|  |  |
| --- | --- |
| **Joint Collaborative Team on Video Coding (JCT-VC)**  **of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11**  28th Meeting: Torino, IT, 15–21 July 2017 | Document: JCTVC-AB1005-v1 |

|  |  |  |  |
| --- | --- | --- | --- |
| *Title:* | HEVC Additional Supplemental Enhancement Information (Draft 3) | | |
| *Status:* | Output document approved by JCT-VC | | |
| *Purpose:* | Draft text for standardization | | |
| *Author(s) or Contact(s):* | Jill Boyce  Adarsh Ramasubramanian  Robert Skupin  Gary J. Sullivan  Alexis Tourapis  Ye-Kui Wang | Email: | jill.boyce@intel.com |
| *Source:* | Editors | | |

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# Abstract

This document is the draft text for changes to the High Efficiency Video Coding (HEVC) standard (Rec. ITU-T H.265 | ISO/IEC 23008-2) to specify additional supplemental enhancement information (SEI) messages for content colour volume, omnidirectional 360° projection, omnidirectional viewport, regional nesting, and motion-constrained tile sets extraction information, along with some corrections to existing specification texts.

**Changes to the specification text:**

*Throughout the specification, replace all single-word instances of "nested" with "scalable-nested" (e.g., there is no single-word instance of "nested" in "non-nested"), and all instances of "non-nested" with "non-scalable-nested", except in 3.158 (which is kept unchanged) and Table F.4 in F.14.3.1 (for which a change is provided below).*

*In clause 3, add the following term definitions (in alphabetical order within the current list):*

**3.X** **azimuth circle**: circle on a sphere connecting all points with the same azimuth value.

NOTE 1 to Entry – An azimuth circle is always a *great circle* like a longitude line on the earth.

**3.X** **constituent picture**: part of a frame-packed stereoscopic video picture that corresponds to one view, or a non-frame-packed monoscopic video picture itself.

**3.X** **coverage sphere region**: *sphere region* that is covered by a *cropped decoded picture*.

**3.X** **elevation circle**: circle on a sphere connecting all points with the same elevation value.

NOTE 1 to Entry – An elevation circle is similar to a lattitude line on the earth. Except when the elevation value is zero, an elevation circle is not a *great circle* like a longitude circle on the earth.

**3.X** **global coordinate axes**: coordinate axes associated with *omnidirectional video* that are associated with an externally referenceable position and orientation.

NOTE 1 to Entry – The global coordinate axes may correspond to the position and orientation of a device or rig used for omnidirectional audio/video acquisition as well as the position of an observer's head in the three-dimensional space of the *omnidirectional video* rendering environment.

**3.X** **great circle**: intersection of a sphere and a plane that passes through the centre point of the sphere.

NOTE 1 to Entry – A great circle is also known as an orthodrome or Riemannian circle.

**3.X** **local coordinate axes**: coordinate axes having a specified rotation relationship relative to the *global coordinate axes*.

**3.X** **omnidirectional video**: video content in a format that enables rendering according to the user's viewing orientation, e.g., if viewed using a head-mounted device, or according to a user's desired *viewport*, reflecting a potentially rotated viewing position.

**3.X** **packed region**: region in a *region-wise packed picture* that is mapped to a *projected region* according to a *region-wise packing*.

**3.X** **projected picture**: picture that uses a *projection format* for *omnidirectional video*.

**3.X** **projected region**: region in a *projected picture* that is mapped to a *packed region* according to a *region-wise packing*.

**3.X** **projection**: specified correspondence between the colour samples of a *projected picture* and azimuth and elevation positions on a sphere.

**3.X** **region-wise packed picture**: decoded picture that contains one or more *packed regions*.

NOTE 1 to Entry – A packed picture may contain a *region-wise packing* of a *projected picture*.

**3.X** **region-wise packing**: transformation, resizing, and relocation of *packed regions* of a *region-wise packed picture* to remap the *packed regions* to *projected region*s of a *projected picture*.

**3.X** **sphere coordinates**: azimuth and elevation angles identifying a location of a point on a sphere.

**3.X** **sphere region**: region on a sphere, specified either by four *great circles* or by two *azimuth circles* and two *elevation circles*, or such a region on a rotated sphere after applying yaw, pitch, and roll rotations.

**3.X** **tilt angle**: angle indicating the amount of tilt of a *sphere region*, measured as the amount of rotation of a *sphere region* along the axis originating from the sphere origin passing through the centre point of the *sphere region*, where the angle value increases clockwise when looking from the origin towards the positive end of the axis.

**3.X** **viewport**: region of *omnidirectional video* content suitable for display and viewing by the user.

*In clause 4, add the following to the list of abbreviations (in alphabetical order):*

MCTS Motion-Constrained Tile Set

*In 5.8, add the following function definitions:*

Asin( x ) the trigonometric inverse sine function, operating on an argument x that is  
in the range of −1.0 to 1.0, inclusive, with an output value in the range of  
−π÷2 to π÷2, inclusive, in units of radians (5‑2)

Atan( x ) the trigonometric inverse tangent function, operating on an argument x, with  
an output value in the range of −π÷2 to π÷2, inclusive, in units of radians (5‑3)

Atan2( y, x ) =  (5‑4)

*Renumber the prior formulae 5-2 through 5-15 as 5-5 to 5-18 to account for the added formulae.*

*In 5.8, add the following function definitions:*

Sin( x ) the trigonometric sine function operating on an argument x in units of radians (5‑19)

*Renumber the prior formulae 5-16 and 5-17 as 5-20 and 5-21 to account for the added formulae.*

*In 5.8, add the following function definition:*

Tan( x ) the trigonometric tangent function operating on an argument x in units of radians (5‑22)

*In NOTE 2 of 7.4.2.4.4, delete the sentence that says “*Consequently, hypothetical reference decoder (HRD) parameters carried in non-nested buffering period, picture timing and decoding unit information SEI messages apply to access units based on such access unit boundary detection.”

*In 7.4.4, replace the following paragraph:*

**general\_non\_packed\_constraint\_flag** equal to 1 specifies that there are neither frame packing arrangement SEI messages nor segmented rectangular frame packing arrangement SEI messages present in the CVS. general\_non\_packed\_constraint\_flag equal to 0 indicates that there may or may not be one or more frame packing arrangement SEI messages or segmented rectangular frame packing arrangement SEI messages present in the CVS.

*with the following:*

**general\_non\_packed\_constraint\_flag** equal to 1 specifies that there are no frame packing arrangement SEI messages, segmented rectangular frame packing arrangement SEI messages, equirectangular projection SEI messages, or cubemap projection SEI messages present in the CVS. general\_non\_packed\_constraint\_flag equal to 0 indicates that there may or may not be one or more frame packing arrangement SEI messages, segmented rectangular frame packing arrangement SEI messages, equirectangular projection SEI messages, or cubemap projection SEI messages present in the CVS.

*In 8.7.2.1, replace the following paragraph:*

The deblocking filter process is applied to all prediction block edges and transform block edges of a picture, except the edges that are at the boundary of the picture, for which the deblocking filter process is disabled by slice\_deblocking\_filter\_disabled\_flag, that coincide with tile boundaries when loop\_filter\_across\_tiles\_enabled\_flag is equal to 0, or that coincide with upper or left slice boundaries of slices with slice\_loop\_filter\_across\_slices\_enabled\_flag equal to 0. For the transform units and prediction units with block edges less than 8 samples in either the vertical or horizontal direction, only the edges lying on the 8x8 sample grid of the considered component are filtered.

*with the following:*

The deblocking filter process is applied to all prediction block edges and transform block edges of a picture, except the following types of edges:

– Edges that are at the boundary of the picture,

– Edges that coincide with tile boundaries when loop\_filter\_across\_tiles\_enabled\_flag is equal to 0,

– Edges that coincide with upper or left boundaries of slices with slice\_loop\_filter\_across\_slices\_enabled\_flag, equal to 0 or slice\_deblocking\_filter\_disabled\_flag equal to 1,

– Edges within slices with slice\_deblocking\_filter\_disabled\_flag equal to 1,

– Edges that do not correspond to 8x8 sample grid boundaries of the considered component,

– Edges within chroma components for which both sides of the edge use inter prediction,

– Edges of chroma transform blocks that are not edges of the associated transform unit.

*In C.1, immediately before the sentence that says “*Figure C.1 shows the types of bitstream conformance points checked by the HRD.”, *add the following NOTE 1, and renumber the existing NOTEs in the clause accordingly:*

NOTE 1 – Decoders conforming to profiles specified in Annex A do not use NAL units with nuh\_layer\_id greater than 0 (e.g., access unit delimiter NAL units with nuh\_layer\_id greater than 0) for access unit boundary detection, except for identification of whether a NAL unit is a VCL or non-VCL NAL unit. Consequently, hypothetical reference decoder (HRD) parameters carried in non-scalable-nested buffering period, picture timing and decoding unit information SEI messages apply to access units that are identified based on such access unit boundary detection.

*Replace D.2.1 with the following:*

**D.2.1 General SEI message syntax**

|  |  |
| --- | --- |
| sei\_payload( payloadType, payloadSize ) { | **Descriptor** |
| if( nal\_unit\_type  = =  PREFIX\_SEI\_NUT ) |  |
| if( payloadType  = =  0 ) |  |
| buffering\_period( payloadSize ) |  |
| else if( payloadType  = =  1 ) |  |
| pic\_timing( payloadSize ) |  |
| else if( payloadType  = =  2 ) |  |
| pan\_scan\_rect( payloadSize ) |  |
| else if( payloadType  = =  3 ) |  |
| filler\_payload( payloadSize ) |  |
| else if( payloadType  = =  4 ) |  |
| user\_data\_registered\_itu\_t\_t35( payloadSize ) |  |
| else if( payloadType  = =  5 ) |  |
| user\_data\_unregistered( payloadSize ) |  |
| else if( payloadType  = =  6 ) |  |
| recovery\_point( payloadSize ) |  |
| else if( payloadType  = =  9 ) |  |
| scene\_info( payloadSize ) |  |
| else if( payloadType  = =  15 ) |  |
| picture\_snapshot( payloadSize ) |  |
| else if( payloadType  = =  16 ) |  |
| progressive\_refinement\_segment\_start( payloadSize ) |  |
| else if( payloadType  = =  17 ) |  |
| progressive\_refinement\_segment\_end( payloadSize ) |  |
| else if( payloadType  = =  19 ) |  |
| film\_grain\_characteristics( payloadSize ) |  |
| else if( payloadType  = =  22 ) |  |
| post\_filter\_hint( payloadSize ) |  |
| else if( payloadType  = =  23 ) |  |
| tone\_mapping\_info( payloadSize ) |  |
| else if( payloadType  = =  45 ) |  |
| frame\_packing\_arrangement( payloadSize ) |  |
| else if( payloadType  = =  47 ) |  |
| display\_orientation( payloadSize ) |  |
| else if( payloadType  = =  56 ) |  |
| green\_metadata( payloadsize ) /\* specified in ISO/IEC 23001-11 \*/ |  |
| else if( payloadType  = =  128 ) |  |
| structure\_of\_pictures\_info( payloadSize ) |  |
| else if( payloadType  = =  129 ) |  |
| active\_parameter\_sets( payloadSize ) |  |
| else if( payloadType  = =  130 ) |  |
| decoding\_unit\_info( payloadSize ) |  |
| else if( payloadType  = =  131 ) |  |
| temporal\_sub\_layer\_zero\_index( payloadSize ) |  |
| else if( payloadType  = =  133 ) |  |
| scalable\_nesting( payloadSize ) |  |
| else if( payloadType  = =  134 ) |  |
| region\_refresh\_info( payloadSize ) |  |
| else if( payloadType  = =  135 ) |  |
| no\_display( payloadSize ) |  |
| else if( payloadType  = =  136 ) |  |
| time\_code( payloadSize ) |  |
| else if( payloadType  = =  137 ) |  |
| mastering\_display\_colour\_volume( payloadSize ) |  |
| else if( payloadType  = =  138 ) |  |
| segmented\_rect\_frame\_packing\_arrangement( payloadSize ) |  |
| else if( payloadType  = =  139 ) |  |
| temporal\_motion\_constrained\_tile\_sets( payloadSize ) |  |
| else if( payloadType  = =  140 ) |  |
| chroma\_resampling\_filter\_hint( payloadSize ) |  |
| else if( payloadType  = =  141 ) |  |
| knee\_function\_info( payloadSize ) |  |
| else if( payloadType  = =  142 ) |  |
| colour\_remapping\_info( payloadSize ) |  |
| else if( payloadType  = =  143 ) |  |
| deinterlaced\_field\_identification( payloadSize ) |  |
| else if( payloadType  = =  144 ) |  |
| content\_light\_level\_info( payloadSize ) |  |
| else if( payloadType  = =  145 ) |  |
| dependent\_rap\_indication( payloadSize ) |  |
| else if( payloadType  = =  146 ) |  |
| coded\_region\_completion( payloadSize ) |  |
| else if( payloadType  = =  147 ) |  |
| alternative\_transfer\_characteristics( payloadSize ) |  |
| else if( payloadType  = =  148 ) |  |
| ambient\_viewing\_environment( payloadSize ) |  |
| else if( payloadType  = =  149 ) |  |
| content\_colour\_volume( payloadSize ) |  |
| else if( payloadType  = =  150 ) |  |
| equirectangular\_projection( payloadSize ) |  |
| else if( payloadType  = =  151 ) |  |
| cubemap\_projection( payloadSize ) |  |
| else if( payloadType  = =  152 ) |  |
| regionwise\_packing( payloadSize ) |  |
| else if( payloadType  = =  153 ) |  |
| omni\_viewport( payloadSize ) |  |
| else if( payloadType  = =  157 ) |  |
| regional\_nesting( payloadSize ) |  |
| else if( payloadType  = =  158 ) |  |
| mcts\_extraction\_info\_sets( payloadSize ) |  |
| else if( payloadType  = =  159 ) |  |
| mcts\_extraction\_info\_nesting( payloadSize ) |  |
| else if( payloadType  = =  160 ) |  |
| layers\_not\_present( payloadSize ) /\* specified in Annex F \*/ |  |
| else if( payloadType  = =  161 ) |  |
| inter\_layer\_constrained\_tile\_sets( payloadSize ) /\* specified in Annex F \*/ |  |
| else if( payloadType  = =  162 ) |  |
| bsp\_nesting( payloadSize ) /\* specified in Annex F \*/ |  |
| else if( payloadType  = =  163 ) |  |
| bsp\_initial\_arrival\_time( payloadSize ) /\* specified in Annex F \*/ |  |
| else if( payloadType  = =  164 ) |  |
| sub\_bitstream\_property( payloadSize ) /\* specified in Annex F \*/ |  |
| else if( payloadType  = =  165 ) |  |
| alpha\_channel\_info( payloadSize ) /\* specified in Annex F \*/ |  |
| else if( payloadType  = =  166 ) |  |
| overlay\_info( payloadSize ) /\* specified in Annex F \*/ |  |
| else if( payloadType  = =  167 ) |  |
| temporal\_mv\_prediction\_constraints( payloadSize ) /\* specified in Annex F \*/ |  |
| else if( payloadType  = =  168 ) |  |
| frame\_field\_info( payloadSize ) /\* specified in Annex F \*/ |  |
| else if( payloadType  = =  176 ) |  |
| three\_dimensional\_reference\_displays\_info( payloadSize ) /\* specified in Annex G \*/ |  |
| else if( payloadType  = =  177 ) |  |
| depth\_representation\_info( payloadSize ) /\* specified in Annex G \*/ |  |
| else if( payloadType  = =  178 ) |  |
| multiview\_scene\_info( payloadSize ) /\* specified in Annex G \*/ |  |
| else if( payloadType  = =  179 ) |  |
| multiview\_acquisition\_info( payloadSize ) /\* specified in Annex G \*/ |  |
| else if( payloadType  = =  180 ) |  |
| multiview\_view\_position( payloadSize ) /\* specified in Annex G \*/ |  |
| else if( payloadType  = =  181 ) |  |
| alternative\_depth\_info( payloadSize ) /\* specified in Annex I \*/ |  |
| else |  |
| reserved\_sei\_message( payloadSize ) |  |
| else /\* nal\_unit\_type  = =  SUFFIX\_SEI\_NUT \*/ |  |
| if( payloadType  = =  3 ) |  |
| filler\_payload( payloadSize ) |  |
| else if( payloadType  = =  4 ) |  |
| user\_data\_registered\_itu\_t\_t35( payloadSize ) |  |
| else if( payloadType  = =  5 ) |  |
| user\_data\_unregistered( payloadSize ) |  |
| else if( payloadType  = =  17 ) |  |
| progressive\_refinement\_segment\_end( payloadSize ) |  |
| else if( payloadType  = =  22 ) |  |
| post\_filter\_hint( payloadSize ) |  |
| else if( payloadType  = =  132 ) |  |
| decoded\_picture\_hash( payloadSize ) |  |
| else if( payloadType  = =  146 ) |  |
| coded\_region\_completion( payloadSize ) |  |
| else |  |
| reserved\_sei\_message( payloadSize ) |  |
| if( more\_data\_in\_payload( ) ) { |  |
| if( payload\_extension\_present( ) ) |  |
| **reserved\_payload\_extension\_data** | u(v) |
| **payload\_bit\_equal\_to\_one** /\* equal to 1 \*/ | f(1) |
| while( !byte\_aligned( ) ) |  |
| **payload\_bit\_equal\_to\_zero** /\* equal to 0 \*/ | f(1) |
| } |  |
| } |  |

