INTERNATIONAL STANDARD

ISO/IEC 23090-14

First edition
2023-06

AMENDMENT 1
2023-11

Information technology — Coded representation of immersive media —

Part 14: Scene description

AMENDMENT 1: Support for immersive media codecs in scene description

Technologies de l'information — Représentation codée de média immersifs —

Partie 14: Description de scènes

AMENDEMENT 1: Support pour les codecs des médias immersifs dans la description de scènes

Reference number
ISO/IEC 23090-14:2023/Amd. 1:2023(E)

ISO copyright office
CP 401 • Ch. de Blandonnet 8
CH-1214 Vernier, Geneva
Phone: +41 22 749 01 11
Email: copyright@iso.org
Website: www.iso.org

Published in Switzerland

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of document should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives or www.iec.ch/members_experts/refdocs).

ISO and IEC draw attention to the possibility that the implementation of this document may involve the use of (a) patent(s). ISO and IEC take no position concerning the evidence, validity or applicability of any claimed patent rights in respect thereof. As of the date of publication of this document, ISO and IEC had not received notice of (a) patent(s) which may be required to implement this document. However, implementers are cautioned that this may not represent the latest information, which may be obtained from the patent database available at www.iso.org/patents and https://patents.iec.ch. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation of the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT) see www.iso.org/iso/foreword.html. In the IEC, see www.iec.ch/understanding-standards.

This document was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 29, *Coding of audio, picture, multimedia and hypermedia information*.

A list of all parts in the ISO/IEC 23090 series can be found on the ISO and IEC websites.

Any feedback or questions on this document should be directed to the user's national standards body. A complete listing of these bodies can be found at www.iso.org/members.html and www.iec.ch/national-committees.

Information technology — Coded representation of immersive media —

Part 14: Scene description

AMENDMENT 1: Support for immersive media codecs in scene description

Normative references

Add the following references:

ISO/IEC 23090-5, *Information technology — Coded Representation of Immersive Media — Part 5: Visual Volumetric Video-based Coding (V3C) and Video-based Point Cloud Compression (V-PCC)*

4.2

Replace Figure 1 by the following figure

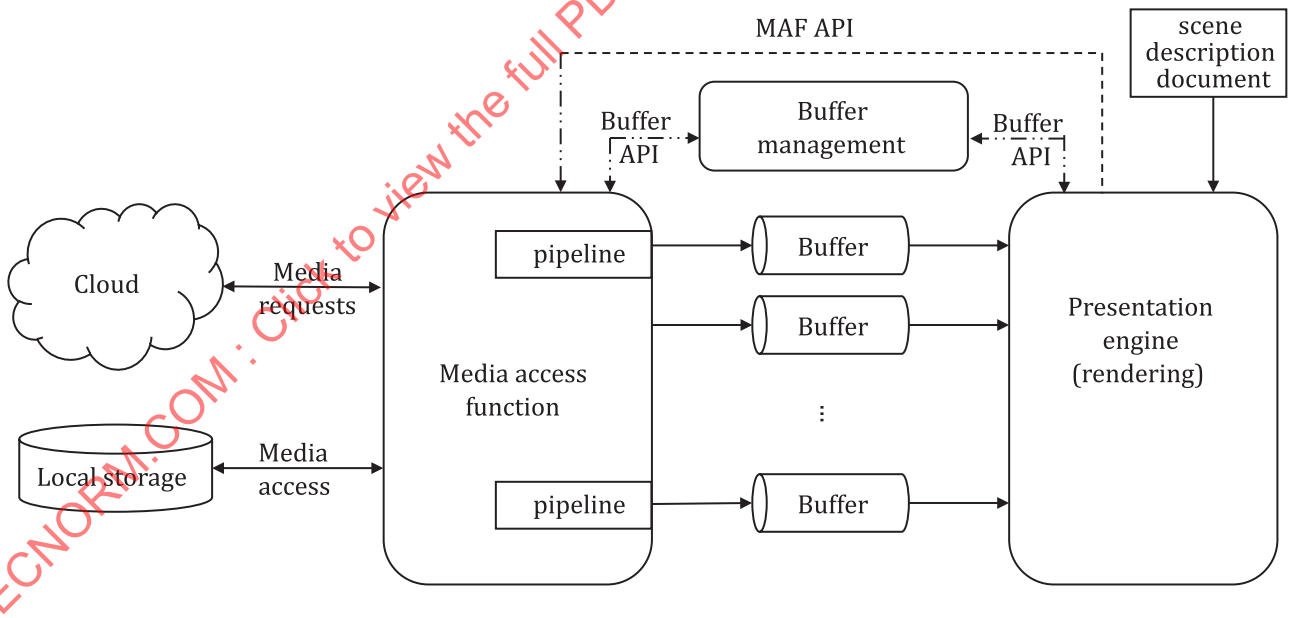

Figure 1 — Scene description reference architecture

3.2

Add the following to the list of abbreviated terms in subclause 3.2:

