

**ISO/IEC JTC 1/SC 29/WG 11**

**Coding of moving pictures and audio**

**Convenorship: UNI (Italy)**

**ISO/IEC JTC 1/SC 29/WG 11 N****18563**

**Document type: Approved WG 11 document**

**Title: Common Test Conditions for Immersive Video**

**Status: Output document**

**Date of document: 2019-08-09**

**Source: Video**

**Expected action:**

**No. of pages: 22**

**Email of convenor: leonardo@chiariglione.org**

**Committee URL: mpeg.chiariglione.org**

**ORGANISATION INTERNATIONALE DE NORMALISATION**

**ISO/IEC JTC1/SC29/WG11**

**CODING OF MOVING PICTURES AND AUDIO**

**ISO/IEC JTC1/SC29/WG11 MPEG/N18563**

**July 2019, Gotenburg, Sweden**

|  |  |
| --- | --- |
| **Source** | Video |
| **Status** | Approved |
| **Title** | Common Test Conditions for Immersive Video |
| **Authors** | Joel Jung, Bart Kroon, Jill Boyce |

# Introduction

Common test conditions are desirable to conduct coding experiments in a well-defined environment and ease the comparison of the outcome of experiments. This document specifies the common test conditions for immersive video activities. The common test conditions are defined to evaluate the coding efficiency, subjective quality, pixel rate and user experience of immersive video solutions. The technical approach is following these steps:

1. Compress test content,
2. Synthesize intermediate views from decoded views and metadata (when available),
3. Render viewports of real/virtual pose traces with a limited or a wider movement,
4. Evaluate coding efficiency and parallax effect, considering both decoded views and synthesized views.

The bitstream shall be viewer independent, meaning that neither the position nor the orientation of the viewer shall be considered when compressing the test content. The range of supported possible viewer positions is constrained and known.

Three different anchors are used: the first one, the “MIV anchor” (Metadata for Immersive Video), is HEVC + TMIV-based, and encodes atlases with the test model. The second one, the “MIV view anchor”, is also HEVC + TMIV based but directly encodes a subset of the source views. The third anchor, the “MV-HEVC anchor”, is based on MV-HEVC and VVS.

# Test material

This section describes the test material that is used in the common test conditions and defines the numbering of the views. References to input documents are included for a more detailed description of each sequence. Subsequent subsections provide download links for sequence data and metadata. CTC-specific configuration files are provided as attachments to this document, as reported in Table 22. The test material is organized in two categories:

* Class CG: computer-generated content, containing CG1, CG2, and CG3 sub-categories,
* Class NC: natural content with estimated depth, containing NC1, NC2 and NC3 sub-categories.

The sequences have a common format as defined in the *Call for MPEG-I Visual Test Materials on 6DoF* [N17462] determining texture and depth representations, filenames and metadata. The views are numbered according to the ordering of the metadata files, counting from zero.

Note that the view numbering exists sometimes in a matrix version for the rig description and an ordinal version for the JSON file. Annex 2 “Source View label conversion” gives the correspondence between the two.

The test material is provided as a set of raw sequences, one per view and component (texture or depth). Texture and depth maps sequences characteristics are reported in the following sub-sections. The CG1 and NC1 sequences are named as follows:

v0\_texture\_4096x2048\_yuv420p10le.yuv

v0\_depth\_4096x2048\_yuv420p16le.yuv

using view 0 of ClassroomVideo as an example. The general format is:

**v**ViewNumber**\_**Component**\_**Width**x**Height**\_**VideoFormat**.yuv**

The attachments, described in Annex 1 [A11], include the json camera parameters to be used for all sequences. Attachment [A13] contains md5sums of the source material.

## Computer-generated content

### ClassroomVideo

The general characteristics of the ClassroomVideo sequence are summarized in Table 1. Source view positions are according to a hexagonally-packed circular disc with an additional top and bottom view, as shown in Figure 1.

Table 1: Summary of the ClassroomVideo sequence

|  |  |
| --- | --- |
| **Category – Name** | CG1 - A |
| **Input contributions** | m42415, m42756 and m42944 |
| **Length & frame rate** | 120 frames (30 fps) |
| **Number of source views** | 15 |
| **Texture format** | YUV 4:2:0 10 bit |
| **Depth format** | YUV 4:2:0 16 bit, normalized disparity in [0.8m, ) range |
| **Source view resolution** | 4096 × 2048 |
| **View FoV & mapping** | 360° × 180° ERP |
| **Global FoV** | 360° × 180° |
| **Download** | http://mpegfs.int-evry.fr/mpegcontent/ws-mpegcontent /MPEG-I/Philips/ClassroomVideo/CE/  Data: v\*.zip |

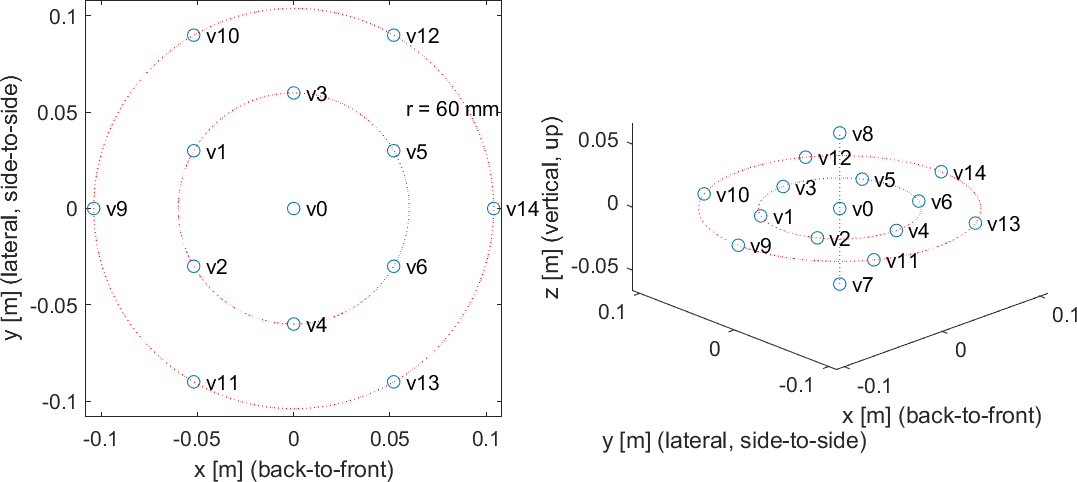


Figure 1: Visualization of the viewpoints of the ClassroomVideo sequence

### TechnicolorMuseum

The general characteristics of the TechnicolorMuseum sequence are summarized in Table 2. The cameras are disposed on a spherical surface of 30 cm radius, and divergent in the direction of the sphere radius. Figure 2 provides the (X, Y, Z) coordinates and the spherical dimension, with an example using the 11th view. The metadata file comprising source and intermediate view positions is attachment A12 to this output document.