*Renumber clause D.2.40 (Reserved SEI message syntax) as D.2.46.*

*Add clauses D.2.40 through D.2.44, as follows:*

**D.2.40 Content colour volume SEI message syntax**

|  |  |
| --- | --- |
| content\_colour\_volume( payloadSize ) { | **Descriptor** |
| **ccv\_cancel\_flag** | u(1) |
| if( !ccv\_cancel\_flag ) { |  |
| **ccv\_persistence\_flag** | u(1) |
| **ccv\_primaries\_present\_flag** | u(1) |
| **ccv\_min\_luminance\_value\_present\_flag** | u(1) |
| **ccv\_max\_luminance\_value\_present\_flag** | u(1) |
| **ccv\_avg\_luminance\_value\_present\_flag** | u(1) |
| **ccv\_reserved\_zero\_2bits** | u(2) |
| if( ccv\_primaries\_present\_flag ) |  |
| for( c = 0; c < 3; c++ ) { |  |
| **ccv\_primaries\_x**[ c ] | i(16) |
| **ccv\_primaries\_y**[ c ] | i(16) |
| } |  |
| if( ccv\_min\_luminance\_value\_present\_flag ) |  |
| **ccv\_min\_luminance\_value** | u(32) |
| if( ccv\_max\_luminance\_value\_present\_flag ) |  |
| **ccv\_max\_luminance\_value** | u(32) |
| if( ccv\_avg\_luminance\_value\_present\_flag ) |  |
| **ccv\_avg\_luminance\_value** | u(32) |
| } |  |
| } |  |

**D.2.41** **Syntax of omnidirectional video specific SEI messages**

**D.2.41.1 Equirectangular projection SEI message syntax**

|  |  |
| --- | --- |
| equirectangular\_projection( payloadSize ) { | **Descriptor** |
| **erp\_cancel\_flag** | u(1) |
| if( !erp\_cancel\_flag ) { |  |
| **erp\_persistence\_flag** | u(1) |
| **erp\_rotation\_flag** | u(1) |
| **erp\_explicit\_coverage\_range\_flag** | u(1) |
| **erp**\_**reserved\_zero\_4bits** | u(4) |
| if( erp\_rotation\_flag  = =  1 ) { |  |
| **erp\_yaw\_rotation** | i(32) |
| **erp\_pitch\_rotation** | i(32) |
| **erp\_roll\_rotation** | i(32) |
| } |  |
| if( erp\_explicit\_coverage\_range\_flag  = =  1 ) { |  |
| **erp\_azimuth\_min** | i(32) |
| **erp\_azimuth\_max** | i(32) |
| **erp\_elevation\_min** | i(32) |
| **erp\_elevation\_max** | i(32) |
| } |  |
| } |  |
| } |  |

**D.2.41.2 Cubemap projection SEI message syntax**

|  |  |
| --- | --- |
| cubemap\_projection( payloadSize ) { | **Descriptor** |
| **cmp\_cancel\_flag** | u(1) |
| if( !cmp\_cancel\_flag ) { |  |
| **cmp\_persistence\_flag** | u(1) |
| **cmp\_rotation\_flag** | u(1) |
| **cmp\_padding\_flag** | u(1) |
| **cmp\_reserved\_zero\_4bits** | u(4) |
| if( cmp\_padding\_flag  = =  1 ) { |  |
| **cmp\_padding\_type** | u(2) |
| **cmp\_reserved\_zero\_6bits** | u(6) |
| **cmp\_padding\_chroma\_sample\_range\_minus1** | u(8) |
| } |  |
| if( cmp\_rotation\_flag  = =  1 ) { |  |
| **cmp\_yaw\_rotation** | i(32) |
| **cmp\_pitch\_rotation** | i(32) |
| **cmp\_roll\_rotation** | i(32) |
| } |  |
| } |  |
| } |  |

**D.2.41.3 Region-wise packing SEI message syntax**

|  |  |
| --- | --- |
| regionwise\_packing( payloadSize ) { | **Descriptor** |
| **rwp\_cancel\_flag** | u(1) |
| if( !rwp\_cancel\_flag ) { |  |
| **rwp\_persistence\_flag** | u(1) |
| **rwp\_reserved\_zero\_6bits** | u(6) |
| **num\_packed\_regions** | u(8) |
| **proj\_picture\_width** | u(16) |
| **proj\_picture\_height** | u(16) |
| for( i = 0; i < num\_packed\_regions; i++ ) { |  |
| **rwp\_reserved\_zero\_4bits**[ i ] | u(4) |
| **packing\_type**[ i ] | u(4) |
| if( packing\_type[ i ]  = =  0 ) { |  |
| **proj\_region\_width**[ i ] | u(16) |
| **proj\_region\_height**[ i ] | u(16) |
| **proj\_region\_top**[ i ] | u(16) |
| **proj\_region\_left**[ i ] | u(16) |
| **transform\_type**[ i ] | u(3) |
| **rwp\_reserved\_zero\_5bits**[ i ] | u(5) |
| **packed\_region\_width**[ i ] | u(16) |
| **packed\_region\_height**[ i ] | u(16) |
| **packed\_region\_top**[ i ] | u(16) |
| **packed\_region\_left**[ i ] | u(16) |
| } |  |
| } |  |
| } |  |
| } |  |

**D.2.41.4 Omnidirectional viewport SEI message syntax**

|  |  |
| --- | --- |
| omni\_viewport( payloadSize ) { | **Descriptor** |
| **omni\_viewport\_id** | u(10) |
| **omni\_viewport\_cancel\_flag** | u(1) |
| if( !omni\_viewport\_cancel\_flag ) { |  |
| **omni\_viewport\_persistence\_flag** | u(1) |
| **omni\_viewport\_cnt\_minus1** | u(4) |
| for( i = 0; i  <=  omni\_viewport\_cnt\_minus1; i++ ) { |  |
| **omni\_viewport\_azimuth\_centre**[ i ] | i(32) |
| **omni\_viewport\_elevation\_centre**[ i ] | i(32) |
| **omni\_viewport\_tilt\_centre**[ i ] | i(32) |
| **omni\_viewport\_hor\_range**[ i ] | u(32) |
| **omni\_viewport\_ver\_range**[ i ] | u(32) |
| } |  |
| } |  |
| } |  |

**D.2.42 Regional nesting SEI message syntax**

|  |  |
| --- | --- |
| regional\_nesting( payloadSize ) { | **Descriptor** |
| **regional\_nesting\_id** | u(16) |
| **regional\_nesting\_num\_rect\_regions** | u(8) |
| for( i = 0; i < regional\_nesting\_num\_rect\_regions; i++ ) { |  |
| **regional\_nesting\_rect\_region\_id**[ i ] | u(8) |
| **regional\_nesting\_rect\_left\_offset**[ i ] | u(16) |
| **regional\_nesting\_rect\_right\_offset**[ i ] | u(16) |
| **regional\_nesting\_rect\_top\_offset**[ i ] | u(16) |
| **regional\_nesting\_rect\_bottom\_offset**[ i ] | u(16) |
| } |  |
| **num\_sei\_messages\_in\_regional\_nesting\_minus1** | u(8) |
| for( i = 0; i  <=  num\_sei\_messages\_in\_regional\_nesting\_minus1; i++ ) { |  |
| **num\_regions\_for\_sei\_message**[ i ] | u(8) |
| for(j = 0; j < num\_regions\_for\_sei\_message[ i ]; j++ ) |  |
| **regional\_nesting\_sei\_region\_idx**[ i ][ j ] | u(8) |
| sei\_message( ) |  |
| } |  |
| } |  |

**D.2.43 Motion-constrained tile sets extraction information sets SEI message syntax**

|  |  |
| --- | --- |
| mcts\_extraction\_info\_sets( ) { | **Descriptor** |
| **num\_info\_sets\_minus1** | ue(v) |
| for( i = 0; i  <=  num\_info\_sets\_minus1; i++ ) { |  |
| **num\_mcts\_sets\_minus1**[ i ] | ue(v) |
| for( j = 0; j  <=  num\_mcts\_sets\_minus1[ i ]; j++ ) { |  |
| **num\_mcts\_in\_set\_minus1**[ i ][ j ] | ue(v) |
| for( k = 0; k  <=  num\_mcts\_in\_set\_minus1[ i ][ j ]; k++ ) |  |
| **idx\_of\_mcts\_in\_set**[ i ][ j ][ k ] | ue(v) |
| } |  |
| **slice\_reordering\_enabled\_flag**[ i ] | u(1) |
| if( slice\_reordering\_enabled\_flag[ i ] ) { |  |
| **num\_slice\_segments\_minus1**[ i ] | ue(v) |
| for( j = 0; j  <=  num\_slice\_segments\_minus1[ i ]; j++ ) |  |
| **output\_slice\_segment\_address**[ i ][ j ] | u(v) |
| } |  |
| **num\_vps\_in\_info\_set\_minus1**[ i ] | ue(v) |
| for( j = 0; j  <=  num\_vps\_in\_info\_set\_minus1[ i ]; j++ ) |  |
| **vps\_rbsp\_data\_length**[ i ][ j ] | ue(v) |
| **num\_sps\_in\_info\_set\_minus1**[ i ] | ue(v) |
| for( j = 0; j  <=  num\_sps\_in\_info\_set\_minus1[ i ]; j++ ) |  |
| **sps\_rbsp\_data\_length**[ i ][ j ] | ue(v) |
| **num\_pps\_in\_info\_set\_minus1**[ i ] | ue(v) |
| for( j = 0; j  <=  num\_pps\_in\_info\_set\_minus1[ i ]; j++ ) { |  |
| **pps\_nuh\_temporal\_id\_plus1**[ i ][ j ] | u(3) |
| **pps\_rbsp\_data\_length**[ i ][ j ] | ue(v) |
| } |  |
| while( !byte\_aligned( ) ) |  |
| **mcts\_alignment\_bit\_equal\_to\_zero** | f(1) |
| for( j = 0; j  <=  num\_vps\_in\_info\_set\_minus1[ i ]; j++ ) |  |
| for( k = 0; k  <=  vps\_rbsp\_data\_length[ i ][ j ]; k++ ) |  |
| **vps\_rbsp\_data\_byte**[ i ][ j ][ k ] | u(8) |
| for( j = 0; j  <=  num\_sps\_in\_info\_set\_minus1[ i ]; j++ ) |  |
| for( k = 0; k  <=  sps\_rbsp\_data\_length[ i ][ j ]; k++ ) |  |
| **sps\_rbsp\_data\_byte**[ i ][ j ][ k ] | u(8) |
| for( j = 0; j  <=  num\_pps\_in\_info\_set\_minus1[ i ]; j++ ) |  |
| for( k = 0; k  <=  pps\_rbsp\_data\_length[ i ][ j ]; k++ ) |  |
| **pps\_rbsp\_data\_byte**[ i ][ j ][ k ] | u(8) |
| } |  |
| } |  |

**D.2.44 Motion-constrained tile sets extraction information nesting SEI message syntax**

|  |  |
| --- | --- |
| mcts\_extraction\_info\_nesting( ) { | **Descriptor** |
| **all\_mcts\_flag** | u(1) |
| if( !all\_mcts\_flag ) { |  |
| **num\_associated\_mcts\_minus1** | ue(v) |
| for( i = 0; i  <=  num\_associated\_mcts\_minus1; i++ ) |  |
| **idx\_of\_associated\_mcts**[ i ] | ue(v) |
| } |  |
| **num\_sei\_messages\_in\_mcts\_extraction\_nesting\_minus1** | ue(v) |
| while( !byte\_aligned( ) ) |  |
| **mcts\_nesting\_zero\_bit** /\* equal to 0 \*/ | u(1) |
| for( i = 0; i  <=  num\_sei\_messages\_in\_mcts\_extraction\_nesting\_minus1; i++ ) |  |
| sei\_message( ) |  |
| } |  |

*In D.3.1, replace the following paragraphs:*

The list SingleLayerSeiList is set to consist of the payloadType values 3, 6, 9, 15, 16, 17, 19, 22, 23, 45, 47, 56, 128, 129, 131, 132, and 134 to 148, inclusive.

The list VclAssociatedSeiList is set to consist of the payloadType values 2, 3, 6, 9, 15, 16, 17, 19, 22, 23, 45, 47, 56, 128, 131, 132, and 134 to 148, inclusive.

The list PicUnitRepConSeiList is set to consist of the payloadType values 0, 1, 2, 6, 9, 15, 16, 17, 19, 22, 23, 45, 47, 56, 128, 129, 131, 132, 133, and 135 to 148, inclusive.

*with the following:*

The list SingleLayerSeiList is set to consist of the payloadType values 3, 6, 9, 15, 16, 17, 19, 22, 23, 45, 47, 56, 128, 129, 131, 132, 134 to 153, inclusive, and 157 to 159, inclusive.

