelines 2.3 [7] (about VR360).

6 Content Security

Content owners and rights holders may want to protect their content from being copied and re-distributed, in particular for high value content. For this reason, content may be encrypted on the server side and a secure key sharing mechanism, typically a Digital Right Management, DRM, system, used to share the decryption keys securely with authorized users. DRM systems typically are designed so that neither the operating system nor any applications other than the DRM system and the secure media display system are able to decrypt and read the media. These client restrictions need to be considered when designing end to end volumetric solutions – the client will either have to be able to work with the media in its encrypted form or the DRM and/or secure display systems modified to support a set functionality which allow volumetric content rendering.

Similarly, owners of high value content may want to forensically watermark the content to allow traceability in the case of any unauthorized distribution. Watermarking volumetric content might require different approaches for different streams – this is an area for further study.

6.1 Security considerations for Volumetric Streaming

The term *security*, in Volumetric Streaming applications, covers a number of areas:

Protecting users

- Inappropriate content
- Verifying what you are seeing and who you are communicating with.
- Interaction between physical world and virtual world.
- Proper controls on personal and usage data.

Protecting service providers

- Not exposing users to danger/risk/data sharing.
- Meet content owners requirements.
- Full lifecycle as with traditional media, from prevention, hardening and monitoring through investigating and enforcing

Protecting Content Owners

- Protect against copying delivered content, models, images etc.
- Provide content and service to entitled users only.
- Trace content that is leaked back to the source of the leak.

Focusing on the last of these groups, it is key to design with security in mind form the start to avoid re-work, or even complete re-architecture, especially on the client device side.

To understand this, it is useful to look at the concept of a 'secure video path' in devices today. This is a mechanism that ensures that encrypted video is decrypted and displayed in a secure pipeline on the device, a pipeline that no other apps, or even the device's OS, has access to. This approach is fundamental to protecting high value 'flat' 2D video today.

Looking at an example on an Android device in the following diagram (Figure 25), the secure path is not accessible to any outside applications or system functions. The only access to modify or change the unencrypted media stream is a small set of API's which allow limited transformations on the media, for example to allow the correct user view be displayed in a 360 projection video.

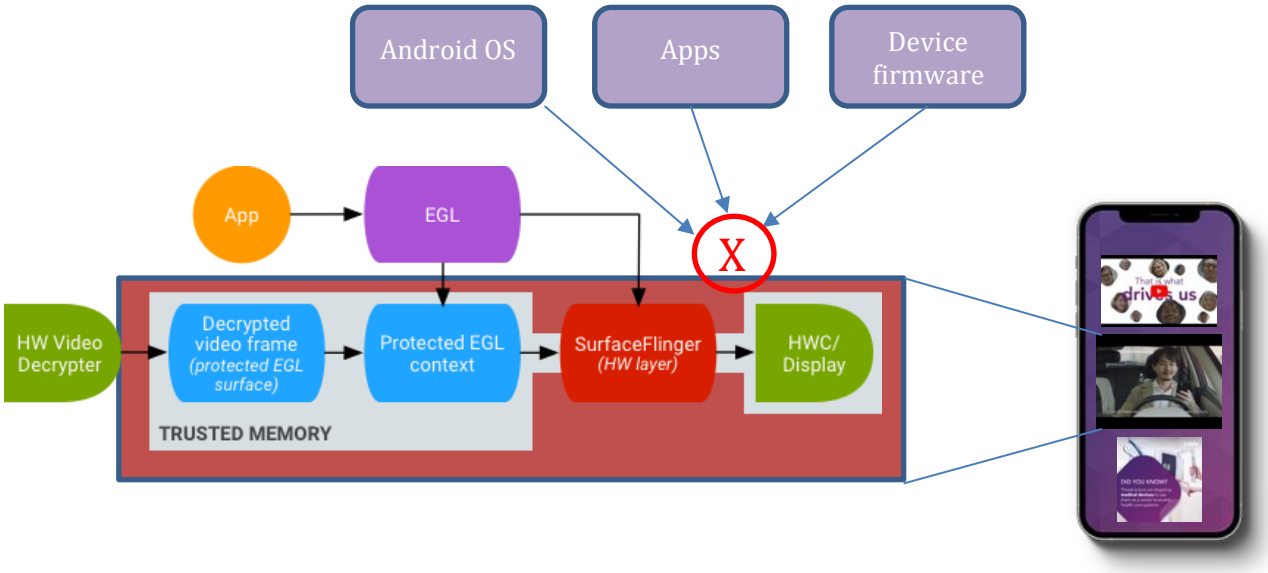


Figure 25: secure video path in Android

The reason this is important for the volumetric world is because volumetric media typically requires new and significant processing on the client device before the media is presented to the user.