Table 2: Summary of the TechnicolorMuseum sequence

|  |  |
| --- | --- |
| **Category - Name** | CG1 - B |
| **Input contribution** | m42349 |
| **Length & frame rate** | 300 frames (30 fps) |
| **Number of source views** | 24 |
| **Source view resolution** | 2048 × 2048 |
| **Texture format** | YUV 4:2:0 10 bit |
| **Depth format** | YUV 4:2:0 16 bit, normalized disparity in [0.5m, 25m] range |
| **View FoV & mapping** | 180° × 180° ERP |
| **Global FoV** | 360° × 180° |
| **Download** | http://mpegfs.int-evry.fr/mpegcontent/ws-mpegcontent/ MPEG-I/Technicolor/TechnicolorMuseum/CE/  Data: v\*.zip |

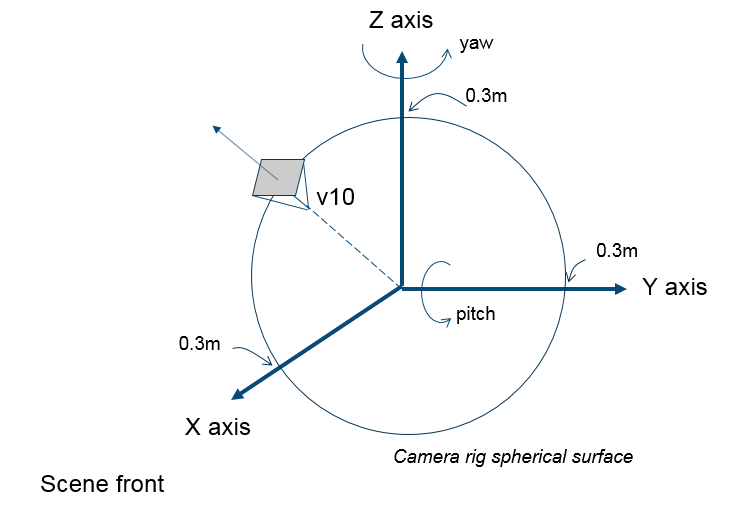


Figure 2: Coordinate system as used by 3D Audio and OMAF, with view 10 of the TechnicolorMuseum sequence superimposed

### TechnicolorHijack

The general characteristics of the TechnicolorHijack sequence are summarized in Table 3. Figure 3 provides a visualization of the virtual camera rig in bias, top and front view respectively. The metadata file comprising source and intermediate view positions is in attachment [A11] of this output document.

Table 3: Summary of the TechnicolorHijack sequence

|  |  |
| --- | --- |
| **Category - Name** | CG1 - C |
| **Input contribution** | m42349 |
| **Length & frame rate** | 300 frames (30 fps) |
| **Number of source views** | 10 |
| **Source view resolution** | 4096 × 4096 |
| **Texture format** | YUV 4:2:0 10 bit |
| **Depth format** | YUV 4:2:0 16 bit, normalized disparity in [0.5m, 25m] range |
| **View FoV & mapping** | 180° × 180° ERP |
| **Global FoV** | 180° × 180° |
| **Download** | http://mpegfs.int-evry.fr/mpegcontent/ws-mpegcontent /MPEG-I/Technicolor/TechnicolorHijack/CE/  Data: v\*.zip |

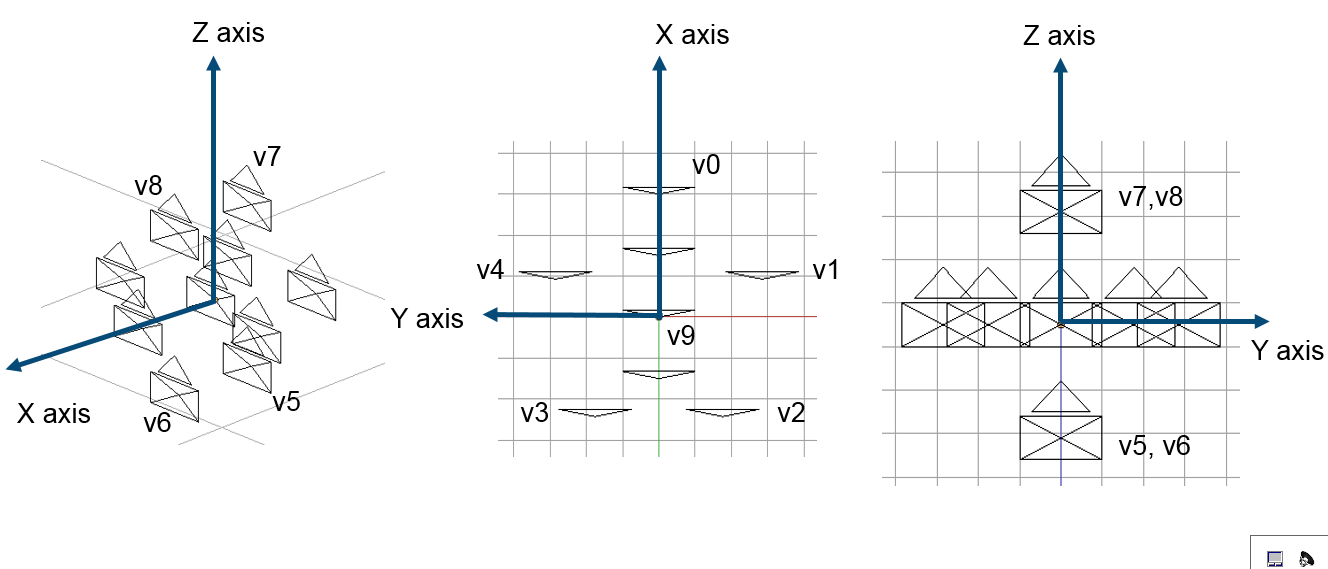


Figure 3: Visualization of the view positions of the TechnicolorHijack sequence

### OrangeShaman

The general characteristics of the OrangeShaman sequence are summarized in Table 4 and source view naming in Table 24. The captured views form a 5×5 planar array and are numbered v0-0 to v4-4 following left to right and top to bottom scan order.