The list VclAssociatedSeiList is set to consist of the payloadType values 2, 3, 6, 9, 15, 16, 17, 19, 22, 23, 45, 47, 56, 128, 131, 132, and 134 to 153, inclusive, and 157 to 159, inclusive.

The list PicUnitRepConSeiList is set to consist of the payloadType values 0, 1, 2, 6, 9, 15, 16, 17, 19, 22, 23, 45, 47, 56, 128, 129, 131, 132, 133, and 135 to 153, inclusive, and 157 to 159, inclusive.

*In D.3.1, in Table D.1, replace the corresponding rows of the table with the following:*

|  |  |
| --- | --- |
| Scene information | The access unit containing the SEI message and up to but not including the next access unit, in decoding order, that contains a scene information SEI message |
| Green metadata | The CLVS containing the SEI message |
| Temporal motion-constrained tile sets | The CLVS containing the SEI message |

*with the following:*

|  |  |
| --- | --- |
| Scene information | The access unit containing the SEI message and up to but not including the next access unit, in decoding order, that contains a scene information SEI message or starts a new CLVS |
| Green metadata | Specified by the syntax of the SEI message |
| Temporal motion-constrained tile sets | The access unit containing the SEI message and up to but not including the next access unit, in decoding order, that contains an SEI message of the same type or starts a new CLVS |

*In D.3.1, in Table D.1, append the following rows to the end of the table:*

|  |  |
| --- | --- |
| Content colour volume | Specified by the syntax of the SEI message |
| Omnidirectional projection information | Specified by the syntax of the SEI message |
| Omnidirectional viewport | Specified by the syntax of the SEI message |
| Regional nesting | Depending on the region-nested SEI messages; each region-nested SEI message has the same persistence scope as if the SEI message was non-region-nested |
| Motion-constrained tile sets extraction information sets | The access unit containing the SEI message and up to but not including the next access unit, in decoding order, that contains an SEI message of the same type or starts a new CLVS |
| Motion-constrained tile sets extraction information nesting | The access unit containing the SEI message |

*In D.3.1, immediately after Table D.1, add the following paragraph:*

The values of some SEI message syntax elements, including pan\_scan\_rect\_id, scene\_id, second\_scene\_id, snapshot\_id, progressive\_refinement\_id, tone\_map\_id, frame\_packing\_arrangement\_id, mcts\_id[ i ], knee\_function\_id, colour\_remap\_id, ilcts\_id[ i ], and regional\_nesting\_id, are split into two sets of value ranges, where the first set is specified as "may be used as determined by the application", and the second set is specified as "reserved for future use by ITU-T | ISO/IEC". Applications should be cautious of potential “collisions” of the interpretation for values of these syntax elements belonging to the first set of value ranges. Since different applications might use these IDs having values in the first set of value ranges for different purposes, particular care should be exercised in the design of encoders that generate SEI messages with these IDs having values in the first set of value ranges, and in the design of decoders that interpret SEI messages with these IDs. This Specification does not define any management for these values. These IDs having values in the first set of value ranges might only be suitable for use in contexts in which "collisions" of usage (i.e., different definitions of the syntax and semantics of an SEI message with one of these IDs having the same value in the first set of value ranges) are unimportant, or not possible, or are managed – e.g., defined or managed in the controlling application or transport specification, or by controlling the environment in which bitstreams are distributed.

*In D.3.4 (Pan-scan rectangle SEI message semantics), replace the following:*

Values of pan\_scan\_rect\_id from 0 to 255 and from 512 to 231 − 1 may be used as determined by the application. Values of pan\_scan\_rect\_id from 256 to 511 and from 231 to 232 − 2 are reserved for future use by ITU-T | ISO/IEC. Decoders encountering a value of pan\_scan\_rect\_id in the range of 256 to 511, inclusive, or in the range of 231 to 232 − 2, inclusive, shall ignore it.

*with the following:*

Values of pan\_scan\_rect\_id from 0 to 255, inclusive, and from 512 to 231 − 1, inclusive, may be used as determined by the application. Values of pan\_scan\_rect\_id from 256 to 511, inclusive, and from 231 to 232 − 2, inclusive, are reserved for future use by ITU-T | ISO/IEC. Decoders encountering a value of pan\_scan\_rect\_id in the range of 256 to 511, inclusive, or in the range of 231 to 232 − 2, inclusive, shall ignore it.

*In D.3.15 (Tone mapping information SEI message semantics), replace the following paragraph:*

Values of tone\_map\_id from 0 to 255 and from 512 to 231 − 1 may be used as determined by the application. Values of tone\_map\_id from 256 to 511, inclusive, and from 231 to 232 − 2, inclusive, are reserved for future use by ITU-T | ISO/IEC. Decoders shall ignore all tone mapping information SEI messages containing a value of tone\_map\_id in the range of 256 to 511, inclusive, or in the range of 231 to 232 − 2, inclusive, and bitstreams shall not contain such values.

*with the following:*

Values of tone\_map\_id from 0 to 255, inclusive, and from 512 to 231 − 1, inclusive, may be used as determined by the application. Values of tone\_map\_id from 256 to 511, inclusive, and from 231 to 232 − 2, inclusive, are reserved for future use by ITU-T | ISO/IEC. Decoders encountering a value of tone\_map\_id in the range of 256 to 511, inclusive, or in the range of 231 to 232 − 2, inclusive, shall ignore it.

*In D.3.16 (Frame packing arrangement SEI message semantics), replace the following paragraph:*

Values of frame\_packing\_arrangement\_id from 0 to 255 and from 512 to 231 − 1 may be used as determined by the application. Values of frame\_packing\_arrangement\_id from 256 to 511 and from 231 to 232 − 2 are reserved for future use by ITU-T | ISO/IEC. Decoders shall ignore all frame packing arrangement SEI messages containing a value of frame\_packing\_arrangement\_id in the range of 256 to 511, inclusive, or in the range of 231 to 232 − 2, inclusive, and bitstreams shall not contain such values.

*with the following:*

Values of frame\_packing\_arrangement\_id from 0 to 255, inclusive, and from 512 to 231 − 1, inclusive, may be used as determined by the application. Values of frame\_packing\_arrangement\_id from 256 to 511, inclusive, and from 231 to 232 − 2, inclusive, are reserved for future use by ITU-T | ISO/IEC. Decoders encountering a value of frame\_packing\_arrangement\_id in the range of 256 to 511, inclusive, or in the range of 231 to 232 − 2, inclusive, shall ignore it.

*In D.3.24, replace the following:*

It is a requirement of bitstream conformance that the following restrictions apply on nesting of SEI messages:

*with the following:*

It is a requirement of bitstream conformance that the following restrictions apply on the containing of SEI messages in a scalable nesting SEI message:

*In D.3.30* *(Temporal motion-constrained tile sets SEI message semantics), replace the following paragraph:*

The temporal motion-constrained tile sets SEI message indicates that the inter prediction process is constrained such that no sample value outside each identified tile set, and no sample value at a fractional sample position that is derived using one or more sample values outside the identified tile set, is used for inter prediction of any sample within the identified tile set.

*with the following:*

The temporal motion-constrained tile sets SEI message indicates that the following constraints apply:

– No sample values outside each identified tile set or outside the picture are referenced for inter prediction.

– For PUs located directly left of the right tile boundary of each identified tile set except the last one at the bottom right, the following applies when CuPredMode[ xPb ][ yPb ] is equal to MODE\_INTER, where ( xPb, yPb ) specifies the top-left sample of the corresponding luma prediction block relative to the top-left sample of the current picture:

– With the number of spatial merging candidates numSpatialMergeCand derived as follows:

numSpatialMergeCand = availableFlagA0 + availableFlagA1 +  (D‑42)  
 availableFlagB0 + availableFlagB1 + availableFlagB2

where availableFlagA0, availableFlagA1, availableFlagB0, availableFlagB1, and availableFlagB2 are the output of the derivation process for spatial merging candidates specified in clause 8.5.3.2.3, the following applies:

– If numSpatialMergeCand is equal to 0, merge\_flag[ xPb ][ yPb ] is equal to 0.

– Otherwise (numSpatialMergeCand is greater than 0), merge\_idx[ xPb ][ yPb ] is in the range of 0 to numSpatialMergeCand − 1, inclusive.

– With the number of spatial motion vector predictor candidates numSpatialMvpCand derived as follows:

if ( availableFlagLXA )  
 numSpatialMvpCand = availableFlagLXA + ( ( mvLXA  !=  mvLXB ) ? availableFlagLXB : 0 )  
else (D‑43)  
 numSpatialMvpCand = availableFlagLXB

where availableFlagLXA, availableFlagLXB, mvLXA, and mvLXB are the output of the derivation process for motion vector predictor candidates from neighbouring prediction unit partitions specified in clause 8.5.3.2.7, the following applies:

– If numSpatialMvpCand is equal to 0, mvp\_l0\_flag[ xPb ][ yPb ] and mvp\_l1\_flag[ xPb ][ yPb ] is equal to 1.

– Otherwise (numSpatialMvpCand is greater than 0), mvp\_l0\_flag[ xPb ][ yPb ] and mvp\_l1\_flag[ xPb ][ yPb ] is in the range of 0 to numSpatialMvpCand − 1, inclusive.

NOTE 1 – The first constraint restricts motion vectors to point to full-sample locations inside each identified tile set and to fractional-sample locations that require only full-sample locations inside each identified tile set for interpolation. The second constraint restricts the usage of motion vector candidates derived from blocks outside each identified tile set.

*In D.3.30 (Temporal motion-constrained tile sets SEI message semantics), renumber the subsequent NOTEs to account for the added NOTE.*

*In D.3.30 (Temporal motion-constrained tile sets SEI message semantics) and subsequent subclauses of Annex D, renumber the subsequent formulae to account for the added formulae.*

*In D.3.30 (Temporal motion-constrained tile sets SEI message semantics), remove the following paragraph:*

The number of temporal motion-constrained tile sets SEI messages applicable to the same nuh\_layer\_id value in each access unit shall not exceed 5.

*In D.3.30 (Temporal motion-constrained tile sets SEI message semantics), add the following paragraphs immediately before the semantics of mc\_all\_tiles\_exact\_sample\_value\_match\_flag:*

When a temporal motion-constrained tile sets SEI message is present, a slice segment that contains one or more tiles in any particular temporal motion-constrained tile set shall not be a dependent slice segment of an independent slice segment that contains one or more tiles that do not belong to that temporal motion-constrained tile set.

For purposes of referencing a particular temporal motion-constrained tile set that is identified in a temporal motion-constrained tile sets SEI message (e.g., for use with a motion-constrained tile sets extraction information sets SEI message or a motion-constrained tile sets extraction information nesting SEI message), an MCTS index is defined as follows:

– If the value of each\_tile\_one\_tile\_set\_flag of the temporal motion-constrained tile sets SEI message is equal to 0, the MCTS index is the value of the variable i within the loop of the num\_sets\_in\_message\_minus1 + 1 sets of MCTS information specified by the temporal MCTS SEI message.

– Otherwise, the MCTS index of each MCTS is the tile position of the tile in tile raster scan order.

*In D.3.30 (Temporal motion-constrained tile sets SEI message semantics), replace the following sentence:*

Decoders shall ignore (remove from the bitstream and discard) those SEI messages containing a value of mcts\_id[ i ] in the range of 256 to 511, inclusive, or in the range of 231 to 232 − 2, inclusive.

*with the following:*

Decoders encountering a value of mcts\_id[ i ] in the range of 256 to 511, inclusive, or in the range of 231 to 232 − 2, inclusive, shall ignore it.

*In D.3.30 (Temporal motion-constrained tile sets SEI message semantics), after the paragraph providing the semantics of top\_left\_tile\_index[ i ][ j ] and bottom\_right\_tile\_index[ i ][ j ], add the following sentence:*

The value of top\_left\_tile\_index[ i ][ j ] and bottom\_right\_tile\_index[ i ][ j ] shall be in the range of 0 to ( num\_tile\_columns\_minus1 + 1 ) \* ( num\_tile\_rows\_minus1 + 1 ) − 1, inclusive.

*In D.3.31 (Chroma resampling filter hint SEI message semantics), add the following paragraph before the semantics of hor\_chroma\_filter\_idc:*

All chroma resampling filter hint SEI messages that apply to the same CLVS shall have the same content.

*In D.3.32 (Knee function information SEI message semantics), replace the following:*

Decoders shall ignore all knee function information SEI messages containing a value of knee\_function\_id in the range of 256 to 511, inclusive, or in the range of 231 to 232 − 2, inclusive, and bitstreams shall not contain such values.

*with the following:*

Decoders encountering a value of knee\_function\_id in the range of 256 to 511, inclusive, or in the range of 231 to 232 − 2, inclusive, shall ignore it.

*In D.3.33 (Colour remapping information SEI message semantics), replace the following paragraph:*

Values of colour\_remap\_id from 0 to 255 and from 512 to 231 − 1 may be used as determined by the application. Values of colour\_remap\_id from 256 to 511, inclusive, and from 231 to 232 − 2, inclusive, are reserved for future use by ITU-T | ISO/IEC. Decoders shall ignore all colour remapping information SEI messages containing a value of colour\_remap\_id in the range of 256 to 511, inclusive, or in the range of 231 to 232 − 2, inclusive, and bitstreams shall not contain such values.

*with the following:*

Values of colour\_remap\_id from 0 to 255, inclusive, and from 512 to 231 − 1, inclusive, may be used as determined by the application. Values of colour\_remap\_id from 256 to 511, inclusive, and from 231 to 232 − 2, inclusive, are reserved for future use by ITU-T | ISO/IEC. Decoders encountering a value of colour\_remap\_id in the range of 256 to 511, inclusive, or in the range of 231 to 232 − 2, inclusive, shall ignore it.

*In D.3.35 (Content light level information SEI message semantics), add the following paragraph before the semantics of max\_content\_light\_level:*

All content light level information SEI messages that apply to the same CLVS shall have the same content.

*Renumber clause D.3.40 (Reserved SEI message semantics) as D.3.46.*

*Add subclauses D.3.40 through D.3.44, as follows:*

**D.3.40 Content colour volume SEI message semantics**

The content colour volume SEI message describes the colour volume characteristics of the associated pictures. These colour volume characteristics are expressed in terms of a nominal range, although deviations from this range may occur.

The variable transferCharacteristics is specified as follows:

– If an alternative transfer characteristics SEI message is present for the CLVS, transferCharacteristics is set equal to preferred\_transfer\_characteristics;

– Otherwise, (an alternative transfer characteristics SEI message is not present for the CLVS), transferCharacteristics is set equal to transfer\_characteristics.

The content colour volume SEI message shall not be present, and decoders shall ignore it, when any of the following conditions is true:

– Any of the values of transferCharacteristics, colour\_primaries, and matrix\_coeffs has a value defined as unspecified.

– The value of transfer\_characteristics is equal to 2, 4, or 5.

– The value of colour\_primaries is equal to 2.