MIV MPEG immersive video

ERP Equirectangular projection

PLR Point Local Reconstruction

EOM Enhanced Occupancy Mode

5.1.1

Add the following sentence after Figure 3

Additional extensions and buffer formats for the support of MPEG-specified immersive media formats in MPEG-I scene description are specified in Annex G.

5.3.1.2, Table 11

Change the Description of the format attribute as follows:

format	string	RGB	O	Indicates the format of the pixel data for this video texture. The allowed values are: RED, GREEN, BLUE, RG, RGB, RGBA, BGR, BGRA, DEPTH_COMPONENT. The semantics of these values are defined in Table 8.3 of OpenGL specification [2]. Additionally, YCbCr formats are supported. The semantics for the YCbCr formats are defined in Table 76 in Vulkan specification [Vulkan 1.3]. A sampler with the `MPEG_sampler_YCbCr` extension shall be linked to a YCbCr texture. The number of components shall match the type indicated by the referenced accessor. Normalization of the pixel data shall be indicated by the normalized attribute of the accessor.

5.2.1.2, Table 6,

Change the Description of the track attribute as follows:

Table 6 — Definitions of items in the tracks array of MPEG_media.alternative extension

Name	Type	Default	Usage	Description
track	string	N/A	M	URL fragment to access the track within the media alternative. The URL structure is defined for the following formats: DASH: Using MPD Anchors (URL fragments) as defined in ISO/IEC 23009-1:2019:Annex C (Table C.1). ISOBMFF: URL fragments as specified in ISO/IEC 14496-12:2020:Annex C. SDP: stream identifier of the media stream as defined in Annex C. When V3C data is referenced in the scene description document as in item in MPEG_media.alternative.tracks and the referenced item corresponds to an ISBOBMFF track, the following applies: — For single-track encapsulated V3C data, the referenced track in MPEG_media shall be the V3C bitstream track. — For multi-track encapsulated V3C data, the referenced track in MPEG_media shall be the V3C atlas track. When G-PCC data is referenced by the scene description file as an item in MPEG_media.alternative.tracks and the referenced item complies with the provisions of track in ISOBMFF, the following applies: — For single-track encapsulated G-PCC data, the track referenced in MPEG_media shall be the G-PCC bitstream track; — For multi-track encapsulated G-PCC data, the track referenced in MPEG_media shall be the G-PCC geometry bitstream track.
codecs	string	N/A	M	The codecs parameter, as defined in IETF RFC 6381, of the media included in the track. When the track includes different types of codecs (e.g. the AdaptationSet includes Representations with different codecs), the codecs parameter may be signaled by comma-separated list of values of the codecs.

Annex B

Add the following entries to Table B.1 in Annex B:

Table B.1 — MPEG attribute registry

Name	Accessor type(s)	Component type(s)	Description	Reference and example shader program
_MPEG_V3C_ATTR_REFLECTANCE	scalar	5123	indicates the reflectance information that is associated with each point in a volumetric frame	
_MPEG_V3C_ATTR_MATERIAL_ID	scalar	5123	indicates a supplemental information that identifies material type of a point in a volumetric frame	
_MPEG_V3C_ATTR_TRANSPARENCY	scalar	5123	indicates the transparency information that is associated with each point in a volumetric frame	

Annex F

Add the following subclauses to Annex F:

F.10 MPEG_primitive_V3C

In the example downloadable from https://standards.iso.org/iso-iec/23090/-14/ed-1/en/amd/1/example_MPEG_primitive_V3C, a usage of the `MPEG_primitive_V3C` is presented.

F.11 MPEG_sampler_YCbCr

In the example downloadable from https://standards.iso.org/iso-iec/23090/-14/ed-1/en/amd/1/example_MPEG_sampler_YCbCr, a usage of the `MPEG_sampler_YCbCr` extension is presented.