Table 4: Characteristics of the OrangeShaman sequence

|  |  |
| --- | --- |
| **Category - Name** | CG2 - H |
| **Input contribution** | m43318 |
| **Length & frame rate** | 300 frames (30 fps) |
| **Number of source views** | 25 (5x5) |
| **Source view resolution** | 1920x1080 |
| **Texture format** | YUV 4:2:0 8 bit |
| **Depth format** | YUV 4:2:0 8 bit |
| **View FoV & mapping** | 77.3° × 48.5° Rectilinear |
| **Lens** | 20 mm |
| **Camera spacing** | 10cm x 10cm |
| **zNear** | 0.4 |
| **zFar** | 4.2 |
| **Download** | http://mpegfs.int-evry.fr/mpegcontent/ws-mpegcontent/MPEG-I/Orange/OrangeShaman  Texture: OrangeShaman\_x\*\*y\*\*.zip  Depth: OrangeShaman\_d\_x\*\*y\*\*.zip |

### OrangeDancing

The general characteristics of the OrangeDancing sequence are summarized in Table 5. The captured views form a 23×3 planar array and are numbered v0-0 to v22-2 following left to right and top to bottom scan order.

Table 5: Characteristics of the OrangeDancing sequence

|  |  |
| --- | --- |
| **Category - Name** | CG2 - I |
| **Input contribution** | m43318 |
| **Length & frame rate** | 300 frames (30 fps) |
| **Number of source views** | 42 (14 x 3) |
| **Source view resolution** | 1920x1080 |
| **Texture format** | YUV 4:2:0 8 bit |
| **Depth format** | YUV 4:2:0 8 bit |
| **View FoV & mapping** | 90° × 58.7° Rectilinear |
| **Lens** | 16 mm |
| **Camera spacing** | 3.9° along ellipse with rx=5 and ry=4 |
| **zNear** | 1.2 |
| **zFar** | 14.2 |
| **Download** | http://mpegfs.int-evry.fr/mpegcontent/ws-mpegcontent/MPEG-I/Orange/OrangeDancing  Texture: OrangeDancing\_x\*\*y\*\*.zip  Depth: OrangeDancing\_d\_x\*\*y\*\*.zip |

### OrangeKitchen

The general characteristics of the OrangeKitchen sequence are summarized in Table 6 and source view positions in Table 24. The captured views form a 5×5 planar array and are numbered v0-0 to v4-4 following left to right and top to bottom scan order.

Table 6: Characteristics of the OrangeKitchen sequence

|  |  |
| --- | --- |
| **Category - Name** | CG2 - J |
| **Input contribution** | m43318 |
| **Length & frame rate** | 97 frames (30 fps) |
| **Number of source views** | 25 (5x5) |
| **Source view resolution** | 1920x1080 |
| **Texture format** | YUV 4:2:0 10 bit |
| **Depth format** | YUV 4:2:0 10 bit |
| **View FoV & mapping** | 53.1° × 31.4° Rectilinear |
| **Lens** | 32 mm |
| **Camera spacing** | 10cm x 10cm |
| **zNear** | 2.2 |
| **zFar** | 7.2 |
| **Download** | http://mpegfs.int-evry.fr/mpegcontent/ws-mpegcontent/MPEG-I/Orange/OrangeKitchen/CE  Data: v\*.zip |

## Natural content with estimated depth

### TechnicolorPainter

The general characteristics of the TechnicolorPainter sequence are summarized in Table 7 and source view naming in Table 23 form a 4x4 planar array and are numbered v0-0 to v3-3 following left to right and top to bottom scan order, as shown in Table 8. The refined depths proposed in [m47445] are used.

Table 7: Summary of the TechnicolorPainter sequence

|  |  |
| --- | --- |
| **Category - Name** | NC1 - D |
| **Input contributions** | m40010, m40011, m43366 and m47445. |
| **Length & frame rate** | 300 frames (30 fps) |
| **Number of source views** | 16 (4x4) |
| **Source view resolution** | 2048 × 1088 |
| **Texture format** | YUV 4:2:0 10 bit |
| **Depth format** | YUV 4:2:0 16 bit for MIV anchor, 10 bit for MV-HEVC anchor |
| **Download** | http://mpegfs.int-evry.fr/mpegcontent/ws-mpegcontent/MPEG-I/Technicolor/TechnicolorPainter/CE/  Data: TechnicolorPainter.zip |

Table 8: View numbering of the TechnicolorPainter camera array

|  |  |  |  |
| --- | --- | --- | --- |
| v0-0 | v1-0 | v2-0 | v3-0 |
| v0-1 | v1-1 | v2-1 | v3-1 |
| v0-2 | v1-2 | v2-2 | v3-2 |
| v0-3 | v1-3 | v2-3 | v3-3 |

### IntelFrog

The general characteristics of the IntelFrog sequence are summarized in Table 9 and source view positions in Table 25. The captured views form a 15x1 line and are numbered v0-0 to v14-0 following left to right scan order. The refined depths proposed in [m47445] are used; these depths do not exist for extreme view positions v0 and v14 and therefore only the views from v1 to v13 are used.

Table 9: Characteristics of the IntelFrog sequence

|  |  |
| --- | --- |
| **Category - Name** | NC1 - E |
| **Input contribution** | m43748, m44914 and m47445 |
| **Length & frame rate** | 300 frames (30fps) |
| **Number of source views** | 13 (13x1) |
| **Source view resolution** | 1920x1080 |
| **Texture format** | YUV 4:2:0 10 bit |
| **Depth format** | YUV 4:2:0 16 bit for MIV anchor, 10 bit for MV-HEVC anchor |
| **View FoV & mapping** | 63.65° × 38.47° Rectilinear |
| **Lens** | 2.16 mm |
| **Camera spacing** | 3.675 cm |
| **zNear** | 0.3 |
| **zFar** | 1.62 |
| **Download** | http://mpegfs.int-evry.fr/mpegcontent/ws-mpegcontent/MPEG-I/Intel/Frog/CE  Data: IntelFrog.zip |

### ETRIChef2

The general characteristics of the ETRIChef2 sequence are summarized in Table 10 and source view positions in Table 26. The captured views form a 5×4 planar array and are numbered v0-0 to v4-3 following left to right and top to bottom scan order.

Table 10: Characteristics of the ETRIChef2 sequence

|  |  |
| --- | --- |
| **Category - Name** | NC2 - K |
| **Input contribution** | m42542 and m46256 |
| **Length & frame rate** | 300 frames (30 fps) |
| **Number of source views** | 20 (5x4) |
| **Source view resolution** | 1920x1048 |
| **Texture format** | YUV 4:2:0 8 bit |
| **Depth format** | YUV 4:2:0 8 bit |
| **Download** | http://mpegfs.int-evry.fr/mpegcontent/ws-mpegcontent/MPEG-I/Etri/ETRIChef2  Texture: EChef2\_x\*\*y\*\*.rar  Depth: EChef2\_d\_x\*\*y\*\*.rar |

### PoznanFencing

The general characteristics of the PoznanFencing sequence are summarized in Table 11 and source view positions in Table 27. The captured views form a 10x1 linear arc and are numbered v0-0 to v9-0 following left to right scan order.