The following applies when converting the signal from a non-linear to a linear representation:

– If the value of transferCharacteristics is equal to 1, 6, 7, 14, or 15, the Rec. ITU‑R BT.1886-0 reference electro-optical transfer function should be used to convert the signal to its linear representation, where the value of screen luminance for white is set equal to 100 cd/m2, the value of screen luminance for black is set equal to 0 cd/m2, and the value of the exponent of the power function is set equal to 2.4.

– Otherwise, if the value of transferCharacteristics is equal to 18, the hybrid log-gamma reference electro-optical transfer function specified in Rec. ITU-R BT.2100-0 should be used to convert the signal to its linear representation, where the value of nominal peak luminance of the display is set equal to 1000 cd/m2, the value of the display luminance for black is set equal to 0 cd/m2, and the value of system gamma is set equal to 1.2.

– Otherwise (the value of transferCharacteristics is not equal to 1, 6, 7, 14, 15, or 18) when the content colour volume SEI message is present, the exact inverse of the transfer function specified in Table E.4 should be used to convert the non-linear signal to a linear representation.

**ccv\_cancel\_flag** equal to 1 indicates that the content colour volume SEI message cancels the persistence of any previous content colour volume SEI message in output order that applies to the current layer. ccv\_cancel\_flagequal to 0 indicates that content colour volume information follows.

**ccv\_persistence\_flag** specifies the persistence of the content colour volume SEI message for the current layer.

ccv\_persistence\_flag equal to 0 specifies that the content colour volume applies to the current decoded picture only.

Let picA be the current picture. ccv\_persistence\_flag equal to 1 specifies that the content colour volume SEI message persists for the current layer in output order until any of the following conditions are true:

– A new CLVS of the current layer begins.

– The bitstream ends.

– A picture picB in the current layer in an access unit containing a content colour volume SEI message that is applicable to the current layer is output for which PicOrderCnt( picB ) is greater than PicOrderCnt( picA ), where PicOrderCnt( picB ) and PicOrderCnt( picA ) are the PicOrderCntVal values of picB and picA, respectively, immediately after the invocation of the decoding process for the picture order count of picB.

**ccv\_primaries\_present\_flag** equal to 1 specifies that the syntax elements ccv\_primaries\_x[ c ] and ccv\_primaries\_y[ c ] are present. ccv\_primaries\_present\_flag equal to 0 specifies that the syntax elements ccv\_primaries\_x[ c ] and ccv\_primaries\_y[ c ] are not present.

**ccv\_min\_luminance\_value\_present\_flag** equal to 1 specifies that the syntax element ccv\_min\_luminance\_value is present. ccv\_min\_luminance\_value\_present\_flag equal to 0 specifies that the syntax element ccv\_min\_luminance\_value is not present.

**ccv\_max\_luminance\_value\_present\_flag** equal to 1 specifies that the syntax element ccv\_max\_luminance\_value is present. ccv\_max\_luminance\_value\_present\_flag equal to 0 specifies that the syntax element ccv\_max\_luminance\_value is not present.

**ccv\_avg\_luminance\_value\_present\_flag** equal to 1 specifies that the syntax element ccv\_avg\_luminance\_value is present. ccv\_avg\_luminance\_value\_present\_flag equal to 0 specifies that the syntax element ccv\_avg\_luminance\_value is not present.

It is a requirement of bitstream conformance that the values of ccv\_primaries\_present\_flag, ccv\_min\_luminance\_value\_present\_flag, ccv\_max\_luminance\_value\_present\_flag, and ccv\_avg\_luminance\_value\_present\_flag shall not all be equal to 0.

**ccv\_reserved\_zero\_2bits**[ i ] shall be equal to 0 in bitstreams conforming to this version of this Specification. Other values for reserved\_zero\_2bits[ i ] are reserved for future use by ITU-T | ISO/IEC. Decoders shall ignore the value of reserved\_zero\_2bits[ i ].

**ccv\_primaries\_x**[ c ] and **ccv\_primaries\_y**[ c ] specify the normalized x and y chromaticity coordinates, respectively, of the colour primary component c of the nominal content colour volume in normalized increments of 0.00002, according to the CIE 1931 definition of x and y as specified in ISO 11664-1 (see also ISO 11664-3 and CIE 15). For describing colour volumes that use red, green, and blue colour primaries, it is suggested that index value c equal to 0 should correspond to the green primary, c equal to 1 should correspond to the blue primary, and c equal to 2 should correspond to the red colour primary (see also Annex E and Table E.3).

The values of ccv\_primaries\_x[ c ] and ccv\_primaries\_y[ c ] shall be in the range of −5 000 000 to 5 000 000, inclusive.

When ccv\_primaries\_x[ c ] and ccv\_primaries\_y[ c ] are not present, they are inferred to be equal to the normalized x and y chromaticity coordinates, respectively, specified by colour\_primaries.

**ccv\_min\_luminance\_value** specifies the normalized minimum luminance value, according to CIE 1931, that is expected to be present in the content, where values are normalized to Lo or Lc as specified in Table E.4 according to the indicated transfer characteristics of the signal. The values of ccv\_min\_luminance\_value are in normalized increments of 0.0000001.

**ccv\_max\_luminance\_value** specifies the maximum luminance value, according to CIE 1931, that is expected to be present in the content, where values are normalized to Lo or Lc as specified in Table E.4 according to the transfer characteristics of the signal. The values of ccv\_max\_luminance\_value are in normalized increments of 0.0000001.

**ccv\_avg\_luminance\_value** specifies the average luminance value, according to CIE 1931, that is expected to be present in the content, where values are normalized to Lo or Lc as specified in Table E.4 according to the transfer characteristics of the signal. The values of ccv\_avg\_luminance\_value are in normalized increments of 0.0000001.

NOTE – The resulting domain from this conversion process may or may not represent light in a source or display domain – it is merely a gamut representation domain rather than necessarily being a representation of actual light in either the scene or display domain. Therefore, the values corresponding to ccv\_min\_luminance\_value, ccv\_max\_luminance\_value, and ccv\_avg\_luminance\_value might not necessarily correspond to a true luminance value.

The value of ccv\_min\_luminance\_value, when present, shall be less than or equal to ccv\_avg\_luminance\_value, when present. The value of ccv\_avg\_luminance\_value, when present, shall be less than or equal to ccv\_max\_luminance\_value, when present. The value of ccv\_min\_luminance\_value, when present, shall be less than or equal to ccv\_max\_luminance\_value, when present.

When the visually relevant region does not correspond to the entire cropped decoded picture, such as for "letterbox" encoding of video content with a wide picture aspect ratio within a taller cropped decoded picture, the indicated ccv\_min\_luminance\_value, ccv\_max\_luminance\_value, and ccv\_avg\_luminance\_value should correspond only to values within the visually relevant region.

**D.3.41 Semantics of omnidirectional video SEI messages**

**D.3.41.1 Equirectangular projection SEI message semantics**

The equirectangular projection SEI message provides information to enable remapping (through an equirectangular projection) of the colour samples of the projected pictures onto a sphere coordinate space in sphere coordinates (ϕ, θ) for use in omnidirectional video applications for which the viewing perspective is from the origin looking outward toward the inside of the sphere. The sphere coordinates are defined so that ϕ is the azimuth (longitude, increasing eastward) and θ is the elevation (latitude, increasing northward).

Rotation angles yaw (α), pitch (β), and roll (γ) are also used in the specification of these semantics.

Relative to an (x, y, z) Cartesian coordinate system, yaw expresses a rotation around the z (vertical, up) axis, pitch rotates around the y (lateral, side-to-side) axis, and roll rotates around the x (back-to-front) axis. Rotations are extrinsic, i.e., around x, y, and z fixed reference axes. The angles increase clockwise when looking from the origin towards the positive end of an axis.

When an equirectangular projection SEI message is present for any picture of a CLVS of a particular layer, an equirectangular projection SEI message shall be present for the first picture of the CLVS and no SEI message indicating a different type of projection shall be present for any picture of the CLVS.

When general\_non\_packed\_constraint\_flag is equal to 1 in the active SPS for the current layer, there shall be no equirectangular projection SEI messages applicable for any picture of the CLVS of the current layer.

When a frame packing arrangement SEI message with frame\_packing\_arrangement\_cancel\_flag equal to 0 is present that applies to the picture, and the value of frame\_packing\_arrangement\_type of the frame packing arrangement SEI message is not equal to 3, 4, or 5, or the value of quincunx\_sampling\_flag of the frame packing arrangement SEI message is not equal to 0, an equirectangular projection SEI message with erp\_cancel\_flag equal to 0 shall not be present that applies to the picture. Decoders shall ignore equirectangular projection SEI messages when a frame packing arrangement SEI message with frame\_packing\_arrangement\_cancel\_flag equal to 0 is present that applies to the picture, and the value of frame\_packing\_arrangement\_type of the frame packing arrangement SEI message is not equal to 3, 4, or 5, or the value of quincunx\_sampling\_flag of the frame packing arrangement SEI message is not equal to 0.

When a segmented rectangular frame packing arrangement SEI message with segmented\_rect\_frame\_packing\_arrangement\_cancel\_flag equal to 0 is present that applies to the picture, an equirectangular projection SEI message with erp\_cancel\_flag equal to 0 shall not be present that applies to the picture. Decoders shall ignore equirectangular projection SEI messages when a segmented rectangular frame packing arrangement SEI message with segmented\_rect\_frame\_packing\_arrangement\_cancel\_flag equal to 0 is present that applies to the picture.

**erp\_cancel\_flag** equal to 1 indicates that the SEI message cancels the persistence of any previous equirectangular projection SEI message in output order. erp\_cancel\_flag equal to 0 indicates that equirectangular projection information follows.

**erp\_persistence\_flag** specifies the persistence of the equirectangular projection SEI message for the current layer.

erp\_persistence\_flag equal to 0 specifies that the equirectangular projection SEI message applies to the current decoded picture only.

Let picA be the current picture. erp\_persistence\_flag equal to 1 specifies that the equirectangular projection SEI message persists for the current layer in output order until one or more of the following conditions are true:

– A new CLVS of the current layer begins.

– The bitstream ends.

– A picture picB in the current layer in an access unit containing an equirectangular projection SEI message that is applicable to the current layer is output for which PicOrderCnt( picB ) is greater than PicOrderCnt( picA ), where PicOrderCnt( picB ) and PicOrderCnt( picA ) are the PicOrderCntVal values of picB and picA, respectively, immediately after the invocation of the decoding process for picture order count for picB.

**erp\_rotation\_flag** equal to 1 indicates that a rotation for conversion between the global and local coordinate systems applies. erp\_rotation\_flag equal to 0 indicates that no rotation is applied and the global and local coordinate systems are identical.

**erp\_explicit\_coverage\_range\_flag** indicates the presence of syntax elements that describe the coverage sphere region.

**erp**\_**reserved\_zero\_4bits**, when present, shall be equal to 0 in bitstreams conforming to this version of this Specification. Other values for erp\_reserved\_zero\_4bits are reserved for future use by ITU-T | ISO/IEC. Decoders shall ignore the value of erp\_reserved\_zero\_4bits.

**erp\_yaw\_rotation** specifies the value of the yaw rotation angle, in units of 2−16 degrees. The value of erp\_yaw\_rotation shall be in the range of −180 \* 216 (i.e., −11 796 480) to 180 \* 216 − 1 (i.e., 11 796 479), inclusive. When not present, the value of erp\_yaw\_rotation is inferred to be equal to 0.

**erp\_pitch\_rotation** specifies the value of the pitch rotation angle, in units of 2−16 degrees. The value of erp\_pitch\_rotation shall be in the range of −90 \* 216 (i.e., −5 898 240) to 90 \* 216 (i.e., 5 898 240), inclusive. When not present, the value of erp\_pitch\_rotation is inferred to be equal to 0.

**erp\_roll\_rotation** specifies the value of the roll rotation angle, in units of 2−16 degrees. The value of erp\_roll\_rotation shall be in the range of −180 \* 216 (i.e., −11 796 480) to 180 \* 216 − 1 (i.e., 11 796 479), inclusive. When not present, the value of erp\_roll\_rotation is inferred to be equal to 0.

**erp\_azimuth\_min** specifies the minimum azimuth value of the coverage sphere region, in units of 2−16 degrees. The value of erp\_azimuth\_min shall be in the range of −360 \* 216 (i.e., −23 592 960) to 360 \* 216−1 (i.e., 23 592 959), inclusive. When not present, the value of erp\_azimuth\_min is inferred to be equal to −180 \* 216 (i.e., 11 796 480).

**erp\_azimuth\_max** specifies the maximum azimuth value of the coverage sphere region, in units of 2−16 degrees. The value of erp\_azimuth\_max shall be in the range of −360 \* 216+1 (i.e., −23 592 959) to 360 \* 216 (i.e., 23 592 960), inclusive. When not present, the value of erp\_azimuth\_max is inferred to be equal to 180 \* 216 (i.e., 11 796 480).

The value of erp\_azimuth\_max shall be greater than erp\_azimuth\_min.

**erp\_elevation\_min** specifies the minimum elevation value of the coverage sphere region, in units of 2−16 degrees. The value of erp\_elevation\_min shall be in the range of −90 \* 216 (i.e., −5 898 240) to 90 \* 216−1 (i.e., 5 898 239), inclusive. When not present, the value of erp\_elevation\_min is inferred to be equal to −90 \* 216.

**erp\_elevation\_max** specifies the maximum elevation value of the coverage sphere region, in units of 2−16 degrees. The value of erp\_elevation\_max shall be in the range of −90 \* 216+1 (i.e., −5 898 239) to 90 \* 216 (i.e., 5 898 240), inclusive. When not present, the value of erp\_elevation\_max is inferred to be equal to 90 \* 216.

The value of erp\_elevation\_max shall be greater than erp\_elevation\_min.

**D.3.41.2 Cubemap projection SEI message semantics**

The cubemap projection SEI message provides information to enable remapping (through a cubemap projection) of the colour samples of the projected pictures onto a sphere coordinate space in sphere coordinates (ϕ, θ) for use in omnidirectional video applications for which the viewing perspective is from the origin looking outward toward the inside of the sphere. The sphere coordinates are defined so that ϕ is the azimuth (longitude, increasing eastward) and θ is the elevation (latitude, increasing northward).

Rotation angles yaw (α), pitch (β), and roll (γ) are also used in the specification of these semantics.

Relative to an (x, y, z) Cartesian coordinate system, yaw expresses a rotation around the z (vertical, up) axis, pitch rotates around the y (lateral, side-to-side) axis, and roll rotates around the x (back-to-front) axis. Rotations are extrinsic, i.e., around x, y, and z fixed reference axes. The angles increase clockwise when looking from the origin towards the positive end of an axis.

When a cubemap projection SEI message is present for any picture of a CLVS of a particular layer, a cubemap projection SEI message shall be present for the first picture of the CLVS and no SEI message indicating a different type of projection shall be present for any picture of the CLVS.

When general\_non\_packed\_constraint\_flag is equal to 1 in the active SPS for the current layer, there shall be no cubemap projection SEI messages applicable for any picture of the CLVS of the current layer.