Add Annex G with the following content

Annex G
(normative)

Support for MPEG-I Media

G.1 MPEG_primitive_V3C extension

G.1.1 General

In order to support V3C compressed objects in MPEG-I scene description, the `MPEG_media` extension is used to refer to V3C compressed bitstreams.

The presentation engine may support the operations to perform the 3D reconstruction of decoded V3C components as indicated in the Figure 2. The presentation engine accesses the decoded V3C data through buffers.

The syntax of the V3C object is provided as an extension to `mesh.primitive` in a scene description format. The extension refers to the decoded data of a V3C object. Each decoded V3C component is signalled using properties defined in the `MPEG_primitive_V3C` extension. The extension is specific to objects coded with a V3C compression scheme (i.e., ISO/IEC 23090-5 or ISO/IEC 23090-12).

Usage of the extension shall be listed in the *`extensionsUsed`* top-level glTF property.

```
 "extensionsUsed": [

    "MPEG_primitive_V3C"

]
```

Figure G.1 depicts the structure of the V3C mesh compression extension:

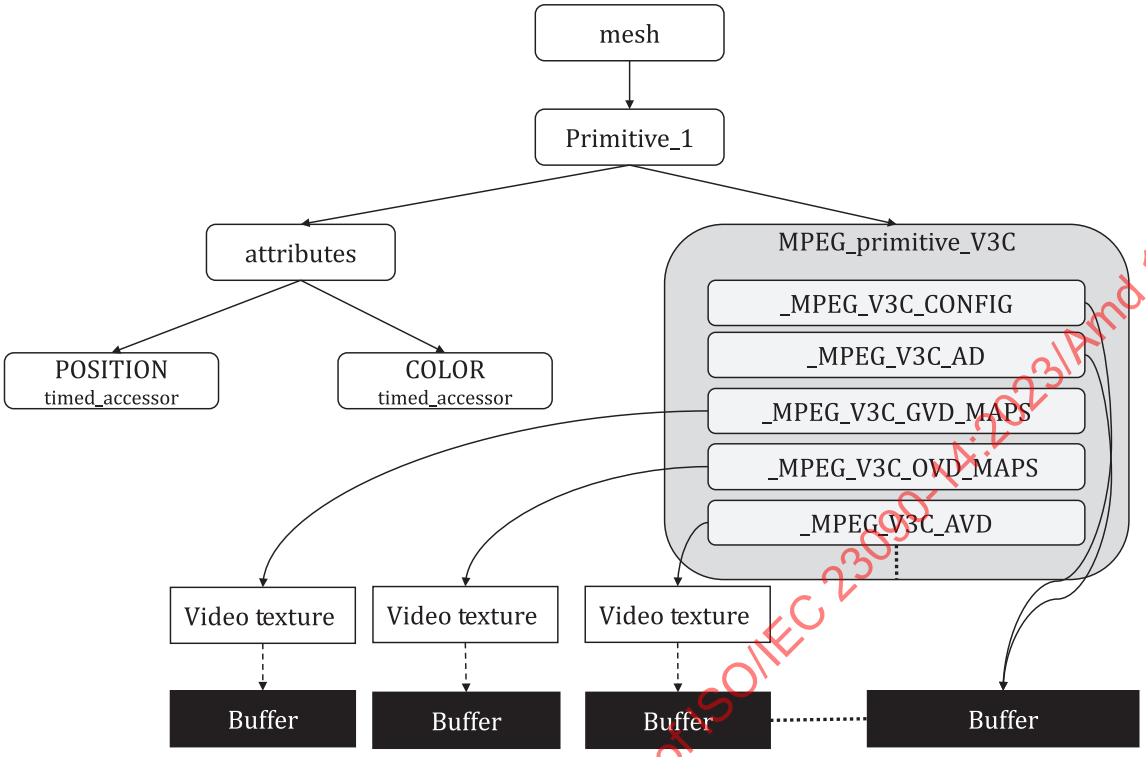

Figure G.1 — Example structure of V3C compressed primitive

If the Presentation Engine does not support the MPEG_primtive_V3C extension, It shall request the reconstructed raw data as described by the primitive attributes.

G.1.2 Semantics

An `MPEG_primitive_V3C` extension refers to several V3C components, containing the decoded projected maps and metadata necessary such as atlas data for the 3D reconstruction process.