Table 11: Characteristics of the PoznanFencing sequence

|  |  |
| --- | --- |
| **Category - Name** | NC2 - L |
| **Input contribution** | m38247 |
| **Length & frame rate** | 250 frames (25 fps) |
| **Number of source views** | 10 |
| **Source view resolution** | 1920x1080 |
| **Texture format** | YUV 4:2:0 10 bit |
| **Depth format** | YUV 4:2:0 16 bit for MIV and MIV View anchors  YUV 4:2:0 8 bit for MV-HEVC anchor |
| **View FoV & mapping** | 63° × 48° |
| **Lens** | 4.5 mm |
| **Camera spacing** | 5 stereopairs (baseline: 22 cm) placed on arc (radius: 4 m),  angle between neighboring stereopairs: 15 degrees,  total angle of the system: 60 degrees |
| **zNear** | 3.5 |
| **zFar** | 7.0 |
| **Download** | http://mpegfs.int-evry.fr/mpegcontent/ws-mpegcontent/MPEG-I/Poznan/Poznan\_Fencing2/CE  Data: v\*.zip |

### ULBUnicornA

The general characteristics of the ULBUnicornA sequence are summarized in Table 12. The captured views form a 9×9 planar array and are numbered v0-0 to v8-8 following left to right and top to bottom scan order, in a similar way as shown in Table 8 for TechnicolorPainter.

Table 12: Summary of the ULBUnicornA sequence

|  |  |
| --- | --- |
| **Category - Name** | NC3 - F |
| **Input contribution** | m41083 |
| **Length & frame rate** | 1 frame |
| **Number of source views** | 81 (9x9) |
| **Source view resolution** | 1920 × 1080 |
| **Texture format** | YUV 4:2:0 8 bit |
| **Depth format** | YUV 4:2:0 8 bit |
| **Download** | http://mpegfs.int-evry.fr/mpegcontent/ws-mpegcontent/MPEG-I/ULB/ULB\_Unicorn/PLANES/Data\_2018-08  Texture: Plane\_A'\_Texture\_w\_Midpoints\_2018-08.zip  Depth: Plane\_A'\_Depth\_eDERS\_2018-08.zip |

# Software tools

The referenced tools are listed in Table 13, with source code location, documentation and version number.

Table 13: List of used tools

|  |  |  |  |
| --- | --- | --- | --- |
| **Tool name** |  | **Location** | **Tag/branch** |
| TMIV | [w18577] | <https://gitlab.com/mpeg-i-visual/tmiv> | v2.0 |
| WS-PSNR | [w18069] | <https://gitlab.com/mpeg-i-visual/wspsnr> | v2.0.1 |
| HM |  | <https://hevc.hhi.fraunhofer.de/svn/svn_HEVCSoftware/tags/HM-16.16> | 16.16 |
| VVS | [w18172] | http://mpegx.int-evry.fr/software/MPEG/Explorations/6DoF/VVS | V1.0 |
| HTM |  | <https://hevc.hhi.fraunhofer.de/svn/svn_3DVCSoftware/tags/HTM-13.0> | 13.0 |
| VMAF | [VMAF] | <https://github.com/Netflix/vmaf> | v1.3.14 |
| IV-PSNR | [w18709] | <https://gitlab.com/mpeg-i-visual/ivpsnr> | v1.0 |

## HM

HM 16.16 is used for the MIV anchor. For perspective sequences, it is used directly, while for omnidirectional sequences, it is used with 360Lib.

## HTM

MV-HEVC is used for generating the MV-HEVC anchor. MV-HEVC version 13.0 (macro HEVC\_EXT set to 1) [w17133], with the patch described in [w16522] applied to solve an implementation issue restricting the total number of views to be above 16.

## VVS

The Versatile View Synthesizer [w18172], named VVS in the rest of the document is used to synthesize views at all positions, both at intermediate and source positions. VVS configuration files are provided in attachments (see Annex 1).

## WS-PSNR

WS-PSNR is a tool in used for computing WS-PSNR or PSNR for objective metrics on images and is used to compare coding and synthesis results against uncompressed source views. For the omnidirectional ERP sequences, computation is done according to §4.2 of [w17197] and is applied to Y, U and V components, but only the Y component is used for evaluation.

The second version of WS-PSNR adds support for perspective images and configuration files using the regular PSNR method.

## VMAF

This tool provides VIF, MS-SSIM and VMAF quality metrics:

* Visual Information Fidelity (VIF) is a full reference image quality assessment index based on natural scene statistics and the notion of image information extracted by the human visual system.
* The structural similarity (SSIM) index is a method for predicting the perceived quality of digital television and cinematic pictures, as well as other kinds of digital images and videos. The SSIM index is a full reference metric. Multiscale SSIM (MS-SSIM) is conducted over multiple scales through a process of multiple stages of sub-sampling, reminiscent of multiscale processing in the early vision system.
* Video Multimethod Assessment Fusion (VMAF) is an objective full-reference video quality metric. The v0.6.1 4K model that is included with this version of VMAF shall be used.

## IV-PSNR

The PSNR for Immersive Video (IV-PSNR) metric is a full-reference metric based on the PSNR. It includes two major changes: the pixel shift, that considers that edges of the objects in the synthesized view may be shifted due to rounding errors, and the global color shift, that considers that different input views may have various color characteristics.

# Anchor definition

Three anchors are considered to encode the multi-view sequences:

* **MIV anchor**: encoding of patch atlases with TMIV + HEVC,
* **MIV view anchor**: encoding of subset of source views with each full coded view represented in a separate atlas using a single patch, with TMIV + HEVC,
* **MV-HEVC anchor**: encoding of views using inter-view prediction, with MV-HEVC, and synthesis done with VVS.

The description of the MIV anchor is provided in [w18470], and coding and synthesis specific details are provided in this section. The MIV view anchor uses a configuration that bypasses most of the MIV encoder. The MV-HEVC anchor, which uses inter-view prediction, is fully described in this section.

The general structure of the MIV anchor is given in provided in [w18470]. The general structure of the MV-HEVC anchor generation is represented in Figure 4. It consists of selecting a subset of the source views to be included in the anchor (possibly all), encoding those multiple views using MV-HEVC. The resulting bitstream is decoded and provides decoded views. From this set of decoded views, non-coded source views and intermediate views are synthesized using VVS reference tool.