When a frame packing arrangement SEI message with frame\_packing\_arrangement\_cancel\_flag equal to 0 is present that applies to the picture, and the value of frame\_packing\_arrangement\_type of the frame packing arrangement SEI message is not equal to 3, 4, or 5, or the value of quincunx\_sampling\_flag of the frame packing arrangement SEI message is not equal to 0, a cubemap projection SEI message with cmp\_cancel\_flag equal to 0 shall not be present that applies to the picture. Decoders shall ignore cubemap projection SEI messages when a frame packing arrangement SEI message with frame\_packing\_arrangement\_cancel\_flag equal to 0 is present that applies to the picture, and the value of frame\_packing\_arrangement\_type of the frame packing arrangement SEI message is not equal to 3, 4, or 5, or the value of quincunx\_sampling\_flag of the frame packing arrangement SEI message is not equal to 0.

When a segmented rectangular frame packing arrangement SEI message with segmented\_rect\_frame\_packing\_arrangement\_cancel\_flag equal to 0 is present that applies to the picture, a cubemap projection SEI message with cmp\_cancel\_flag equal to 0 shall not be present that applies to the picture. Decoders shall ignore cubemap projection SEI messages when a segmented rectangular frame packing arrangement SEI message with segmented\_rect\_frame\_packing\_arrangement\_cancel\_flag equal to 0 is present that applies to the picture.

**cmp\_cancel\_flag** equal to 1 indicates that the SEI message cancels the persistence of any previous cubemap projection SEI message in output order. cmp\_cancel\_flag equal to 0 indicates that cubemap projection information follows.

**cmp\_persistence\_flag** specifies the persistence of the cubemap projection SEI message for the current layer.

cmp\_persistence\_flag equal to 0 specifies that the cubemap projection SEI message applies to the current decoded picture only.

Let picA be the current picture. cmp\_persistence\_flag equal to 1 specifies that the cubemap projection SEI message persists for the current layer in output order until one or more of the following conditions are true:

– A new CLVS of the current layer begins.

– The bitstream ends.

– A picture picB in the current layer in an access unit containing a cubemap projection SEI message that is applicable to the current layer is output for which PicOrderCnt( picB ) is greater than PicOrderCnt( picA ), where PicOrderCnt( picB ) and PicOrderCnt( picA ) are the PicOrderCntVal values of picB and picA, respectively, immediately after the invocation of the decoding process for picture order count for picB.

**cmp\_rotation\_flag** equal to 1 indicates that a rotation for conversion between the global and the local coordinate systems applies. cmp\_rotation\_flag equal to 0 indicates that no rotation is applied and the global and local coordinate systems are identical.

**cmp**\_**reserved\_zero\_4bits** when present, shall be equal to 0 in bitstreams conforming to this version of this Specification. Other values for cmp\_reserved\_zero\_4bits are reserved for future use by ITU-T | ISO/IEC. Decoders shall ignore the value of cmp\_reserved\_zero\_4bits.

**cmp\_padding\_type** equal to 0 indicates that the sample values within the padding area are unspecified. cmp\_padding\_type equal to 1 indicates that the value of each sample inside the padding area is equal to the value of the spatially nearest sample outside the padding area in the adjacent padded face. cmp\_padding\_type equal to 2 indicates that the values of the samples inside the padding area equivalent to the values of the samples that are projected to the face neighbouring the padded face. cmp\_padding\_type equal to 3 indicates that the values of samples inside the padding area are derived through projection to the extended planar surface of the face adjacent to the padding area. The value of cmp\_padding\_type shall be in the range of 0 to 3.

**cmp**\_**reserved\_zero\_6bits**, when present, shall be equal to 0 in bitstreams conforming to this version of this Specification. Other values for cmp\_reserved\_zero\_6bits are reserved for future use by ITU-T | ISO/IEC. Decoders shall ignore the value of cmp\_reserved\_zero\_6bits.

**cmp\_padding\_chroma\_sample\_range\_minus1** plus 1 indicates the thickness of the padding areas in units of chroma samples. The value of cmp\_padding\_chroma\_sample\_range\_minus1 shall be in the range of 0 to 255.

**cmp\_yaw\_rotation** specifies the value of the yaw rotation angle, in units of 2−16 degrees. The value of cmp\_yaw\_rotation shall be in the range of −180 \* 216 (i.e., −11796480) to 180 \* 216 − 1 (i.e., 11796479), inclusive. When not present, the value of cmp\_yaw\_rotation is inferred to be equal to 0.

**cmp\_pitch\_rotation** specifies the value of the pitch rotation angle, in units of 2−16 degrees. The value of cmp\_pitch\_rotation shall be in the range of −90 \* 216 (i.e., −5898240) to 90 \* 216 (i.e., 5898240), inclusive. When not present, the value of cmp\_pitch\_rotation is inferred to be equal to 0.

**cmp\_roll\_rotation** specifies the value of the roll rotation angle, in units of 2−16 degrees. The value of cmp\_roll\_rotation shall be in the range of −180 \* 216 (i.e., −11796480) to 180 \* 216 − 1 (i.e., 11796479), inclusive. When not present, the value of cmp\_roll\_rotation is inferred to be equal to 0.

**D.3.41.3 Region-wise packing SEI message semantics**

The region-wise packing SEI message provides information to enable remapping of the colour samples of the cropped output pictures onto projected pictures.

**rwp\_cancel\_flag** equal to 1 indicates that the SEI message cancels the persistence of any previous region-wise packing SEI message in output order. rwp\_cancel\_flag equal to 0 indicates that region-wise packing information follows.

**rwp\_persistence\_flag** specifies the persistence of the region-wise packing SEI message for the current layer.

rwp\_persistence\_flag equal to 0 specifies that the region-wise packing SEI message applies to the current decoded picture only.

Let picA be the current picture. rwp\_persistence\_flag equal to 1 specifies that the region-wise packing SEI message persists for the current layer in output order until one or more of the following conditions are true:

– A new CLVS of the current layer begins.

– The bitstream ends.

– A picture picB in the current layer in an access unit containing a region-wise packing SEI message that is applicable to the current layer is output for which PicOrderCnt( picB ) is greater than PicOrderCnt( picA ), where PicOrderCnt( picB ) and PicOrderCnt( picA ) are the PicOrderCntVal values of picB and picA, respectively, immediately after the invocation of the decoding process for picture order count for picB.

When an equirectangular projection SEI message with erp\_cancel\_flag equal to 0 or a cubemap projection SEI message with cmp\_cancel\_flag equal to 0 is not present in the CLVS that applies to the current picture and precedes the region-wise packing SEI message in decoding order, a region-wise packing SEI message with rwp\_persistence\_flag equal to 0 shall not be present in the CLVS that applies to the current picture. Decoders shall ignore region-wise packing SEI messages with rwp\_persistence\_flag equal to 0 that do not follow, in decoding order, an equirectangular projection SEI message with erp\_cancel\_flag equal to 0 or a cubemap projection SEI message with cmp\_cancel\_flag equal to 0 in the CLVS that applies to the current picture.

**rwp\_reserved\_zero\_6bits** shall be equal to 0 in bitstreams conforming to this version of this Specification. Other values for rwp\_reserved\_zero\_6bits[ i ] are reserved for future use by ITU-T | ISO/IEC. Decoders shall ignore the value of rwp\_reserved\_zero\_6bits[ i ].

**num\_packed\_regions** specifies the number of packed regions. The value of num\_packed\_regions shall be greater than 0.

**proj\_picture\_width** and **proj\_picture\_height** specify the width and height, respectively, of the projected picture. The value of proj\_picture\_width and proj\_picture\_height shall be both greater than 0.

**rwp\_reserved\_zero\_4bits** shall be equal to 0 in bitstreams conforming to this version of this Specification. Other values for rwp\_reserved\_zero\_4bits[ i ] are reserved for future use by ITU-T | ISO/IEC. Decoders shall ignore the value of rwp\_reserved\_zero\_4bits[ i ].

**packing\_type**[ i ] specifies the type of region-wise packing. packing\_type[ i ] equal to 0 indicates rectangular region-wise packing. Other values are reserved. The value of packing\_type[ i ] shall be equal to 0 in this version of this Specification. Decoders shall allow values of packing\_type[ i ] greater than 0 and shall ignore all region-wise packing SEI messages with packing\_type[ i ] greater than 0 for any value of i.

**proj\_region\_width**[ i ], **proj\_region\_height**[ i ], **proj\_region\_top**[ i ], and **proj\_region\_left**[ i ] are indicated in units of luma samples in a projected picture with width and height equal to proj\_picture\_width and proj\_picture\_height, respectively.

**proj\_region\_width**[ i ] specifies the width of the i-th projected region. proj\_region\_width[ i ] shall be greater than 0.

**proj\_region\_height**[ i ] specifies the height of the i-th projected region. proj\_region\_height[ i ] shall be greater than 0.

**proj\_region\_top**[ i ] and **proj\_region\_left**[ i ] specify the top luma sample row and the left-most luma sample column, respectively, in the projected picture. The values of proj\_region\_top[ i ] and proj\_region\_left[ i ], shall be in the range from 0, inclusive, indicating the top-left corner of the projected picture, to proj\_picture\_height − 1, inclusive, and proj\_picture\_width − 1, inclusive, respectively.

The sum of proj\_region\_width[ i ] and proj\_region\_left[ i ] shall be less than proj\_picture\_width. The sum of proj\_region\_height[ i ] and proj\_region\_top[ i ] shall be less than proj\_picture\_height.

The values of proj\_region\_width[ i ], proj\_region\_height[ i ], proj\_region\_top[ i ] and proj\_region\_left[ i ] shall be such that the projected region identified by these fields is within a single constituent picture of the projected picture.

NOTE – Two projected regions may partially or entirely overlap with each other.

**transform\_type**[ i ] specifies the rotation and mirroring to be applied to the i-th packed region to remap to the i-th projected region. When transform\_type[ i ] specifies both rotation and mirroring, rotation applies before mirroring. The values of transform\_type[ i ] are specified in Table D. X:

Table D.X – transform\_type[ i ] values

|  |  |
| --- | --- |
| **Value** | **Description** |
| 0 | no transform |
| 1 | mirroring horizontally |
| 2 | rotation by 180 degrees (counter-clockwise) |
| 3 | rotation by 180 degrees (counter-clockwise) after mirroring horizontally |
| 4 | rotation by 90 degrees (counter-clockwise) before mirroring horizontally |
| 5 | rotation by 90 degrees (counter-clockwise) |
| 6 | rotation by 270 degrees (counter-clockwise) before mirroring horizontally |
| 7 | rotation by 270 degrees (counter-clockwise) |

**rwp\_reserved\_zero\_5bits** shall be equal to 0 in bitstreams conforming to this version of this Specification. Other values for rwp\_reserved\_zero\_5bits[ i ] are reserved for future use by ITU-T | ISO/IEC. Decoders shall ignore the value of rwp\_reserved\_zero\_5bits[ i ].

**packed\_region\_width**[ i ], **packed\_region\_height**[ i ], **packed\_region\_top**[ i ], and **packed\_region\_left**[ i ] specify the width, height, the top luma sample row, and the left-most luma sample column, respectively, of the packed region in the region-wise packed picture.

Let packedPicWidth and packedPicHeight be the width and height of the conformance cropping window of the region-wise packed picture. The values of packed\_region\_width[ i ], packed\_region\_height[ i ], packed\_region\_top[ i ], and packed\_region\_left[ i ] are constrained as follows:

* packed\_region\_width[ i ] and packed\_region\_height[ i ] shall both be greater than 0.
* The values of packed\_region\_top[ i ] and packed\_region\_left[ i ] shall be in the range from 0, inclusive, indicating the top-left corner luma sample of the region-wise packed picture, to packedPicHeight − 1, inclusive, and packedPicWidth − 1, inclusive, respectively.
* The sum of packed\_region\_width[ i ] and packed\_region\_left[ i ] shall be less than packedPicWidth.
* The sum of packed\_region\_height[ i ] and packed\_region\_top[ i ] shall be less than packedPicHeight.
* The rectangle specified by packed\_region\_width[ i ], packed\_region\_height[ i ], packed\_region\_top[ i ], and packed\_region\_left[ i ] shall be non-overlapping with the rectangle specified by packed\_region\_width[ j ], packed\_region\_height[ j ], packed\_region\_top[ j ], and packed\_region\_left[ j ] for any value of j in the range of 0 to i – 1, inclusive.

**D.3.41.4 Omnidirectional viewport SEI message semantics**

The omnidirectional viewport SEI message specifies the coordinates of one or more regions of spherical-coordinate geometry, bounded by four great circles, corresponding to viewports recommended for display.

**omni\_viewport\_id** contains an identifying number that may be used to identify the purpose of the one or more recommended viewport regions.

omni\_viewport\_id equal to 0 indicates that the recommended viewports are per director’s cut. omni\_viewport\_id equal to 1 indicates that the recommended viewports are most-viewed viewports by statistical measurements.

Values of omni\_viewport\_id from 2 to 511, inclusive, may be used as determined by the application. Values of omni\_viewport\_id from 512 to 1023 are reserved for future use by ITU-T | ISO/IEC. Decoders encountering a value of omni\_viewport\_id in the range of 512 to 1023, inclusive, shall ignore it.

**omni\_viewport\_cancel\_flag** equal to 1 indicates that the SEI message cancels the persistence of any previous omnidirectional viewport SEI message in output order. omni\_viewport\_cancel\_flag equal to 0 indicates that omnidirectional viewport information follows.

**omni\_viewport\_persistence\_flag** specifies the persistence of the omnidirectional viewport SEI message for the current layer.

omni\_viewport\_persistence\_flag equal to 0 specifies that the omnidirectional viewport SEI message applies to the current decoded picture only.

Let picA be the current picture. omni\_viewport\_persistence\_flag equal to 1 specifies that the omnidirectional viewport SEI message persists for the current layer in output order until one or more of the following conditions are true:

– A new CLVS of the current layer begins.

– The bitstream ends.

– A picture picB in the current layer in an access unit containing an omnidirectional viewport SEI message that is applicable to the current layer is output for which PicOrderCnt( picB ) is greater than PicOrderCnt( picA ), where PicOrderCnt( picB ) and PicOrderCnt( picA ) are the PicOrderCntVal values of picB and picA, respectively, immediately after the invocation of the decoding process for picture order count for picB.

When an equirectangular projection SEI message with erp\_cancel\_flag equal to 0 or a cubemap projection SEI message with cmp\_cancel\_flag equal to 0 is not present in the CLVS that applies to the current picture and precedes the omnidirectional viewport SEI message in decoding order, an omnidirectional viewport SEI message with omni\_viewport\_cancel\_flag equal to 0 shall not be present in the CLVS that applies to the current picture. Decoders shall ignore omnidirectional viewport SEI messages with omni\_viewport\_cancel\_flag equal to 0 that do not follow, in decoding order, an equirectangular projection SEI message with erp\_cancel\_flag equal to 0 or a cubemap projection SEI message with cmp\_cancel\_flag equal to 0 in the CLVS that applies to the current picture.

**omni\_viewport\_cnt\_minus1** plus 1 specifies the number of recommended viewport regions that are indicated by the SEI message.