Table G.1 provides a list of the possible components and their description:

Table G.1 — MPEG_primitive_V3C properties

Name	Type	Default	Usage	Description
_MPEG_V3C_CONFIG	integer	N/A	M	This component provides a reference to a timed accessor that contains configuration information that is applicable to a sequence of frames of the V3C decoded mesh primitive. The binary format of the configuration buffer is provided in clause G.1.3.
Legend: For attributes: M=mandatory, O=optional, OD=optional with default value, CM=conditionally mandatory.				

Table G.1 *(continued)*

Name	Type	Default	Usage	Description
_MPEG_V3C_AD	object	N/A	M	this component shall reference a timed accessor that provides the V3C atlas data buffer. The atlas buffer format is defined in clause G.1.4. Future specifications of the atlas data buffer format shall use a different version. Exactly one atlas component shall be present, irrespective of the version.
_MPEG_V3C_GVD_MAPS	array(integer)	N/A	M	this component shall provide an array of video texture references, each of which corresponds to one map of the decoded geometry video data.
_MPEG_V3C_OVD_MAP	integer	N/A	O	this component shall provide a video texture reference, which corresponds to the decoded occupancy video data map.
_MPEG_V3C_AVD	array(object)	N/A	O	this component shall provide an array of objects, each of which describing an attribute component of the V3C compressed mesh primitive. The properties of the components are described in Table G.2.
_MPEG_V3C_CAD	object	N/A	CM	This object lists different properties described for the Common Atlas Data in ISO/IEC 23090-5.
Legend: For attributes: M=mandatory, O=optional, OD=optional with default value, CM=conditionally mandatory.				

The `_MPEG_V3C_AD` object shall have the structure as describe in Table G.2:

Table G.2 — Properties of _MPEG_V3C_AD object

Name	Type	Default	Usage	Description
buffer_format	string	"baseline"	O	provides an identifier of the associated atlas data buffer format. A list of supported atlas data buffer formats is provided in Table G.4.
accessor	integer	N/A	M	This provides the index of the timed accessor that provides access to the atlas data buffer.

The `_MPEG_V3C_AVD` object shall have the following structure:

Table G.3 — Properties of _MPEG_V3C_AVD object

Name	Type	Default	Usage	Description
type	uint8	0	O	provides the type of the attribute as defined by the "V3C attribute types" in ISO/IEC 23090-5.
maps	array(integer)	N/A	M	This array shall provide a list of video texture references, each of which corresponds to one map of the decoded attribute video data.

Each mesh primitive shall reference exactly one atlas data buffer.

Different buffer formats with unique string identifier for the atlas data are defined in Table G.4.

Table G.4 — List of atlas data buffer formats

buffer_format	Description
baseline	The configuration data is defined in Table G.5 and the corresponding atlas data buffer format is defined in Table G.6.
extended	Atlas data with common atlas parameters and and PROJECTED patch type application-specific data with PLR information, EOM patch type application-specific data, and RAW patch type application-specific data. The configuration data format is defined in G.5 and the corresponding atlas data buffer format is defined in Table G.7.
miv	Atlas data with common atlas parameters and PROJECTED patch type application-specific parameters for MIV. The configuration data format is defined in G.5 and the corresponding atlas data buffer format is defined in Table G.8.

G.1.3 Configuration Data Buffer Format

The configuration data buffer is binary formatted data that provides static configuration data that is applicable for the V3C compressed primitive. The data shall comply to the following format:

Table G.5 — Configuration data buffer format

Field	Type	Description
frame_width	uint16	indicates the frame width in luma samples of the atlas and all other associated V3C components.
frame_height	uint16	indicates the frame height in luma samples of the atlas and all other associated V3C components.
map_count	uint8	indicates the number of maps used for encoding the geometry and attribute data for the current atlas.
patch_packing_block_size	uint8	specifies the value of the variable PatchPackingBlockSize in ISO/IEC 23090-5, that is used for the horizontal and vertical placement of the patches within the current atlas.

G.1.4 Atlas Data Buffer Format

The atlas data buffer is binary formatted data that shall comply to the following formats in Table G.6, Table G.7 and Table G.8 depending on the buffer format for the atlas data. The atlas buffer format for "`buffer_format`" value "baseline" is described in Table G.7.