For example, as shown in **Error! Reference source not found.**, the client device will receive separate 2D video streams for Texture, Geometry and Occupancy and must process and combine these to produce the point cloud representation to the end user.

If the 2D Video is required to be encrypted and leverage a secure media path then these operations will need to be added to the client devices secure path capabilities.

Considering these security requirements in the initial design and architecting solutions which address them means that the volumetric streaming solution will be designed and ready for high value content.

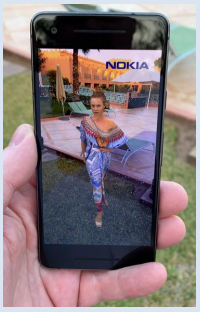
7 Vertical 1: Volumetric Streaming (one asset)

This vertical addressed the use case of streaming one volumetric asset.

The service provider (content aggregator) offers a library of Volumetric Assets. The service provider wants to create a portal to distribute the content to a multitude of devices that can support volumetric video and volumetric processing and rendering. The device may implement functions in hardware for reduced power and battery consumption and optimized processing. Some solutions may be embodied in software (such apps). Typically, volumetric applications make us of well-defined interfaces to hardware functionality, notably decoders.

The Volumetric streaming use case has the potential to render content in a variety of viewing conditions, allowing to the most unskilled user (with minimum knowledge on how to setup the playback, using for instance, flat screens) to consume the content with more immersion experience compared to regular 2D streaming (AR and VR displays), see Table 7.

Table 7: Viewing conditions depending the type of device used

Type of Device	Type of Interaction with the object	Type of Immersion	Available Devices	
Flat Screen /Handheld	- Using touch screen to rotate the view of the object - Use as a window. The object is placed within the real room and the user moves around with the flat screen to visualize different point of views (6DOF)	Immersed in the real world with Augmented reality. Immersion is reduced since the device is handheld.	Mobile phones and tablets.	 ©Nokia
AR Glasses	The object is rendered and overlaid with the user's real view. The user moves around the object to visualize different perspectives (6DOF)	Immersed in the real world with Augmented reality, the object (volumetric asset) is placed in a real environment	Microsoft HoloLens 2 Magic Leap One Vuzix nreal	 © Fraunhofer HHI
VR display	The object is placed in a virtual room. The user can move around with 6DOF	Fully virtual. The object is placed in a virtual environment.	Oculus Quest HTC Vive	 © Arri & Volucap

The service provider provides an application (e.g. browser-based, native app) or makes use of an installed third party application, and may rely on the decoding and rendering capabilities of the device, typically in hardware or by pre-installed or downloaded software decoders.

The service provider wants to avoid testing each device, but rather prefers simple interoperability, e.g. using standardized interfaces.

7.1 Point Cloud based streaming of volumetric video

V3C V-PCC enables compression of point clouds using existing video codecs and metadata, as defined in ISO/IEC 23090-5 [2]. These can be stored in tracks of ISOBMFF as defined in 23090-10 [7] and streamed over DASH as video tracks and metadata tracks. Each video coded component of V3C V-PCC (e.g. geometry, occupancy and attribute) can be streamed as a separate video using existing streaming technologies and infrastructure. However, as with any streaming scenario aiming to deliver multiple parallel video streams, certain considerations need to be made by the client.

Firstly, streaming multiple video tracks over DASH client may result in competition of bitrate between the individual constituent video streams. Client application should therefore consider selecting content alternatives in a manner that does not result in such competition by aiming to target a total bitrate within the bandwidth limits of the system. As with video streaming with DASH, 23090-10 [7] defines mechanisms for grouping alternative representations of V3C components. The client can parse these definitions to select representations of the components that fall within the estimate bandwidth budget.