When omni\_viewport\_id is equal to 0, between any two recommended viewports per director's cut, the i-th recommended viewport has higher priority than the j-th recommended viewport for any values of i and j when i is less than j. The 0-th recommended viewport per director's cut has the highest priority.

When omni\_viewport\_id is equal to 1, between any two most-viewed recommended viewports, the i-th recommended viewport has higher popularity than the j-th recommended viewport for any values of i and j when i is less than j. The 0-th most-viewed recommended viewport has the highest popularity.

**omni\_viewport\_azimuth\_centre**[ i ] and **omni\_viewport\_elevation\_centre**[ i ] indicate the centre of the i-th recommended viewport region, in units of 2−16 degrees. The value of omni\_viewport\_azimuth\_centre[ i ] shall be in the range of −180 \* 216 (i.e., −11796480) to 180 \* 216 − 1 (i.e., 11796479), inclusive. The value of omni\_viewport\_elevation\_centre[ i ] shall be in the range of −90 \* 216 (i.e., −5898240) to 90 \* 216 (i.e., 5898240), inclusive.

**omni\_viewport\_tilt\_centre**[ i ] indicates the tilt angle of the i-th recommended viewport region, in units of 2−16 degrees. The value of omni\_viewport\_tilt\_centre[ i ] shall be in the range of −180 \* 216 (i.e., −11796480) to  216 − 1 (i.e., 11796479), inclusive.

**omni\_viewport\_hor\_range**[ i ] indicates the azimuth range of the i-th recommended viewport region, in units of 2−16 degrees. The value of omni\_viewport\_hor\_range[ i ] shall be in the range of 1 to 360 \* 216 (i.e., 23592960), inclusive.

**omni\_viewport\_ver\_range**[ i ] indicates the elevation range of the i-th recommended viewport region, in units of 2−16 degrees. The value of omni\_viewport\_ver\_range[ i ] shall be in the range of 1 to 180 \* 216 (i.e., 11796480), inclusive.

**D.3.41.5 Sample location remapping process**

***D.3.41.5.1 General***

To remap colour sample locations of a region-wise packed picture to a unit sphere, the following ordered steps are applied:

– A region-wise packed picture is obtained as the cropped output picture by decoding a coded picture. For purposes of interpretation of chroma samples, the input to the indicated remapping process is the set of decoded sample values after applying an (unspecified) upsampling conversion process to the 4:4:4 colour sampling format as necessary when chroma\_format\_idc is equal to 1 (4:2:0 chroma format) or 2 (4:2:2 chroma format). This (unspecified) upsampling process should account for the relative positioning relationship between the luma and chroma samples as indicated by chroma\_sample\_loc\_type\_top\_field and chroma\_sample\_loc\_type\_bottom\_field, when present.

– If region-wise packing is indicated, the sample locations of the region-wise packed picture are converted to sample locations of the respective projected picture as specified in clause D.3.41.5.4. Otherwise, the projected picture is identical to the region-wise packed picture.

– If frame packing is indicated, the sample locations of the projected picture are converted to sample locations of the respective constituent picture of the projected picture, as specified in clause D.3.41.5.6. Otherwise, the constituent picture of the projected picture is identical to the projected picture.

– The sample locations of a constituent picture the projected picture are converted to sphere coordinates relative to the local coordinate axes, as specified in clause D.3.41.5.2.

– If rotation is indicated, the sphere coordinates relative to the local coordinate axes are converted to sphere coordinates relative to the global coordinate axes, as specified in clause D.3.41.5.3. Otherwise, the global coordinate axes are identical to the local coordinate axes.

The overall process for mapping of luma sample locations within a region-wise packed picture to sphere coordinates relative to the global coordinate axes is normatively specified in clause D.3.41.5.5.

For each region-wise packed picture corresponding to a decoded picture, the following applies:

– If an equirectangular projection SEI message with erp\_cancel\_flag equal to 0 and erp\_rotation\_flag equal to 1 that applies to the picture is present, or a cubemap projection SEI message with cmp\_cancel\_flag equal to 0 and cmp\_rotation\_flag equal to 1 that applies to the picture is present, RotationFlag is set equal to 1, and the following applies.

– If equirectangular projection is indicated, RotationYaw, RotationPitch, and RotationRoll are set equal to erp\_yaw\_rotation ÷ 216, erp\_pitch\_rotation ÷ 216, and erp\_roll\_rotation ÷ 216, respectively.

– Otherwise, RotationYaw, RotationPitch, and RotationRoll are set equal to cmp\_yaw\_rotation ÷ 216, cmp\_pitch\_rotation ÷ 216, and cmp\_roll\_rotation ÷ 216, respectively.

– Otherwise, RotationFlag is set equal to 0.

– If a frame packing arrangement SEI message with frame\_packing\_arrangement\_cancel\_flag equal to 0 that applies to the picture is not present, StereoFlag, TopBottomFlag, and SideBySideFlag are all set equal to 0, HorDiv1 is set equal to 1, and VerDiv1 is set equal to 1.

– Otherwise, the following applies:

– StereoFlag is set equal to 1.

– If the value of frame\_packing\_arrangement\_type of the frame packing arrangement SEI message is equal to 3, TopBottomFlag is set equal to 0, SideBySideFlag is set equal to 1, HorDiv1 is set equal to 2 and VerDiv1 is set equal to 1.

– Otherwise, if the value of frame\_packing\_arrangement\_type of the frame packing arrangement SEI message is equal to 4, TopBottomFlag is set equal to 1, SideBySideFlag is set equal to 0, HorDiv1 is set equal to 1, and VerDiv1 is set equal to 2.

– Otherwise, TopBottomFlag is set equal to 0, SideBySideFlag is set equal to 0, HorDiv1 is set equal to 1, and VerDiv1 is set equal to 1.

– If a region-wise packing SEI message with rwp\_cancel\_flag equal to 0 that applies to the picture is not present, RegionWisePackingFlag is set equal to 0, and ConstituentPicWidth and ConstituentPicHeight are set to be equal to cropPicWidth / HorDiv1 and cropPicHeight / VerDiv1, respectively, where cropPicWidth and cropPicHeight are the width and height, respectively, of the cropped output picture.

– Otherwise, RegionWisePackingFlag is set equal to 1, and ConstituentPicWidth and ConstituentPicHeight are set equal to proj\_picture\_width / HorDiv1 and proj\_picture\_height / VerDiv1, respectively.

***D.3.41.5.2 Projection for a sample location***

Inputs to this clause are:

– pictureWidth and pictureHeight, which are the width and height, respectively, of a monoscopic projected luma picture, in luma samples, and

– the centre point of a sample location (i, j) along the horizontal and vertical axes, respectively.

Outputs of this clause are:

– sphere coordinates (φ, θ) for the sample location in degrees relative to the coordinate axes specified in clause D.3.41.5.1.

The projection for a sample location is derived as follows:

– If equirectangular projection is indicated, the following applies:

φ = ( erp\_azimuth\_min + ( 0.5 − i ÷ pictureWidth ) \* ( erp\_azimuth\_max − erp\_azimuth\_min ) ) \* 2−16  
 (D‑XX)  
θ = ( erp\_elevation\_min + ( 0.5 − j ÷ pictureHeight ) \* ( erp\_elevation\_max − erp\_elevation\_min ) ) \* 2−16  
 (D‑XX)

– Otherwise (cubemap projection is indicated), it is a requirement of bitstream conformance that pictureWidth shall be a multiple of 3 and pictureHeight shall be a multiple of 2, and the following applies:

lw = pictureWidth / 3  
lh = pictureHeight / 2  
i′ = −( 2 \* ( i % lw ) ÷ lw ) + 1  
j′ = −( 2 \* ( j % lh ) ÷ lh ) + 1  
w = Floor( i ÷ lw )  
h = Floor( j ÷ lh )  
if( w = = 1 && h = = 0 ) { /\* front face \*/  
 x = 1.0  
 y = −i′  
 z = j′  
} else if( w = = 1 && h = = 1 ) { /\* back face \*/  
 x = −1.0  
 y = j′  
 z = −i′  
} else if( w = = 2 && h = = 1 ) { /\* top face \*/ (D‑XX)  
 x = −i′  
 y = j′  
 z = 1.0  
} else if( w = = 0 && h = = 1 ) { /\* bottom face \*/  
 x = i′  
 y = j′  
 z = −1.0′  
} else if( w = = 0 && h = = 0 ) { /\* right face \*/  
 x = −i′  
 y = −1.0  
 z = j′  
} else { /\* ( w = = 2 && h = = 0 ), left face \*/  
 x = i′  
 y = 1.0  
 z = j′  
}  
φ = Atan2( y, x ) \* 180 ÷ π  
θ =

***D.3.41.5.3 Conversion from the local coordinate axes to the global coordinate axes***

Inputs to this clause are:

– rotation\_yaw (α), rotation\_pitch (β), rotation\_roll (γ), all in units of degrees, and

– sphere coordinates (φ, θ) relative to the local coordinate axes.

Outputs of this clause are:

– sphere coordinates (φ′, θ′) relative to the global coordinate axes.

The outputs are derived as follows:

x1 = Cos( φ ) \* Cos( θ )  
y1 = Sin( φ ) \* Cos( θ )  
z1 = Sin( θ )  
x2 = Cos( β ) \* Cos ( γ ) \* x1 − Cos( β ) \* Sin( γ ) \* y1 + Sin( β ) \* z1  
y2 = ( Cos( α ) \* Sin( γ ) + Sin( α ) \* Sin( β ) \* Cos( γ ) ) \* x1 +  
 ( Cos( α ) \* Cos( γ ) − Sin( α ) \* Sin( β ) \* Sin( γ ) ) \* y1 −  
 Sin( α ) \* Cos( β ) \* z1 (D‑XX)  
z2 = ( Sin( α ) \* Sin( γ ) − Cos( α ) \* Sin( β ) \* Cos( γ ) ) \* x1 +  
 ( Sin( α ) \* Cos( γ ) + Cos( α ) \* Sin( β ) \* Sin( γ ) ) \* y1 +  
 Cos( α ) \* Cos( β ) \* z1  
φ′ = Atan2( y2, x2 ) \* 180 ÷ π  
θ′ = Asin( z2 ) \* 180 ÷ π

***D.3.41.5.4 Conversion of sample locations for rectangular region-wise packing***

Inputs to this clause are:

– sample location (x, y) within the packed region in integer luma sample units,

– the width and the height of the projected region in luma sample units (projRegWidth, projRegHeight),

– the width and the height of the packed region in sample units (packedRegWidth, packedRegHeight),

– transform type (transformType), and

– offset values for sampling position (offsetX, offsetY).

Outputs of this clause are:

– the centre point of the sample location (i, j) within the projected region in sample units.

The outputs are derived as follows:

if( transformType  = =  0  | |  transformType  = =  1  | |  transformType  = =  2  | |  transformType  = =  3 ) {  
 horRatio = projRegWidth ÷ packedRegWidth  
 verRatio = projRegHeight ÷ packedRegHeight  
} else if ( transformType  = =  4  | |  transformType  = =  5  | |  transformType  = =  6  | |  
 transformType  = =  7 ) {  
 horRatio = projRegWidth ÷ packedRegHeight  
 verRatio = projRegHeight ÷ packedRegWidth  
}  
if( transformType = = 0 ) {  
 i = horRatio \* ( x + offsetX )  
 j = verRatio \* ( y + offsetY )  
} else if ( transformType = = 1 ) {  
 i = horRatio \* ( packedRegWidth − x − offsetX )  
 j = verRatio \* ( y + offsetY )  
} else if ( transformType = = 2 ) {  
 i = horRatio \* ( packedRegWidth − x − offsetX )  
 j = verRatio \* ( packedRegHeight − y − offsetY ) (D‑XX)  
} else if ( transformType = = 3 ) {  
 i = horRatio \* ( x + offsetX )  
 j = verRatio \* ( packedRegHeight − y − offsetY )  
} else if ( transformType = = 4 ) {  
 i = horRatio \* ( y + offsetY )  
 j = verRatio \* ( x + offsetX )  
} else if ( transformType = = 5 ) {  
 i = horRatio \* ( y + offsetY )  
 j = verRatio \* ( packedRegWidth − x − offsetX )  
} else if ( transformType = = 6 ) {  
 i = horRatio \* ( packedRegHeight − y − offsetY )  
 j = verRatio \* ( packedRegWidth − x − offsetX )  
} else if ( transformType = = 7 ) {  
 i = horRatio \* ( packedRegHeight − y − offsetY )  
 j = verRatio \* ( x+ offsetX )  
}

***D.3.41.5.5 Mapping of luma sample locations within a region-wise packed picture to sphere coordinates relative to the global coordinate axes***

This clause specifies the semantics of luma sample locations within a region-wise packed picture to sphere coordinates relative to the global coordinate axes.

This clause uses variables HorDiv1, VerDiv1, RotationFlag, StereoFlag, TopBottomFlag, SideBySideFlag, ConstituentPicWidth, ConstituentPicHeight, and RegionWisePackingFlag that are derived specific to the type of the decoded picture for which the corresponding region-wise packed picture this clause is applied to.

offsetX is set equal to 0.5 and offsetY is set equal to 0.5.

If RegionWisePackingFlag is equal to 1, the following applies for each packed region n in the range of 0 to num\_regions − 1, inclusive:

– For each sample location (xPackedPicture, yPackedPicture) belonging to the n-th packed region with packing\_type[ n ] equal to 0 (i.e., with rectangular region-wise packing), the following applies:

– The corresponding sample location (xProjPicture, yProjPicture) of the projected picture is derived as follows:

– x is set equal to xPackedPicture – packed\_region\_left[ n ].

– y is set equal to yPackedPicture – packed\_region\_top[ n ].

– Clause D.3.41.5.4 is invoked with x, y, packed\_region\_width[ n ], packed\_region\_height[ n ], proj\_region\_width[ n ], proj\_region\_height[ n ], transform\_type[ n ], offsetX and offsetY as inputs, and the output is assigned to sample location (i, j).

– xProjPicture is set equal to proj\_region\_left[ n ] + i.

– When StereoFlag is equal to 0 or TopBottomFlag is equal to 1, and when xProjPicture is greater than or equal to proj\_picture\_width, xProjPicture is set equal to xProjPicture − proj\_picture\_width.

– When SideBySideFlag is equal to 1, the following applies:

– When proj\_region\_left[ n ] is less than proj\_picture\_width / 2 and xProjPicture is greater than or equal to proj\_picture\_width / 2, xProjPicture is set equal to xProjPicture − proj\_picture\_width / 2.

– When proj\_region\_left[n] is greater than or equal to proj\_picture\_width / 2 and xProjPicture is greater than or equal to proj\_picture\_width, xProjPicture is set equal to xProjPicture − proj\_picture\_width / 2.

– yProjPicture is set equal to proj\_region\_top[ n ] + j.

– Clause D.3.41.5.6 is invoked with xProjPicture, yProjPicture, ConstituentPicWidth, and ConstituentPicHeight as inputs, and the outputs indicating the sphere coordinates and the constituent picture index (for frame-packed stereoscopic video) for the luma sample location (xPackedPicture, yPackedPicture) belonging to the n-th packed region in the decoded picture.