Table G.6 — Atlas data buffer format for buffer_format:baseline

Field	Type	Description
patch_count	uint16	provides the total number of patches.
for(i=0;i<patch_count;i++) {		
2d_pos_x	float	specifies the x-coordinate of the top-left corner of the patch bounding box for the current patch.
2d_pos_y	float	specifies the y-coordinate of the top-left corner of the patch bounding box for the current patch.
2d_size_x	float	specifies the width of the current patch.
2d_size_y	float	specifies the height of the current patch.
3d_offset_u	float	specifies the shift to be applied to the reconstructed patch points in the current patch along the tangent axis.
3d_offset_v	float	specifies the shift to be applied to the reconstructed patch points in the current patch along the bi-tangent axis.
3d_offset_d	float	specifies the shift to be applied to the reconstructed patch points in the current patch along the normal axis.
patch_projection_id	uint8	specifies the identifier of the projection mode and the index of the normal to the projection plane of the current patch.
patch_orientation	uint8	specifies the index of the patch orientation of the current patch.
lod_scale_x	uint16	specifies the LOD scaling factor to be applied to the tangent axis of the current patch.
lod_scale_y	uint16	specifies the LOD scaling factor to be applied to the bi-tangent axis of the current patch.
}		

The atlas buffer format for "`buffer_format`" value "extended" is described in Table G.7.

Table G.7 — Atlas data buffer format for buffer_format:extended

Field	Type	Description
patch_count	uint16	provides the total number of patches.
for (i=0;i<patch_count;i++) {		
patch_type	uint8	specifies the type of patch
2d_pos_x	float	specifies the x-coordinate of the top-left corner of the patch bounding box for the current patch.
2d_pos_y	float	specifies the y-coordinate of the top-left corner of the patch bounding box for the current patch.
2d_size_x	float	specifies the width of the current patch.
2d_size_y	float	specifies the height of the current patch.
3d_offset_u	float	specifies the shift to be applied to the reconstructed patch points in the current patch along the tangent axis.
3d_offset_v	float	specifies the shift to be applied to the reconstructed patch points in the current patch along the bi-tangent axis.
3d_offset_d	float	specifies the shift to be applied to the reconstructed patch points in the current patch along the normal axis.
patch_projection_id	uint8	specifies the identifier of the projection mode and the index of the normal to the projection plane of the current patch.

Table G.7 *(continued)*

Field	Type	Description
patch_orientation	uint8	specifies the index of the patch orientation of the current patch.
lod_scale_x	uint16	specifies the LOD scaling factor to be applied to the tangent axis of the current patch.
lod_scale_y	uint16	specifies the LOD scaling factor to be applied to the bi-tangent axis of the current patch.
if (patch_type == PROJECTED) {		
plri_map_present	bool	specifies if the plr information is present
if (plri_map_present) {		
plrd_level	bool	specifies the level of PLR data for a patch
if (plr_level == 0) {		
for (b = 0; b < blockcount < b++) {		
plrd_present_block_flag	bool	specifies whether the PLR data is present for a block
if (plrd_present_block_flag == 1) {		
plrd_block_mode	uint8	specifies the mode of PLR data for a block
}		
}		
} else {		
plrd_present_patch_flag	bool	specifies whether the PLR data is present for the patch
if (plrd_present_patch_flag) {		
plrd_patch_mode	uint8	specifies the mode of the PLR data for the patch
}		
}		
}		
else if (patch_type == EOM)		
eom_patch_count	uint8	specifies the number of patches that may be associated with the current patch
eom_points	uint8	specifies the number of EOM coded points in the patch associated with the current patch
associated_patch_index	uint8	specifies the index of the i-th patch associated with the current patch
}		
else if (patch_type == RAW)		
raw_points	uint8	specifies the number of RAW coded points in the current patch
}		
}		

NOTE The calculation of `blockCount` is specified in function `BlockCnt (xSize, ySize)` in ISO/IEC 23090-5:2021, 8.4.7.9. The arguments to the function are patch dimensions, i.e., `2d_size_x`, and `2d_size_y`.

The atlas buffer format for "`buffer_format`" value "miv" is described in Table G.8.