Secondly, arbitrary selection of alternative representations may not result in optimal quality of the reconstructed experience. Alternative representations of V3C video components should therefore be not arbitrarily combined. For example, high resolution version of a geometry component combined with low resolution version of occupancy component results in rendering artefacts along the object contours. Better quality reconstruction is achieved when alternatives of similar quality are used together. To help client application make decision which representations can be combined, 23090-10 [7] offers playout groups to indicate which representations may be used together.

Thirdly, synchronization of video streams according to presentation timestamps need to be performed by the client upon receiving and decoding individual video streams. Handling multiple parallel decoder instances and synchronizing the output is not always trivial and special considerations may be required, especially on resource constrained platforms and devices. 23090-5 [2] and 23090-10 [7] offer a tool for packing video components into a single video track to alleviate the afore mentioned problem.

7.2 Mesh-based streaming of volumetric video

For any AR or VR application on desktop PC, tablet or smart phone, usually standard render engines such as Unity3D or Unreal are used. Therefore, the streaming itself is more less independent on the end user application. At the beginning of the volumetric video journey, there was no streaming format available at all, because render engines were not aware about rendering of a sequence of meshes. Hence, the sequence of meshes has been integrated in the application by creating a special asset. During start of the application, the assets had to be loaded into the memory, which led to significant waiting time depending on the length and number of volumetric assets. Furthermore, memory limitations of mobile devices constrained the length and number of volumetric assets.

Since beginning of 2018, first streaming formats have been presented that allowed direct streaming of volumetric video into the application. Microsoft presented their own streaming format for volumetric assets and is continuously increasing the list of dedicated plugins for different render engines and XR players including their own player.

4DViews offers their volumetric assets in an own proprietary format, called .4DS, as well as in a standard open format, called .ABC.

In July 2019, Fraunhofer HHI presented a Volumetric Video player demo at the 3GPP TSG WG SA4 #104 [8] meeting in Cork, Ireland. The related document proposes a different approach for encoding mesh sequences. The basic elements of a volumetric video stream are

- obj-mesh sequence
- texture file sequence
- audio file

Each of the elements is encoded separately in a single track and multiplexed into a MP4-file. On the receiving side at the render engine, a dedicated plug-in is available that de-multiplexes the stream and decodes the different tracks. The decoding of volumetric video assets performs in real-time. More details can be found in the related 3GPP document [1].

7.3 Reference Architecture

7.3.1 Distribution architecture

The architecture introduced in this section addresses the service scenario for volumetric streaming, including DASH-based services. The role of Volumetric Content Provider, Volumetric Service Provider and Volumetric Service Platform are differentiated.

Figure 26 considers a functional architecture for such scenario based on the volumetric workflow discussed in section 4.1. The role of Volumetric Content Provider is to capture and generate the volumetric asset in point cloud format. The Volumetric Service Provider is in charge of preparing the content for distribution, that is, encoding (point cloud or mesh) and encapsulation of the content for delivery. Figure 26 highlights the role of Volumetric Service Provider using point cloud encoding and streaming.