Otherwise, the following applies for each sample location (x, y) within the region-wise packed picture:

– xProjPicture is set equal to x + offsetX.

– yProjPicture is set equal to y + offsetY.

– Clause D.3.41.5.6 is invoked with xProjPicture, yProjPicture, ConstituentPicWidth, and ConstituentPicHeight as inputs, and the outputs indicating the sphere coordinates and the constituent picture index (for frame-packed stereoscopic video) for the sample location (x, y) within the region-wise packed picture.

***D.3.41.5.6 Conversion from a sample location in a projected picture to sphere coordinates relative to the global coordinate axes***

Inputs to this clause are

– the centre point of a sample location (xProjPicture, yProjPicture) within a projected picture, and

– pictureWidth and pictureHeight, which are the width and height, respectively, of a monoscopic projected luma picture, in luma samples.

Outputs of this clause are:

– sphere coordinates (azimuthGlobal, elevationGlobal), in units of degrees relative to the global coordinate axes, and

– when StereoFlag is equal to 1, the index of the constituent picture (constituentPicture) equal to 0 or 1.

The outputs are derived with the following ordered steps:

– If xProjPicture is greater than or equal to pictureWidth or yProjPicture is greater than or equal to pictureHeight, the following applies:

– constituentPicture is set equal to 1.

– If xProjPicture is greater than or equal to pictureWidth, xProjPicture is set to xProjPicture − pictureWidth.

– If yProjPicture is greater than or equal to pictureHeight, yProjPicture is set to yProjPicture − pictureHeight.

– Otherwise, constituentPicture is set equal to 0.

– Clause D.3.41.5.2 is invoked with pictureWidth, pictureHeight, xProjPicture, and yProjPicture as inputs, and the output is assigned to azimuthLocal, elevationLocal.

– If RotationFlag is equal to 1, clause D.3.41.5.3 is invoked with azimuthLocal, elevantionLocal, RotationYaw, RotationPitch, and RotationRoll as inputs, and the output is assigned to azimuthGlobal and elevationGlobal.

– Otherwise, azimuthGlobal is set equal to azimuthLocal and elevationGlobal is set equal to elevationLocal.

**D.3.42 Regional nesting SEI message semantics**

The regional nesting SEI message provides a mechanism to associate SEI messages with regions of the picture. The associated SEI messages are conveyed within the regional nesting SEI message.

A regional nesting SEI message contains one or more SEI messages. When an SEI message is contained in a regional nesting SEI message, the contained SEI message is referred to as a region-nested SEI message. When an SEI message is not contained in a regional nesting SEI message, the SEI message is referred to as a non-region-nested SEI message.

For each region-nested SEI message in a regional nesting SEI message, one or more regions are specified in the regional nesting SEI message, and the semantics of the region-nested SEI message are to be interpreted as applying to each of these regions.

The list listOfRegionNestableMessageTypes includes the following types of SEI messages:

– Film grain characteristics SEI message,

– Post filter hint SEI message,

– Tone mapping information SEI message identified with a particular value of tone\_map\_id,

– Chroma resampling filter hint SEI message,

– Knee function information SEI message identified with a particular value of knee\_function\_id,

– Colour remapping information SEI message identified with a particular value of colour\_remap\_id,

– Content colour volume SEI message.

NOTE  – SEI messages of each of the following are considered different types of SEI messages: 1) tone mapping information SEI messages with different values of tone\_map\_id, 2) knee function information SEI messages with different values of knee\_function\_id, and 3) colour remapping information SEI messages with different values of colour\_remap\_id.

When an SEI message of a particular type in listOfRegionNestableMessageTypes has film\_grain\_characteristics\_cancel\_flag, tone\_map\_cancel\_flag, knee\_function\_cancel\_flag, or colour\_remap\_cancel\_flag equal to 1, regardless of whether it is region-nested or non-region-nested, it cancels the persistence of all the region-nested SEI messages of that type, regardless of their associated regions. When an SEI message of a particular type having film\_grain\_characteristics\_persistence\_flag, tone\_map\_persistence\_flag, knee\_function\_persistence\_flag, or colour\_remap\_persistence\_flag equal to 1 is region-nested, the persistence of the SEI message is determined by the semantics of the SEI message, irrespective of which region it applies to.

NOTE 2 – A region-nested SEI message has the same persistence scope as if the SEI message was non-region-nested.

NOTE 3 – A region-nested SEI message does not cancel the persistence of a non-region-nested SEI message of the same type.

The list listOfAllowedRegionalNestableMessageTypes includes all the entries in the list listOfRegionNestableMessageTypes and also the following additional types of SEI messages:

– User data registered by Rec. ITU-T T.35 SEI message,

– User data unregistered SEI message.

In bitstreams conforming to this version of this Specification, the regional nesting SEI message shall not contain any SEI message that is not in listOfAllowedRegionNestableMessageTypes. Decoders encountering a region-nested SEI message that does not belong to listOfAllowedRegionNestableMessageTypes shall ignore the region-nested SEI message.

When an access unit contains both region-nested SEI messages of a particular type in listOfRegionNestableMessageTypes and non-region-nested SEI messages of the same type, decoders shall ignore either all the region-nested SEI message of that type or all the non-region-nested SEI messages of that type. Unless indicated otherwise by some means not specified in this Specification, when an access unit contains both region-nested SEI messages of a particular type in listOfRegionNestableMessageTypes and non-region-nested SEI messages of the same type, the region-nested SEI messages should be preferred to be considered as applicable to the access unit.

A region-nested SEI messages should not be extracted and sent as a non-region-nested SEI message, as the values signalled in the region-nested SEI message may not be applicable outside the indicated regions.

**regional\_nesting\_id** contains an identifying number that may be used to identify the purpose of the one or more SEI messages that are region-nested in the regional nesting SEI message. The value of regional\_nesting\_id shall be in the range of 0 to 216 − 1, inclusive.

Values of regional\_nesting\_id from 0 to 255 and from 512 to 215 − 1 may be used as determined by the application. Values of regional\_nesting\_id from 256 to 511 and from 215 to 216 − 1 are reserved for future use by ITU-T | ISO/IEC. Decoders encountering a value of regional\_nesting\_id in the range of 256 to 511, inclusive, or in the range of 215 to 216 − 1, inclusive, shall ignore it.

**regional\_nesting\_num\_rect\_regions** specifies the number of rectangular regions specified in the regional nesting SEI message. The value of regional\_nesting\_num\_rect\_regions shall be in the range of 1 to 255, inclusive. The value of regional\_nesting\_num\_rect\_regions equal to 0 is reserved for future use by ITU-T | ISO/IEC and shall not be used in bitstreams conforming to this version of this Specification. Decoders shall ignore regional nesting SEI messages with regional\_nesting\_num\_rect\_regions equal to 0.

**regional\_nesting\_rect\_region\_id**[ i ] specifies the identifier for the i-th rectangular region specified in the regional nesting SEI message.

Unless indicated otherwise by some means not specified in this Specification, when a sample belongs to more than one region indicated as applying to more than one region-nested SEI message of a particular type in listOfRegionNestableMessageTypes, among these region-nested SEI messages, only those that are associated with the region that has the greatest value of regional\_nesting\_rect\_region\_id[ ] are considered as applying to the sample, while the rest of these region-nested SEI messages are considered as not applying to the sample.

NOTE 4 – When there are more than one of these region-nested SEI messages associated with the region that has the greatest value of regional\_nesting\_rect\_region\_id[ ], they are identical per other expressed constraints.

It is a requirement of bitstream conformance that the value of regional\_nesting\_rect\_region\_id[ i ] shall not be the same for any two different values of i in the range of 0 to regional\_nesting\_num\_rect\_regions − 1, inclusive, in the regional nesting SEI message.

When a region-nested SEI message of a particular type in listOfRegionNestableMessageTypes is indicated as applying to a list of regions listA in the current picture and another region-nested SEI message of the same type is indicated as applying to another list of regions listB in the current picture, it is a requirement of bitstream conformance that, for any pair of regions formed by choosing one from listA and the other from listB, the value of regional\_nesting\_rect\_region\_id[ ] of the two regions shall not be the same unless the two regions are identical (i.e., both position and size are the same) and the two region-nested SEI messages are identical.

**regional\_nesting\_rect\_‌left\_offset**[ i ], **regional\_nesting\_rect\_‌right\_offset**[ i ], **regional\_nesting\_rect\_‌top\_offset**[ i ], and **regional\_nesting\_rect\_‌bottom\_offset**[ i ] specify the coordinates of the i-th rectangular region specified in the SEI message. The offsets for the rectangular region are specified in units of luma samples. The i-th rectangular region contains the luma samples with horizontal picture coordinates from SubWidthC \* regional\_nesting\_rect\_**‌**left\_offset[ i ] to pic\_width\_in\_luma\_samples − ( SubWidthC \* regional\_nesting\_rect\_**‌**right\_offset[ i ] + 1 ), inclusive, and vertical picture coordinates from SubHeightC \* regional\_nesting\_rect\_**‌**top\_offset[ i ] to pic\_height\_in\_luma\_samples − ( SubHeightC \* regional\_nesting\_rect\_**‌**bottom\_offset[ i ] + 1 ), inclusive.

The value of SubWidthC \* ( regional\_nesting\_rect\_**‌**left\_offset[ i ] + regional\_nesting\_rect\_**‌**right\_offset[ i ] ) shall be less than pic\_width\_in\_luma\_samples and the value of SubHeightC \* ( regional\_nesting\_rect\_**‌**top\_offset[ i ] + regional\_nesting\_rect\_**‌**bottom\_offset[ i ] ) shall be less than pic\_height\_in\_luma\_samples.

**num\_sei\_messages\_in\_regional\_nesting\_minus1** plus 1 specifies the number of region-nested SEI messages in the regional nesting SEI message. The value of num\_sei\_messages\_in\_regional\_nesting\_minus1 shall be in the range of 0 to 255, inclusive.

**num\_regions\_for\_sei\_message**[ i ] specifies the number of regions to which the i-th region-nested SEI message is associated. When regional\_nesting\_num\_rect\_regions is greater than 0, the value of num\_regions\_for\_sei\_message[ i ] shall be in the range of 0 to regional\_nesting\_num\_rect\_regions, inclusive. When num\_regions\_for\_sei\_message[ i ] is equal to 0, the i-th region-nested SEI message applies to all the regions specified in the regional nesting SEI message.

**regional\_nesting\_sei\_region\_idx**[ i ][ j ] specifies the index, into the list of regions specified in the regional nesting SEI message, of the j-th region to which the i-th region-nested SEI message in the regional nesting SEI message is associated. The value of regional\_nesting\_sei\_region\_idx[ i ][ j ] shall be in the range of 0 to regional\_nesting\_num\_rect\_region − 1, inclusive.

**D.3.43 Motion-constrained tile sets extraction information sets SEI message semantics**

The motion-constrained tile sets extraction information sets SEI message provides supplemental information that can be used in the motion-constrained tile set (MCTS) sub-bitstream extraction as specified below to generate a conforming bitstream for an MCTS set. The information consists of a number of extraction information sets, each defining a number of MCTS sets and containing RBSP bytes of the replacement VPSs, SPSs, and PPSs to be used during the MCTS sub-bitstream extraction process. Each extraction information set can be shared by multiple MCTS sets, i.e., is used for extraction of any of these MCTS sets.

An MCTS extraction information sets SEI message shall not be present in an access unit unless there is a temporal MCTS SEI message present in the access unit. A temporal MCTS SEI message present in the same access unit as an MCTS extraction information sets SEI message is referred to as the associated temporal MCTS SEI message of the MCTS extraction information sets SEI message.

An MCTS extraction information sets SEI message applies to the same set of pictures as the associated temporal MCTS SEI message, i.e., the associatedPicSet of the MCTS extraction information sets SEI message is the same as the associatedPicSet of the associated temporal MCTS SEI message.

When more than one MCTS extraction information sets SEI message is present for the pictures of associatedPicSet, these MCTS extraction information sets SEI messages shall contain identical content.

NAL units that contain tiles belonging to any particular MCTS mctsA shall not contain tiles that do not belong to mctsA.

**num\_info\_sets\_minus1** plus 1 specifies the number of extraction information sets contained in the MCTS extraction information sets SEI message. The value of num\_info\_sets\_minus1 shall be in the range of 0 to 2047, inclusive.

The i-th extraction information set is assigned an MCTS extraction information set identifier value equal to i.

**num\_mcts\_sets\_minus1**[ i ] plus 1 specifies the number of MCTS sets that share the i-th extraction information set. The value of num\_mcts\_sets\_minus1[ i ] shall be in the range of 0 to 2047, inclusive.

**num\_mcts\_in\_set\_minus1**[ i ][ j ] plus 1 specifies the number of MCTSs in the j-th MCTS set that is associated with the i-th extraction information set. The value of num\_mcts\_in\_set\_minus1[ i ][ j ] shall be in the range of 0 to 511, inclusive.

**idx\_of\_mcts\_in\_set**[ i ][ j ][ k ] specifies the MCTS index of the k-th MCTS in the j-th MCTS set that is associated with the i-th extraction information set. The value of idx\_of\_mcts\_in\_set[ i ][ j ][ k ] shall be in the range of 0 to 511, inclusive.

**slice\_reordering\_enabled\_flag**[ i ] equal to 1 specifies that the MCTS sub-bitstream extraction using the i-th extraction information set includes reordering of extracted slices and that the slice\_segment\_address of all num\_slice\_segments\_minus1[ i ] + 1extracted slices are set to output\_slice\_segment\_address[ ][ ]. slice\_reordering\_enabled\_flag[ i ] equal to 0 indicates that the MCTS sub-bitstream extraction using the i-th extraction information set does not include a reordering of extracted slices and that the slice\_segment\_address of all extracted slices is calculated during extraction.

**num\_slice\_segments\_minus1**[ i ]plus 1 specifies the number of slice segments associated with any MCTS set of the i-th extraction information set when slice\_reordering\_enabled\_flag[ i ] is equal to 1. The value of num\_slice\_segments\_minus1[ i ] shall be in the range of 0 to 1024, inclusive.

**output\_slice\_segment\_address**[ i ][ j ]specifies the slice segment address of the j-th slice segment in bitstream order associated with any of the MCTS sets of the i-th extraction information set when slice\_reordering\_enabled\_flag[ i ] is equal to 1. The length of the output\_slice\_segment\_address[ i ][ j ] syntax element is Ceil( Log2( PicSizeInCtbsY ) ) bits. The value of output\_slice\_segment\_address shall be in the range of 0 to PicSizeInCtbsY − 1, inclusive and no value of output\_slice\_segment\_address[ i ][ j ] shall be equal to some output\_slice\_segment\_address[ i ][ k ] for j not equal to k in the extraction information set SEI message.

**num\_vps\_in\_info\_set\_minus1**[ i ] plus 1 specifies the number of replacement VPSs in the i-th extraction information set. The value of num\_vps\_in\_info\_set\_minus1[ i ] shall be in the range of 0 to 15, inclusive.

**vps\_rbsp\_data\_length**[ i ][ j ] specifies the number of RBSP data bytes of the j-th replacement VPS in the i-th extraction information set.