Table G.8 — Atlas data buffer format for buffer_format:miv

Field	Type	Description
patch_count	uint16	provides the total number of patches.
for(i=0;i<patch_count;i++) {		
2d_pos_x	float	specifies the x-coordinate of the top-left corner of the patch bounding box for the current patch.
2d_pos_y	float	specifies the y-coordinate of the top-left corner of the patch bounding box for the current patch.
2d_size_x	float	specifies the width of the current patch.
2d_size_y	float	specifies the height of the current patch.
3d_offset_u	float	specifies the shift to be applied to the reconstructed patch points in the current patch along the tangent axis.
3d_offset_v	float	specifies the shift to be applied to the reconstructed patch points in the current patch along the bi-tangent axis.
3d_offset_d	float	specifies the shift to be applied to the reconstructed patch points in the current patch along the normal axis.
patch_projection_id	uint8	specifies the identifier of the projection mode and the index of the normal to the projection plane of the current patch.
patch_orientation	uint8	specifies the index of the patch orientation of the current patch.
lod_scale_x	uint16	specifies the LOD scaling factor to be applied to the tangent axis of the current patch.
lod_scale_y	uint16	specifies the LOD scaling factor to be applied to the bi-tangent axis of the current patch.
patch_view_index	uint8	specifies the index in the buffer format for the view parameter
patch_entity_id	uint8	specifies the patch entity ID for the current patch
patch_depth_occ_threshold	uint8	specifies the threshold below with the occupancy value is defined to be unoccupied for the current patch
tile_patch_texture_offset_1	uint8	specifies the offset applied to the first component sample values of the attribute for the current patch
tile_patch_texture_offset_2	uint8	specifies the offset applied to the second component sample values of the attribute for the current patch
tile_patch_texture_offset_3	uint8	specifies the offset applied to the third component sample values of the attribute for the current patch
}		

G.1.5 Common atlas data

G.1.5.1 Overview

The common atlas data is common to all atlases and shall correspond to the Common Atlas Data in ISO/IEC 23090-5.

G.1.5.2 MIV extension to CAD

Some of the common atlas data information which is common for the atlases in a V3C bitstream is specified in ISO/IEC 23090-12 such as view parameters. The syntax for the MIV extension to common atlas data is specified ISO/IEC 23090-12. It includes a list of view parameters which can be used during

the rendering process (ISO/IEC 23090-12:2023, H.1). An `MIV_view_parameters` property is defined for the `_MPEG_V3C_CAD` object as shown in Table G.9.

The `_MPEG_V3C_CAD` object can be extended to describe additional properties that may be introduced in future iterations of ISO/IEC 23090-5 or in extensions to that specification.

Table G.9 — Definition of properties defined in _MPEG_V3C_CAD in MPEG_V3C extension

Name	Type	Default	Usage	Description
mivViewParameters	integer	N/A	O	This component provides a reference to a timed-accessor that contains the view parameters stored in the common atlas data that is applicable to a sequence of frames of the V3C decoded mesh primitive. The buffer format for the view parameters is described in Table G.10.
Legend: For attributes: M=mandatory, O=optional, OD=optional with default value, CM=conditionally mandatory.				

G.1.5.3 Buffer format for MIV view parameters

Table G.10 describes the binary buffer format for view parameters.

Table G.10 — Buffer format for view parameters

Field	Type	Description
num_views	uint16	number of views
for (int p = 0; p < num_views ; p++) {		
view_id_to_index	uint8	mapping of the id associated with each view
view_in_paint_flag	bool	specifies if the view is an inpaint view
view_pos_x	uint8	specifies in scene units the x-coordinate of the location of the view with view index equal to v.
view_pos_y	uint8	specifies in scene units the y-coordinate of the location of the view with view index equal to v.
view_pos_z	uint8	specifies in scene units the z-coordinate of the location of the view with view index equal to v.
view_quat_x	uint8	specifies the x components for the rotation of the view with view index equal to v using the quaternion representation
view_quat_y	uint8	specifies the y components for the rotation of the view with view index equal to v using the quaternion representation
view_quat_z	uint8	specifies the z components for the rotation of the view with view index equal to v using the quaternion representation
view_quat_w	uint8	specifies the w components for the rotation of the view with view index equal to v using the quaternion representation
view_type	uint8	specifies the projection method of the view
projection_plane_width	uint8	specifies the horizontal resolution of projection plane
projection_plane_height	uint8	specifies the vertical resolution of the projection plane
if (view_type == 0) {		equirectangular projection
erp_phi_min	float32	specifies the minimum longitude range for an ERP projection in units of degrees
erp_phi_max	float32	specifies the maximum longitude range for an ERP projection in units of degrees
erp_theta_min	float32	specifies the minimum latitude range for an ERP projection in units of degrees