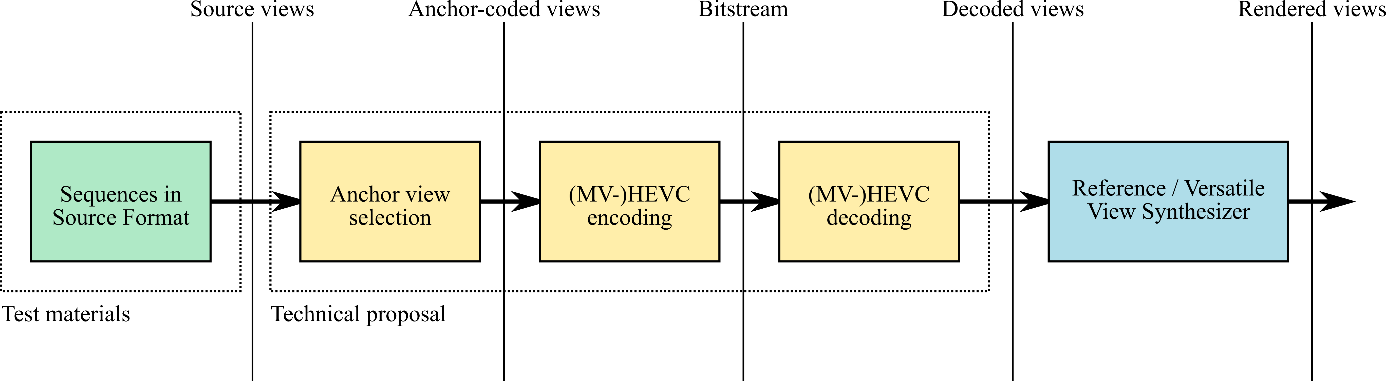


Figure 4: Definition of the anchor

## Coding of the anchor views

The frames to encode for each sequence are reflected in Table 14.

Table 14: Frames to encode.

|  |  |  |  |
| --- | --- | --- | --- |
| Id | Sequence | Start frame | Number of frames |
| CG1 – A | ClassroomVideo | 23 | 97 |
| CG1 – B | TechnicolorMuseum | 100 | 97 |
| CG1 – C | TechnicolorHijack | 0 | 97 |
| CG2 – H | OrangeShaman | 150 | 97 |
| CG2 – I | OrangeDancing | 190 | 16 |
| CG2 – J | OrangeKitchen | 0 | 97 |
| NC1 – D | TechnicolorPainter | 10 | 97 |
| NC1 – E | IntelFrog | 135 | 97 |
| NC2 – K | EtriChef2 | 8 | 97 |
| NC2 – L | PoznanFencing | 30 | 97 |
| NC3 – F | ULBUnicornA | 0 | 1 |

WS-PSNR (for omnidirectional content) or PSNR (for perspective content) is computed with the WS-PSNR tool referenced in Table 13.

Specific details for each anchor are given in the following sub-sections.

### Coding for MIV anchor

The coding of the MIV anchor is explained in [w18470]. For each video sequence, two sets of QP points are considered, medium and low, corresponding respectively to QP1, QP2, QP3, QP4, and QP2, QP3, QP4, QP5, as defined in Table 15.

Table 15: QPs used for depth and texture

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **QP1** | **QP2** | **QP3** | **QP4** | **QP5** |
| Texture QP | 22 | 27 | 32 | 37 | 42 |
| Depth QP | 12 | 17 | 22 | 27 | 32 |

The anchor encodes all source views with 16bits depths input. For each condition, WS-PSNR bitrate distortion curves are provided for each of the anchor-coded views, and for all anchor-coded views combined. The anchor bitstreams themselves include decoded picture hashes for automatic consistency checking.

### Coding for MIV view anchor

The same parameters are used as for the MIV anchor except for the following:

1) ViewOptimizerMethod = NoViewOptimizer

2) SourceCameraNames = Anchor-coded view as shown in Table 18

The NoViewOptimizer component forwards all source views as “basic” and none as “additional” such that the atlas constructor creates trivial atlases: one per source view and one big patch for per atlas.

Table 16: Anchor-coded views per class

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sequence Name** | **Format** | **# of source views** | **# of anchor-coded views** | **Anchor-coded views** |
| ClassroomVideo | ERP | 15 | 9 | v0, v7…v14 |
| TechnicolorMuseum | ERP | 24 | 8 | v0, v1, v4, v8, v11, v12, v13, v17 |
| TechnicolorHijack | semi-ERP | 10 | 5 | v1, v4, v5, v8, v9 |
| TechnicolorPainter | perspective | 16 | 8 | v0, v3, v5, v6, v9, v10, v12, v15 |
| IntelFrog | perspective | 13 | 7 | v1, v3, v5, v7, v9, v11, v13 |
| OrangeKitchen | perspective | 25 | 9 | v00, v02, v04, v10, v12, v14, v20, v22, v24 |
| PoznanFencing | perspective | 10 | 5 | v00, v02, v04, v06, v08 |

### Coding for MV-HEVC anchor

The anchor is generated with MV-HEVC. The coding configurations to generate the anchor are provided in attachment [A2] for textures and [A3] for depths. In particular, the following coding configuration is used:

* The inter-view prediction is set as described in Figure 5.
* The quantization parameters are given in Table 17.

Anchor results for the coded views are reported in the attached template file [A1].



Figure 5: Serpentine scan

Two sets of bitrates are considered, medium and low, corresponding respectively to QP1, QP2, QP3, QP4, and QP2, QP3, QP4, QP5, as defined in Table 17.

Table 17: QP used for depth and texture

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **QP1** | **QP2** | **QP3** | **QP4** | **QP5** |
| Texture QP | 25 | 30 | 35 | 40 | 45 |
| Depth QP | 25 | 30 | 35 | 40 | 45 |

The anchor encodes a subset of the source views or all source views, as defined in Table 18.

Table 18: Anchor-coded views

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Sequence | # of source  views | # of anchor-coded  views | Anchor-coded  views |
| CG2 – H | OrangeShaman | 25 (5x5) | 25 (5x5) | All |
| CG2 – I | OrangeDancing | 69 (23x3) | 69 (23x3) | All |
| CG2 – J | OrangeKitchen | 25 (5x5) | 25 (5x5) | All |
| NC1 – D | TechnicolorPainter | 16 (4x4) | 16 (4x4) | All |
| NC1 – E | IntelFrog | 13 (13x1) | 13 (13x1) | All |
| NC2 – K | EtriChef2 | 25 (5x5) | 25 (5x5) | All |
| NC2 – L | PoznanFencing | 10 (10x1) | 10 (10x1) | All |
| NC3 – F | ULBUnicornA | 81 (9x9) | 25 (5x5) | v0-0, v2-0, …, v8-0  v0,2, v2-2, …, v8-2, … |

## Synthesis of the intermediate views

Both for objective and subjective testing, a range of frames of each sequence are synthesized at source positions. For the synthesis, decoded available views except the target views are used as input of the view synthesis algorithm. For the objective evaluation, the frames as reported in Table 14 are used for view synthesis.