**num\_sps\_in\_info\_set\_minus1**[ i ] plus 1 specifies the number of replacement SPSs in the i-th extraction information set. The value of num\_sps\_in\_info\_set\_minus1[ i ] shall be in the range of 0 to 15, inclusive.

**sps\_rbsp\_data\_length**[ i ][ j ] specifies the number of RBSP data bytes of the j-th replacement SPS in the i-th extraction information set.

**num\_pps\_in\_info\_set\_minus1**[ i ] plus 1 specifies the number of replacement PPSs in the i-th extraction information set. The value of num\_pps\_in\_info\_set\_minus1[ i ] shall be in the range of 0 to 63, inclusive.

**pps\_nuh\_temporal\_id\_plus1**[ i ][ j ] minus1 specifies the temporal identifier of the j-th replacement PPS in the i-th extraction information set.

**pps\_rbsp\_data\_length**[ i ][ j ] specifies the number of RBSP data bytes of the j-th replacement PPS in the i-th extraction information set.

**mcts\_alignment\_bit\_equal\_to\_zero** shall be equal to 0.

**vps\_rbsp\_data\_byte**[ i ][ j ][ k ] contains the k-th byte of the RBSP data of the j-th replacement VPS in the i-th extraction information set.

**sps\_rbsp\_data\_byte**[ i ][ j ][ k ] contains the k-th byte of the RBSP data of the j-th replacement SPS in the i-th extraction information set.

**pps\_rbsp\_data\_byte**[ i ][ j ][ k ] contains the k-th byte of the RBSP data of the j-th replacement PPS in the i-th extraction information set.

The MCTS sub-bitstream extraction process is specified as follows:

* Let a bitstream inBitstream, a target MCTS set index mctsSetIdxTarget, a target MCTS extraction information set identifier mctsEisIdTarget, and a target highest TemporalId value mctsTidTarget be the inputs to the MCTS sub-bitstream extraction process.
* The output of the MCTS sub-bitstream extraction process is a sub-bitstream outBitstream derived as follows:

– The bitstream outBitstream is set to be identical to the bitstream inBitstream.

– The lists ausWithVps, ausWithSps. and ausWithPps are set to consist of all access units within outBitstream containing non-VCL NAL units with nal\_unit\_type equal to VPS\_NUT, SPS\_NUT, or PPS\_NUT.

– Remove all SEI NAL units that contain non-MCTS-nested SEI messages.

NOTE  – A "smart" bitstream extractor might include appropriate non-MCTS-nested SEI messages in the extracted MCTS sub-bitstream, provided that the SEI messages applicable to the MCTS sub-bitstream were present as MCTS-nested SEI messages in the original bitstream.

– Remove from outBitstream all of the following NAL units:

– VCL NAL units that contain tiles not belonging to any of the MCTSs with MCTS index equal to idx\_of\_mcts\_in\_set[ mctsEisIdTarget ][ mctsSetIdxTarget ][ k ] for each value of k in the range of 0 to num\_mcts\_in\_set\_minus1[ mctsEisIdTarget ][ mctsSetIdxTarget ], inclusive.

– Non-VCL NAL units with nal\_unit\_type equal to VPS\_NUT, SPS\_NUT, or PPS\_NUT.

– Insert into each access unit within the list ausWithVps in outBitstream num\_vps\_in\_info\_set\_minus1[ mctsEisIdTarget ] plus 1 VPS NAL units generated from the RBSP data of the list of replacement VPSs in the mctsEisIdTarget-th MCTS extraction information set. For each VPS NAL unit that is generated the nuh\_layer\_id is set equal to 0 and nuh\_temporal\_id\_plus1 is set equal to 1.

– Insert into each access unit within the list ausWithSps in outBitstream num\_sps\_in\_info\_set\_minus1[ mctsEisIdTarget ] plus 1 SPS NAL units generated from the RBSP data of the list of replacement SPSs in the mctsEisIdTarget-th MCTS extraction information set. For each SPS NAL unit that is generated the nuh\_layer\_id is set equal to 0 and nuh\_temporal\_id\_plus1 is set equal to 1.

– Insert into each access unit within the list ausWithPps in outBitstream PPS NAL units generated from the RBSP data of the replacement PPSs, in the mctsEisIdTarget-th MCTS extraction information set, for which pps\_nuh\_temporal\_id\_plus1[ mctsEisIdTarget ][ j ] is less than or equal to mctsTidTarget. For each PPS NAL unit that is generated the nuh\_layer\_id is set equal to 0, and for the PPS NAL unit that is generated from the RBSP data of the j-th replacement PPS in the mctsEisIdTarget-th MCTS extraction information set, nuh\_temporal\_id\_plus1 is set equal to pps\_nuh\_temporal\_id\_plus1[ mctsEisIdTarget ][ j ].

NOTE 2 – The values of pps\_pic\_parameter\_set\_idof the replacement PPSs should be identical to the values of pps\_pic\_parameter\_set\_idof the removed PPSs to retain the value of slice\_pic\_parameter\_set\_id in slice headers of the original bitstream.

– Remove from outBitstream all NAL units with TemporalId greater than mctsTidTarget.

– If slice\_reordering\_enabled\_flag[ mctsEISIdTarget ] is equal to 0, the coding tree block raster and tile scanning conversion process as specified in clause 6.5.1 is invoked with the syntax element values of the replacement SPS and PPS as inputs. The output CtbAddrRsToTs[ ctbAddrRs ] is assigned to extCtbAddrRsToTs[ ctbAddrRs ] and CtbAddrTsToRs[ ctbAddrTs ] is assigned to extCtbAddrTsToRs[ ctbAddrTs ]. For each remaining VCL NAL units in outBitstream, adjust the slice segment header as follows:

– For the first VCL NAL unit within each access unit, set the value of first\_slice\_segment\_in\_pic\_flag equal to 1, and set the value of slice\_segment\_address to be equal to 0.

– For each remaining VCL NAL units in outBitstream, let ctbAddrRs be the value of the raster scan address of the last CTB in the previous VCL NAL unit in bitstream order within a coded picture of outBitstream, set the value of first\_slice\_segment\_in\_pic\_flag equal to 0, and set the value of slice\_segment\_address equal to extCtbAddrTsToRs[ extCtbAddrRsToTs[ ctbAddrRs ] + 1 ].

– Otherwise (slice\_reordering\_enabled\_flag[ mctsEISIdTarget ] is equal to 1), the following applies:

* For the k-th VCL NAL units of each access unit in outBitstream, set the value of first\_slice\_segment\_in\_pic\_flag equal to 0 and slice\_segment\_address equal to output\_slice\_segement\_address[ mctsEISIdTarget ][ j ], where j is in the range of 0 to num\_slice\_segments\_minus1[ mctsEisIdTarget ], inclusive.
* Reorder the VCL NAL units within each access unit for ascending values of slice\_segment\_address.
* For the first VCL NAL unit within each access unit, set the value of first\_slice\_segment\_in\_pic\_flag equal to 1.

NOTE 3 – The extracted MCTS sub-bitstream might have a smaller luma picture size compared to the original bitstream which might require adjustment of the length of slice\_segment\_address and the byte\_alignment in slice headers.

It is a requirement of bitstream conformance for the input bitstream that any output sub-bitstream that is the output of the MCTS sub-bitstream extraction process specified in this clause shall be a conforming bitstream.

**D.3.44 Motion-constrained tile sets extraction information nesting SEI message semantics**

The motion-constrained tile sets extraction information nesting SEI message, also referred to as the MCTS nesting SEI message, provides a mechanism to carry and associate the SEI messages with bitstream subsets corresponding to one or more MCTSs. An SEI message contained in an MCTS nesting SEI message is referred to as MCTS-nested or an MCTS-nested SEI message, and an SEI message that is not contained in an MCTS nesting SEI message is referred to as a non-MCTS-nested SEI message.

In the MCTS sub-bitstream extraction process as specified in the semantics of the MCTS extraction information sets SEI message, the MCTS-nested SEI messages applicable to the MCTSs in an MCTS set in an access unit can be included in the corresponding access unit of the extracted sub-bitstream as non-MCTS-nested SEI messages.

An MCTS-nesting SEI message shall not be present for a picture unless the picture belongs to the associatedPicSet of a temporal MCTS SEI message. This temporal MCTS SEI message is referred to as the associated temporal MCTS SEI message of the MCTS-nesting SEI message.

An SEI NAL unit containing an MCTS nesting SEI message shall not contain any other SEI message that is not MCTS-nested in the MCTS nesting SEI message.

**all\_mcts\_flag** equal to 0 specifies that the list nesting\_mctsid is set to consist of the MCTS indices indicated by all instances of idx\_of\_associated\_mcts[ i ] for i from 0 to num\_associated\_mcts\_minus1, inclusive. all\_mcts\_flag equal to 1 specifies that the list nesting\_mctsid consists of the MCTS indices of all the MCTSs indicated by the associated temporal MCTS SEI message.

The MCTS-nested SEI messages apply to all MCTSs for which the MCTS indices are included in the list nesting\_mctsid.

**num\_associated\_mcts\_minus1** plus 1 specifies the number of the following MCTS indices. The value of num\_associated\_mcts\_minus1 shall be in the range of 0 to 511, inclusive.

**idx\_of\_associated\_mcts**[ i ] indicates the MCTS index of the i-th MCTS associated with the following MCTS-nested SEI messages. The value of idx\_of\_associated\_mcts[ i ] shall be in the range of 0 to 511, inclusive.

**num\_sei\_messages\_in\_mcts\_extraction\_nesting\_minus1** plus 1 specifies the number of the following MCTS-nested SEI messages.

**mcts\_nesting\_zero\_bit** shall be equal to 0.

*In E.3.1 (VUI parameters semantics), after the paragraphs specifying the semantics of chroma\_sample\_loc\_type\_top\_field and chroma\_sample\_loc\_type\_bottom\_field, add the following paragraph:*

When chroma\_format\_idc is equal to 1 and the decoded video content is intended for interpretation according to Rec. ITU-R BT.2020 or Rec. ITU-R BT.2100, chroma\_loc\_info\_present\_flag should be equal to 1, and chroma\_sample\_loc\_type\_top\_field and chroma\_sample\_loc\_type\_bottom\_field should both be equal to 2.

*In F.14.3.1 (General SEI payload semantics), in Table F.4, replace the row for the bitstream partition nesting SEI message with the following:*

|  |  |
| --- | --- |
| Bitstream partition nesting | Depending on the SEI messages contained in the bitstream partition nesting SEI message: each of these SEI messages has the same persistence scope as if the SEI message was not contained in the bitstream partition nesting SEI message. |

*In F.14.3.1 (General SEI payload semantics), replace the following paragraphs:*

The list VclAssociatedSeiList is set to consist of the payloadType values 2, 3, 6, 9, 15, 16, 17, 19, 22, 23, 45, 47, 56, 128, 131, 132, 134 to 148, inclusive, 161, 165, 167 and 168.

The list PicUnitRepConSeiList is set to consist of the payloadType values 0, 1, 2, 6, 9, 15, 16, 17, 19, 22, 23, 45, 47, 56, 128, 129, 131, 132, 133, 135 to 148, inclusive, and 160 to 168, inclusive.

*with the following:*

The list VclAssociatedSeiList is set to consist of the payloadType values 2, 3, 6, 9, 15, 16, 17, 19, 22, 23, 45, 47, 56, 128, 131, 132, 134 to 153, inclusive, 157 to 159, inclusive, 161, 165, 167, and 168.

The list PicUnitRepConSeiList is set to consist of the payloadType values 0, 1, 2, 6, 9, 15, 16, 17, 19, 22, 23, 45, 47, 56, 128, 129, 131, 132, 133, 135 to 153, inclusive, 157 to 159, inclusive, and 160 to 168, inclusive.

*In F.14.3.2.7 (Scalable nesting SEI message semantics for multi-layer extensions), replace the following phrase*

It is a requirement of bitstream conformance that the following restrictions apply on nesting of SEI messages:

*with the following:*

It is a requirement of bitstream conformance that the following restrictions apply on containing of SEI messages in a scalable nesting SEI message:

*In F.14.3.4 (Inter-layer constrained tile sets SEI message semantics), replace the following sentence:*

Decoders encountering an indicated value of ilcts\_id[ i ] in the range of 256 to 511, inclusive, or in the range of 231 to 232 − 2, inclusive, shall ignore (remove from the bitstream and discard) it.

*with the following:*

Decoders encountering an indicated value of ilcts\_id[ i ] in the range of 256 to 511, inclusive, or in the range of 231 to 232 − 2, inclusive, shall ignore it.

*In G.14.3.1 (General SEI payload semantics), replace the following paragraphs:*

The list VclAssociatedSeiList is set to consist of payloadType values 2, 3, 6, 9, 15, 16, 17, 19, 22, 23, 45, 47, 56, 128, 131, 132, 134 to 148, inclusive, 161, 165, 167, 168, 177, 178 and 179.

The list PicUnitRepConSeiList is set to consist of payloadType values 0, 1, 2, 6, 9, 15, 16, 17, 19, 22, 23, 45, 47, 56, 128, 129, 131, 132, 133, 135 to 148, inclusive, 160 to 168, inclusive, and 176 to 180, inclusive.

*with the following:*

The list VclAssociatedSeiList is set to consist of payloadType values 2, 3, 6, 9, 15, 16, 17, 19, 22, 23, 45, 47, 56, 128, 131, 132, 134 to 153, inclusive, 157 to 159, inclusive, 161, 165, 167, 168, 177, 178, and 179.

The list PicUnitRepConSeiList is set to consist of payloadType values 0, 1, 2, 6, 9, 15, 16, 17, 19, 22, 23, 45, 47, 56, 128, 129, 131, 132, 133, 135 to 153, inclusive, 157 to 159, inclusive, 160 to 168, inclusive, and 176 to 180, inclusive.

*In I.14.3.1 (General SEI payload semantics), replace the following paragraphs:*

The list VclAssociatedSeiList is set to consist of payloadType values 2, 3, 6, 9, 15, 16, 17, 19, 22, 23, 45, 47, 56, 128, 131, 132, 134 to 148, inclusive, 161, 165, 167, 168, 177, 178 and 179.

The list PicUnitRepConSeiList is set to consist of payloadType values 0, 1, 2, 6, 9, 15, 16, 17, 19, 22, 23, 45, 47, 56, 128, 129, 131, 132, 133, 135 to 148, inclusive, 160 to 168, inclusive, and 176 to 181, inclusive.

*with the following:*

The list VclAssociatedSeiList is set to consist of payloadType values 2, 3, 6, 9, 15, 16, 17, 19, 22, 23, 45, 47, 56, 128, 131, 132, 134 to 153, inclusive, 157 to 159, inclusive, 161, 165, 167, 168, 177, 178, and 179.

The list PicUnitRepConSeiList is set to consist of payloadType values 0, 1, 2, 6, 9, 15, 16, 17, 19, 22, 23, 45, 47, 56, 128, 129, 131, 132, 133, 135 to 153, inclusive, 157 to 159, inclusive, 160 to 168, inclusive, and 176 to 181, inclusive.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_