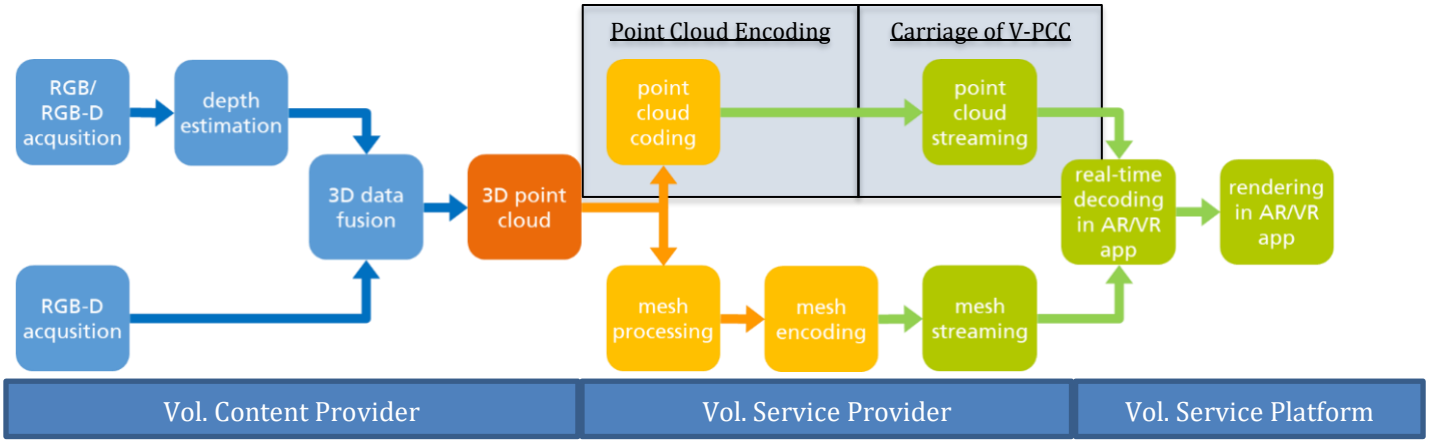


Figure 26: Distribution Architecture

At the receiving end, the Volumetric Service Platform, namely, the delivery client will perform the file format decapsulation, decoding and rendering of the media content.

7.4 Technical enablers

7.4.1 Suitable Media Profiles

7.4.1.1 Video

The HEVC Main10 V-PCC Basic Unconstrained media profile as presented in section 5.3.1.2 may be used to fulfil the use case and provide broad interoperability for streaming point cloud volumetric assets.

7.4.1.2 Audio

The audio profiles defined in VRIF Guidelines 2.3 [7] are suitable for volumetric video.

7.4.1.3 Download

ISO/IEC 23090-10 [9], also known as carriage of V3C data, defines two modes of encapsulation for volumetric content: single-track and multi-track mode. Single-track mode is mostly intended for content creation and editing by studios whereas the latter allows easier integration into existing content delivery infrastructure and acts as the last-mile format for the end user. It is therefore recommended that multi-track encapsulation mode should be used for V3C content. While ISO/IEC 23090-10 defines support for content adaptation and carriage of multiple point clouds in the same file, it is suggested that for downloadable content such features are not used. Each adaptation and different point cloud should be provided in their own file to avoid distributing unnecessary content to the end-user who might be mindful about their data limits.

7.4.1.4 DASH Distribution

DASH integration is specified in ISO/IEC 23090-10 [7], subclause 11.2. The document specifies two designs for enabling the streaming of V3C V-PCC content, one using a single-track mode and one using a multi-track mode.

The single-track mode in enables streaming of V3C ISOBMFF files where V3C content is stored using single-track encapsulation. The single-track mode in DASH should be represented as one Adaptation Set with one or more Representations. While enabling single-track streaming, in many cases multi-track mode may be more preferred method for streaming V3C V-PCC content. Single-track mode is intended for simple distribution and does not handle well bandwidth adaptation. For the purposes of this document, multi-track streaming mode should be the preferred choice.

For multi-track mode each V3C video component shall be represented in the DASH manifest (MPD) file as a separate Adaptation Set. These Adaptation Sets are referred to as Video Component Adaptation Sets. An additional Adaptation Set for atlas information serves as the Main Adaptation Set for the V3C content. If a geometry or attribute component has multiple maps, each map may be signalled using a separate **AdaptationSet** element.

The Main Adaptation Set shall have the @codecs attribute set to 'v3c1', 'v3cg' or 'v3cb' while the @codecs attribute for the Video Component Adaptation Sets, or Representations of these Adaptation Sets, if @codecs is not signalled for the **AdaptationSet** element, is set based on the respective codec used for encoding the component. The value of @codecs shall be set to 'resv.vvvc.hev1'.

The Main Adaptation Set shall contain a single Initialization Segment at the adaptation set level. The Initialization Segment shall contain all parameter sets needed to initialize the V3C decoder, including V3C parameter set as well as other parameter sets for component sub-bitstreams.

Media Segments for the Representation of the Main Adaptation Set shall contain one or more track fragments of the V3C atlas track. Media Segments for the Representations of Video Component Adaptation Sets shall contain one or more track fragments of the corresponding video component track at the file format level. Complete schema for V3C DASH signalling is provided in 23090-10 [7],Annex B.

V3C Dash manifest may contain V3C preselection either by using a Preselection element within the Period element or a Preselection descriptor at the Adaptation Set level. A V3C Preselection element is signaled , as defined in ISO/IEC 23009-1 [9], with an id list for the preselectionComponents attribute including the id of the Main Adaptation Set for the volumetric media followed by the ids of the Video Component Adaptation Sets. The @codecs attribute for the Preselection shall be set to 'v3c1', 'v3cg' or 'v3cb' indicating that the media represented by the Preselection is visual volumetric video-based coding media.