Proposals are not required to code views corresponding to all anchor-coded views but are required to be able to generate viewport and omnidirectional or perspective video sequences for any intermediate view position in the designated range for each test sequence. The field of view for ERP (e.g. 180° or 360° degrees) will be the same as the source content.

Specific details for each anchor are given in the following sub-sections.

### Synthesis for MIV anchors

The synthesis of the MIV anchor is explained in [w18470]. The format of each synthesized view is an omnidirectional image with equirectangular projection with the same angular resolution (pixels / degree) for ERP or semi-ERP test materials, and a linear perspective projection for linear perspective input content. The synthesis result is 10-bit YUV 4:2:0 format for subjective evaluation and 10-bit YUV 4:2:0 for objective evaluation.

Inpainting of invalid pixels is used for both subjective and objective testing.

For subjective viewing, each sequence is also synthesized according to a set of pose traces. An HMD pose trace specifies for each frame the position and orientation of the viewport to synthesize. Each pose trace is stored as a comma-separated table with position (X, Y, Z) and orientation (Yaw, Pitch, Roll) columns and exactly one row per frame of the sequence. The format of each synthesized view is an image with perspective projection with at most 2048 × 2048 pixels resolution, at most 90-degree field of view and 10-bit YUV 4:2:0 color format. The purpose is to mimic natural viewing on a head-mounted display (HMD) while using offline tools and a 2D monitor.

Because of the large difference in visual comfort between a viewer that voluntarily initiates head motion versus a viewer watching the same viewport on a 2D monitor, pose traces will have a small amount of motion. For each sequence there are three pose traces – named *X*p01, *X*p02 and *X*p03 – which are meant to represent a diversity of natural head movement compliant with the overall dimension of the capture rig, as indicated in Table 21. Attachment [A12] contains all pose traces. The TMIV decoder is configured to extend the video to 300 frames by mirroring the 97-frame sequences. ClassroomVideo pose traces have been replaced by pose traces that are 300 frames long. The other pose traces are the same as for the CfP [w18145].

It is meaningful to define the pose traces according to the conditions of capture, and typically to define the related path within the volume of the camera rig. It is convenient to formulate this range as a volume in 3D space, as in Table 19.

Table 19: Intermediate view position ranges

|  |  |  |
| --- | --- | --- |
| Test classes | Name | Range volume description |
| A1, A2 | ClassroomVideo | Spheroid centered at source view v0 eg (0, 0, 0) meter position, with equatorial radius 104 mm and polar distance 60 mm: |
| B1, B2 | TechnicolorMuseum | Sphere centered at position [0, 0, 1.65] meter with a 300 mm radius: |
| C1, C2 | TechnicolorHijack |
| D1, D2 | TechnicolorPainter | Spheroid centered at position [0, -0.35, -0.35] meter, covering a vertical square of side equal to 20cm and developed in the forward axis by 25cm max. |
| E1, E2 | IntelFrog | Rectangle centered at position [0, 0, 0] meter with a 15cm width, 44.1cm length, and no z component. |
| J1, J2 | OrangeKitchen | Spheroid centered at position [0, -0.4, -0.4] meter, covering a vertical square of side equal to 0.8m and developed in the forward axis by 1m max. |
| L1, L2 | PoznanFencing | Arc centered at position [-9.59, 3.66, 0.34] meter with a 2.34m width, 6.89m length, and no z component. The forward facing direction is [-0.33, 0.94, 0.02]. |

### Synthesis for MV-HEVC anchor

For the MV-HEVC anchor, VVS is used for synthesis, with the configuration files as provided in attachment [A6]. The camera parameters representing the positions of the views are provided in attachment [A7]. The decoded views used by VVS for the synthesis are the four closest reference views (left, top, bottom, up for a camera array) when available. When not available, only three or two of them are used (in the edges or the corners of the array, or for 1D).

Objective evaluation relies on comparisons made on source view positions. Subjective evaluation considers a navigation path made of coded views and intermediate views. The exact positions of each source view and intermediate views that follow the navigation path, and the reference views used to generate them are reported in attachment [A7]. Those anchor results for the synthesized views from decoded views are reported in the attached template file [A1], columns C-K.

# Rules for proponents

Experimental results are required for the sequences listed in Table 20. Experimental results may optionally be provided for the other sequences.

Table : Selection of test classes per anchor

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Id** | **Sequence** | **MIV anchor** | **MIV view anchor** | **MV-HEVC anchor** |
| CG1 – A | ClassroomVideo | x | x | x |
| CG1 – B | TechnicolorMuseum | x | x | x |
| CG1 – C | TechnicolorHijack | x | x | x |
| CG2 – H | OrangeShaman |  |  | x |
| CG2 – I | OrangeDancing |  |  | x |
| CG2 – J | OrangeKitchen | x | x | x |
| NC1 – D | TechnicolorPainter | x | x | x |
| NC1 – E | IntelFrog | x | x | x |
| NC2 – K | EtriChef2 |  |  | x |
| NC2 – L | PoznanFencing | x | x | x |
| NC3 – F | ULBUnicornA |  |  | x |

It is highly recommended for a proponent to prepare side-by-side comparisons of the anchor and the proposal of the pose traces for subjective viewing at the meeting, to provide additional information beyond the objective quality metrics. The general structure of the anchor generation is represented in Figure 4. It consists of encoding multiple views, with the anchor coded views possibly being a subset of the available source views. The resulting bitstream is decoded and provides decoded views. From this set of decoded views, non-coded source views and intermediate views are synthesized.

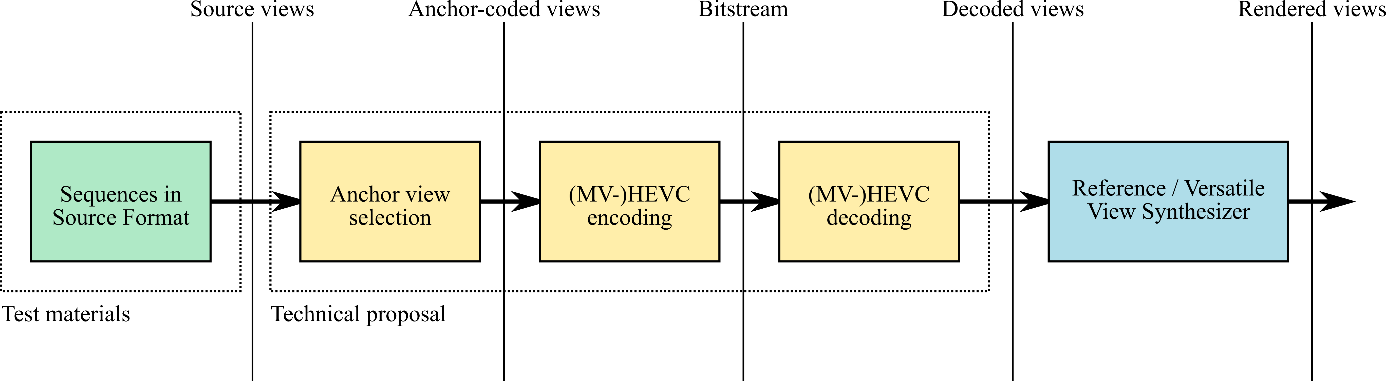


Figure 6: Definition of the anchor

# Objective evaluation of proposals

The following definitions apply:

* “**Coded source view**” corresponds to a source view that is coded by the anchor,
* “**Synthesized view**” corresponds to a source view that is synthesized (interpolated) by the anchor using multiple decoded source views, always excluding the target view.

**Objective quality**

Coded views (texture or depth) are evaluated on the frames of each of the sequences as reported in Table 14. The original view is the uncompressed source view. Synthesized views are evaluated on the same frames.

The new methods should be compared with the anchor coding results, by reporting the metrics as in Table 21. The columns PSNR diff., VIF diff., VMAF diff., MS-SSIM diff. and IV-PSNR diff. respectively represent the differences for the corresponding objective metrics, between the original source view and the synthesized view, averaged among the 5 QPs. All metrics except for VIF are mandatory.

Table 21: Presentation of the results.

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Configuration | Sequence | Coded source view BD rate | Synthesized view BD rate | Encoding time | Decoding time |  | Synthesis time | PSNR diff. | ViF diff. | VMAF diff. | MS-SSIM diff. | IV-PSNR diff. |
| Random Access | Sequence 1 | x% | x% | x% | x% |  | x% | xdB | xdB | xdB | xdB | xdB |
| Low bitrate | Sequence 2 | x% | x% | x% | x% |  | x% | xdB | xdB | xdB | xdB | xdB |
| **Average** | x% | x% | x% | x% |  | x% | xdB | xdB | xdB | xdB | xdB |
| Random Access | Sequence 1 | x% | x% | x% | x% |  | x% | xdB | xdB | xdB | xdB | xdB |
| High bitrate | Sequence 2 | x% | x% | x% | x% |  | x% | xdB | xdB | xdB | xdB | xdB |
|  | **Average** | x% | x% | x% | x% |  | x% | xdB | xdB | xdB | xdB | xdB |

For all test classes, WS-PSNR based BD-rate values will be provided for synthesized source views. For perspective views WS-PSNR reduces to regular PSNR. BD-rate values for coded source views will only be provided for the subset of views coded both by the anchor and proposal.

The comparison of proponents with the anchors will be expressed in terms of BD rate computed on low- and medium bitrate rate-distortion RD curves.

* **Coded source view BD rate**, only when anchor and proposal code the same set of views, obtained from:
  + For the anchor RD curve:
    - The average over each view and frames (Table 14) of the WS-PSNR (for ERP and semi-ERP content) or PSNR (for perspective content) between the coded view (coded texture) and the corresponding source view,
    - The total bitrate required to encode the views including texture and depths for all frames.
  + For the proponent’s RD curve:
    - The average over each view and all frames (Table 14) of the WS-PSNR (for ERP and semi-ERP content) or PSNR (for perspective content) between the proponent’s view (coded texture) corresponding to anchor coded view positions, and the source view,
    - The total bitrate of the proponent’s bitstream for all frames including texture, depth and metadata.
* **Synthesized view BD rate,** obtained from:
  + For the anchor RD curve:
    - The average over each source view and specified frames (Table 14) of the WS-PSNR (for ERP and semi-ERP content) or PSNR (for perspective content) between the intermediate view synthesized from decoded views and the original/non-compressed source views,
    - The total bitrate required to encode the views (including depths) for all frames.
  + For the proponent’s RD curve:
    - The average over each source view and specified frames (Table 14) of the WS-PSNR (for ERP and semi-ERP content) or PSNR (for perspective content) between the proponent’s synthesized intermediate view and the original/non-compressed source view,
    - The total bitrate of the proponent’s bitstream for all frames.
* Encoding, decoding and rendering runtimes ratios, compared to the anchor.

The reporting template provided as attachment [A1] should be used for all contributions. This template will compare a proposal with all anchors.

Because TMIV makes use of floating point operations, it is important to report the compiler and operating system that are used for evaluation. Preferred compilers are GCC 7 or newer and VC15. The TMIV software should be built in Release mode and the CMake project should be generated using one of the included build scripts.

**Pixel rate**

Objective evaluation criteria include pixel rate, which is included in the reporting template. Contributions are required to provide pixel rate for each tested sequence. Proponents should report results which they believe are the most optimal compromise between pixel rate and quality. To provide a meaningful reference for pixel rate values, the following constraints are defined:

**Low pixel rate test condition constraints:**

* + The combined maximum luma sample rate across all decoders is maximally 1,069,547,520 samples per second (e.g. 32 MP @ 30 fps, corresponding to HEVC Main 10 profile @ Level 5.2)
  + Each decoder instantiation is constrained to a maximum luma picture size of 8,912,896 pixels (e.g. 4096 x 2048, corresponding to HEVC Main 10 profile @ Level 5.2).
  + The maximum number of simultaneous decoder instantiations is four.
* **High pixel rate test condition constraints:**
  + The combined maximum luma sample rate across all decoders is maximally 4,278,190,080 samples per second (e.g. 128 MP @ 30 fps, corresponding to HEVC Main 10 profile @ Level 6.2)
  + Each decoder instantiation is constrained to a maximum luma picture size of 35,651,584 pixels (e.g. 8192 x 4096, corresponding to HEVC Main 10 profile @ Level 6.2).
  + The maximum number of simultaneous decoder instantiations is four.

These conditions are orthogonal to low/high bitrate conditions, and apply in principle to the MIV anchor, MIV view anchor and the MV-HEVC+VVS anchor and all proposals.

**Runtimes**

Runtimes should be reported for anchors and proposals:

* Metadata generation runtime (incl. all preprocessing)
* Video encoding and decoding runtime for texture and depth
* Rendering runtime (incl. all postprocessing)

The reference software includes measurement of CPU runtime, excluding loading from disk and writing to disk. Proposals should include a similar runtime measurement.

# Annex 1: List of attachments

Table 22 provides an overview of files that are provided together with this document. Attachments [A10], [A11], [A12] and [A13] are also included as part of the TMIV reference software.