Figure 27 illustrates the DASH structure using pre-selections.

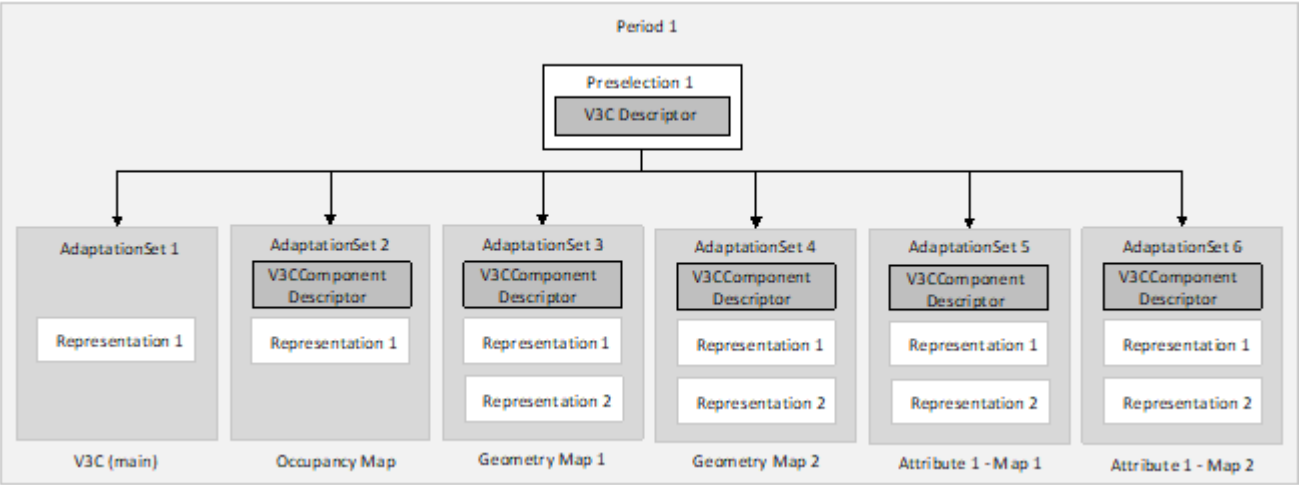


Figure 27: DASH Structure using pre-selections

A SupplementalProperty element with a @schemeIdUri equal to "urn:mpeg:mpegI:v3c:2020:v3c" is referred to as a V3C descriptor. At most one V3C descriptor may be present in Main Adaptation Set, Atlas Adaptation Set or V3C Preselection.

In order to identify the type of Video Component Adaptation Set, a V3CVideoComponent descriptor shall be used. A V3CVideoComponent descriptor is an EssentialProperty descriptor with the @schemeIdUri set to "urn:mpeg:mpegI:v3c:2020:videoComponent". At Adaptation Set level, one V3CVideoComponent descriptor shall be signaled for each V3C video component that is present in the Representations of the Video Component Adaptation Set. The @value of the V3CVideoComponent descriptor shall not be present. The V3CVideoComponent descriptor shall include elements and attributes which describe V3C unit header level information.

Multiple versions of the same volumetric media can be signalled using separate V3C preselections . Preselections that represent alternative versions of the same V3C content shall contain a V3C descriptor with the same @vId value. At most one V3C descriptor shall be present at the preselection level. These Preselections are therefore alternatives to each other and the id of the Main Adaptation Set of the Preselection, first id in the list of Adaptation Set ids for the @preselectionComponents, may be different (where each version of the visual volumetric media has a separate Main Adaptation Set signalled in the MPD file).

7.5 Guidelines for Services Providers

7.5.1.1 HEVC Main10 V-PCC Basic Unconstrained media profile

7.5.1.1.1 V3C CodecGroup constraints

The CodecGroup profile component describes the supported video decoding specifications and their profiles, the proposed HEVC Main10 profile as specified in ISO/IEC 23008-2:2020 [3] Annex A.