Table 22: List of attachments to this document

|  |  |  |
| --- | --- | --- |
| **ID** | **Filename** | **Short description** |
| [A1] | reporting\_template.xlsm | 2 Excel templates for reporting objective results for MIV, MIV View anchors, and MV-HEVC anchor |
| [A2] | mvhevc\_texture\_cfg.zip | MV-HEVC encoder configuration for texture sequences |
| [A3] | mvhevc\_depth\_cfg.zip | MV-HEVC encoder configuration for depth sequences |
| [A4] | mvhevc\_vvs\_bit\_md5.rar | MD5 sums for the MV-HEVC anchor bitstreams |
| [A5] | mvhevc\_vvs\_yuv\_md5.rar | MD5 sums for the MV-HEVC anchor decoded YUVs |
| [A6] | vvs\_cfg.zip | VVS configuration for MV-HEVC coded content |
| [A7] | vvs\_camparam.zip | Camera parameters representing view positions for MV-HEVC coded content |
| [A8] | vvs\_depth\_range.zip | Znear and ZFar values for each view, for MV-HEVC coded content |
| [A9] | vvs\_navigation\_paths.rar | Navigation paths for subjective evaluation of MV-HEVC coded content |
| [A10] | tmiv\_config.zip | Configuration files for MIV anchor and MIV view anchor |
| [A11] | tmiv\_camparam.zip | Metadata files according to the updated format [w18068] with source and intermediate view positions for all sequences (omnidirectional and perspective) |
| [A12] | tmiv\_posetraces.zip | Three pose traces per sequence |
| [A13] | source\_md5.zip | MD5 sums for all source data |

# Annex 2: Source view label conversion

Table 23: View naming of the TechnicolorPainter sequence

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Original view name** | **Json view name** | **Original view name** | **Json view name** | **Original view name** | **Json view name** | **Original view name** | **Json view name** |
| v0-0 | v0 | v1-0 | v1 | v2-0 | v2 | v3-0 | v3 |
| v0-1 | v4 | v1-1 | v5 | v2-1 | v6 | v3-1 | v7 |
| v0-2 | v8 | v1-2 | v9 | v2-2 | v10 | v3-2 | v11 |
| v0-3 | v12 | v1-3 | v13 | v2-3 | v14 | v3-3 | v15 |

Table 24: View naming of the OrangeShaman & OrangeKitchen sequences

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Original view name** | **Json view name** | **Original view name** | **Json view name** | **Original view name** | **Json view name** | **Original view name** | **Json view name** | **Original view name** | **Json view name** |
| v0-0 | v00 | v0-1 | v05 | v0-2 | v10 | v0-3 | v15 | v0-4 | v20 |
| v1-0 | v01 | v1-1 | v06 | v1-2 | v11 | v1-3 | v16 | v1-4 | v21 |
| v2-0 | v02 | v2-1 | v07 | v2-2 | v12 | v2-3 | v17 | v2-4 | v22 |
| v3-0 | v03 | v3-1 | v08 | v3-2 | v13 | v3-3 | v18 | v3-4 | v23 |
| v4-0 | v04 | v4-1 | v09 | v4-2 | v14 | v4-3 | v19 | v4-4 | v24 |

Table 25: View naming of the IntelFrog sequence

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Original view name** | **Json view name** | **Original view name** | **Json view name** | **Original view name** | **Json view name** | **Original view name** | **Json view name** |
| cam00 | v14 | cam04 | v10 | cam08 | v06 | cam12 | v02 |
| cam01 | v13 | cam05 | v09 | cam09 | v05 | cam13 | v01 |
| cam02 | v12 | cam06 | v08 | cam10 | v04 | cam14 | v00 |
| cam03 | v11 | cam07 | v07 | cam11 | v03 |  |  |

Table 26: View naming of the ETRIChef sequence

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Original view name** | **Json view name** | **Original view name** | **Json view name** | **Original view name** | **Json view name** | **Original view name** | **Json view name** |
| v0-0 | v00 | v0-1 | v05 | v0-2 | v10 | v0-3 | v15 |
| V1-0 | v01 | V1-1 | v06 | V1-2 | v11 | V1-3 | v16 |
| V2-0 | v02 | V2-1 | v07 | V2-2 | v12 | V2-3 | v17 |
| V3-0 | v03 | V3-1 | v08 | V3-2 | v13 | V3-3 | v18 |
| V4-0 | v04 | V4-1 | v09 | V4-2 | v14 | V4-3 | v19 |

Table 27: View naming of the PoznanFencing sequence

|  |  |  |  |
| --- | --- | --- | --- |
| **Original view name** | **Json view name** | **Original view name** | **Json view name** |
| v0-0 | v00 | v5-0 | v05 |
| v1-0 | v01 | v6-0 | v06 |
| v2-0 | v02 | v7-0 | v07 |
| v3-0 | v03 | v8-0 | v08 |
| v4-0 | v04 | v9-0 | v09 |

# References

[VMAF] [vmaf: Perceptual video quality assessment based on multi-method fusion](https://github.com/Netflix/vmaf), Netflix, Inc., 2017-07-14.

[w17462] *Call for MPEG-I Visual Test Materials on 6DoF*, ISO/IEC JTC1/SC29/WG11 MPEG/N17462, January 2018, Gwangju, Korea.

[w18068] Reference View Synthesizer (RVS) manual, ISO/IEC JTC1/SC29/WG11 MPEG/N18068, October 2018, Macau SAR, China.

[w18069] ERP WS-PSNR manual, ISO/IEC JTC1/SC29/WG11 MPEG/N18069, October 2018, Macau SAR, China.

[w17197] Algorithm descriptions of projection format conversion and video quality metrics in 360Lib Version 5, JVET-H1004, ISO/IEC JTC1/SC29/WG11 MPEG/N17197, October 2017, Macau SAR, China.

[w17133] Limitations of Multiview extensions of HEVC and fixes for MPEG-I Phase 2, ISO/IEC JTC1/SC29/WG11 MPEG/N17133, October 2017, Macau SAR, China.

[w16522] FTV Software Summary, ISO/IEC JTC1/SC29/WG11 MPEG/N16522, October 2016, Chengdu, China.

[w18172] Versatile View Synthesizer (VVS) 2.0 manual, ISO/IEC JTC1/SC29/WG11 MPEG/N18172, January 2019, Marrakesh, Morocco.

[w18145] Call for Proposals on 3DoF+ Visual, ISO/IEC JTC1/SC29/WG11 MPEG/N18145, January 2019, Marrakesh, Morocco.

[w18577] Test Model 2 for Immersive Video, ISO/IEC JTC1/SC29/WG11 MPEG/N18577, July 2019, Gothenburg, Sweden.

[w18709] Software manual of IV-PSNR for Immersive Video, ISO/IEC JTC1/SC29/WG11 MPEG/N18709, July 2019, Gothenburg, Sweden.