To indicate the HEVC Main10 CodecGroup profile component,

`ptl_profile_codec_group_idc` syntax element provided in V3C Parameter Set structure shall be set to 1

All video sub-bitstreams shall conform to the stream format identified by 4CC code equal to 'hev1', as specified in Table A-1 of ISO/IEC 23090-5 [2].

The chroma format for all video sub-bitstreams shall be 4:2:0.

The bit depth of all video sub-bitstreams shall be 8 or 10 bit.

7.5.1.1.2 V3C Toolset constraints

The basic toolset profile component was specified to ensure the low complexity decoding and reconstruction; as well as to focus on performance and throughput, while still achieving good quality.

To indicate the V-PCC Basic Toolset profile component,

`ptl_profile_toolset_idc` shall be set to 0

For a V3C bitstream to comply to the V-PCC Basic profile component,

- each decoded V3C video component output picture, associated with a specific composition time value, `picCompTimeVal`, there shall exist a decoded output atlas frame with a composition time value `atlasCompTimeVal`, where `atlasCompTimeVal` is equal to `picCompTimeVal`.
- All video sub-bitstreams shall have an intra random access point (IRAP) at IRAP points of an atlas sub-bitstream.
- `asps_eom_patch_enabled_flag` shall be set to 0
- `asps_map_count_minus1` shall be set to 0 or 1
- `vps_multiple_map_streams_present_flag` when present, shall be set to 1
- `vps_atlas_count_minus1` shall be set to 0
- `asps_plr_enabled_flag` shall be set to 0
- `ai_attribute_dimension_minus1` shall be set to 2
- `ai_attribute_dimension_partitions_minus1` shall be set to 1
- `asps_use_eight_orientations_flag` shall be set to 0
- `asps_extended_projection_enabled_flag` shall be set to 0
- `vps_occupancy_video_present_flag` shall be set to 1
- `vps_geometry_video_present_flag` shall be set to 1
- `vps_extension_present_flag` shall be set to 0
- `vps_packing_information_present_flag` shall be set to 0
- `vps_miv_extension_present_flag` shall be set to 0
- `asps_miv_extension_present_flag` shall be set to 0
- `casps_extension_present_flag` shall be set to 0
- `casps_miv_extension_present_flag` shall be set to 0
- `caf_extension_present_flag` shall be set to 0
- `caf_miv_extension_present_flag` shall be set to 0
- If `asps_pixel_deinterleaving_enabled_flag` is equal to 1 `asps_map_count_minus1` shall be equal to 0

Furthermore:

- The values of `asps_atlas_sequence_parameter_set_id` and `afps_atlas_sequence_parameter_set_id` shall be in the range of 0 to 15, inclusive.
- The values of `afps_atlas_frame_parameter_set_id` and `ath_atlas_frame_parameter_set_id` shall be in the range of 0 to 63, inclusive.
- The values of `aaps_atlas_adaptation_parameter_set_id` and `ath_atlas_adaptation_parameter_set_id` shall be in the range of 0 to 63, inclusive.
- `TemporalID` shall be equal to 0.

- If `DecOccFullRange`, `DecOccColourPrimaries`, `DecOccTransferCharacteristics`, and `DecOccMatrixCoeffs` are present, their values shall be set to 1, 2, 8, and 2, respectively.
- If `DecGeoFullRange`, `DecGeoColourPrimaries`, `DecGeoTransferCharacteristics`, and `DecGeoMatrixCoeffs` are present, their values shall be set to 1, 2, 8, and 2, respectively.
- If `DecGeoAuxFullRange`, `DecGeoAuxColourPrimaries`, `DecGeoAuxTransferCharacteristics`, and `DecGeoAuxMatrixCoeffs` are present, their values shall be set to 1, 2, 8, and 2, respectively.

7.5.1.1.3 V3C reconstruction constraints

For the Unconstrained reconstruction profile component, reconstruction operations can be ignored. Decoder manufacturers are allowed to implement their own reconstruction operations, thus allowing for maximum manufacturer flexibility.

To indicate the Unconstrained reconstruction profile component,

- `ptl_profile_reconstruction_idc` syntax element provided in V3C Parameter Set structure shall be set to 255

The Unconstrained reconstruction profile component does not apply any constraints on reconstruction operations regarding:

- Chroma format conversion
- Resolution upsampling
- Frame rate conversion
- Pixel deinterleaving
- PLR
- EOM
- Duplicate point removal
- RAW patches
- Geometry Smoothing
- Attribute
- Transfer
- Attribute Smoothing
- Occupancy Synthesis

7.5.1.1.4 V3C level constraints

V3C level 1.5 is equivalent to HEVC Level 4.1. To indicate V3C Level 1.5,

`ptl_level_idc syntax element provided in V3C Parameter Set structure` shall be set to 45

Level constraints as set in ISO/IEC 23090-5 [2], Annex 6.2, shall be followed. In particular:

AspsFrameSize shall be less than or equal to 2 228 224
vps_map_count_minus1 + 1 shall be less than or equal to 2
ai_attribute_count shall be less than or equal to 3
ai_attribute_dimension_minus1 + 1 shall be less than or equal to 3
afti_num_tiles_in_atlas_frame_minus1 + 1 shall be less than 50
When the current atlas frame is an IDR coded atlas, ath_atlas_frm_order_cnt_lsb shall be equal to 0
When the current common atlas frame is an IRAP coded common atlas, caf_common_atlas_frm_order_cnt_lsb shall be equal to 0

7.5.1.1.5 Elementary stream constraints

The V3C NAL sample stream shall comply with HEVC Main10 V-PCC Basic Unconstrained Level 1.5, as specified in sections 5.3.1.2.3.1.

The V3C NAL sample stream format is specified in ISO/IEC 23090-5 [2], Annex D.

For each picture, the following SEI messages shall be present in the bit stream, as specified in ISO/IEC 23090-5 [2], Annex F:

SEI manifest SEI message
SEI prefix indication SEI message
Active substreams SEI message
Component codec mapping SEI message
Buffering period SEI message
Atlas frame timing SEI message

7.5.1.1.6 ISO base media file format constraints

`compatible_brands` in `FileTypeBox` shall include `'v3mt'`. File readers conforming to the 'v3mt' brand shall support multi-track encapsulation of V3C data specified in ISO/IEC 23090-10 [7], subclause 7.4.

A multi-track encapsulated V3C data container confirming to the specified profile shall include the following:

- A V3C atlas track which contains a V3C parameter set and atlas parameter sets in the sample entry and atlas component bitstream NAL units in the samples. A V3C atlas track may also include track references to other tracks carrying the payloads of video compressed V3C units (i.e., V3C unit types equal to V3C_OVD, V3C_GVD, and V3C_AVD) or to V3C atlas tile tracks.
- One V3C video component tracks where the samples contain access units of a video-coded elementary stream for occupancy data (i.e., payloads of V3C units of type equal to V3C_OVD).
- One or two V3C video component tracks where the samples contain access units of video-coded elementary streams for geometry data (i.e., payloads of V3C units of type equal to V3C_GVD).
- One or two V3C video component tracks where the samples contain access units of video-coded elementary streams for attribute data (i.e., payloads of V3C units of type equal to V3C_AVD).

7.5.1.1.7 Receiver requirements

Receivers conforming to the HEVC Main10 V-PCC Basic Unconstrained profile shall be capable of processing either all referenced SEI messages in section 5.3.1.2.3.1.1 or all allowed boxes referenced in section 5.3.1.2.3.1.2

Receivers conforming to the HEVC Main10 V-PCC Basic Unconstrained profile shall be capable of decoding all V3C bitstreams or collection of V3C sub-bitstreams, according to ISO/IEC 23090-5 [2], clause 9, for which all of the following conditions apply:

The V3C bitstream or the collection of V3C sub-bitstreams are indicated to conform to the supported CodecGroup and Toolset profile components, as indicated in sections 5.3.1.2.3.1.2 and 5.3.1.2.3.2
The V3C bitstream or the collection of V3C sub-bitstreams are indicated to conform to a level that is lower than or equal to the specified level, as indicated in section 5.3.1.2.3.2.

Receivers conforming to this media profile shall ignore any reserved values specified in ISO/IEC 23090-5 [2].

Receivers conforming to this media profile shall ignore all NAL units with values of nal_layer_id not equal to 0.

Receivers conforming to this media profile shall ignore all asps_extension_data_flag, aaps_extension_data_flag, casps_extension_data_flag, and caf_extension_data_flag syntax elements.

