|  |  |
| --- | --- |
| **Joint Collaborative Team on Video Coding (JCT-VC)**  **of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11**  40th Meeting: by teleconference, 24 June – 1 July 2020 | Document: JCTVC-AN1006 |

|  |  |  |  |
| --- | --- | --- | --- |
| *Title:* | **Annotated regions and shutter interval information SEI messages for AVC (Draft 1)** | | |
| *Status:* | Output document approved by JCT-VC | | |
| *Purpose:* | Draft text | | |
| *Author(s) or Contact(s):* | Jill Boyce  Sean McCarthy  Ye-Kui Wang | Email: | jill.boyce@intel.com  sean.mccarthy@dolby.com  yekui.wang@bytedance.com |
| *Source:* | Editors | | |

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

# Abstract

This document contains the draft text for changes to the Advanced Video Coding (AVC) standard (Rec. ITU-T H.264 | ISO/IEC 14496-10) to specify the annotated regions and shutter interval information SEI messages.

**Changes to the specification text:**

*Replace 2.4 with the following:*



## Additional references

Recommendation ITU-T T.35 (in force), *Procedure for the allocation of ITU-T defined codes for non‑standard facilities*.

– ISO/IEC 10646: in force, *Information technology – Universal Coded Character Set (UCS)*.

– ISO/IEC 11578: in force, *Information technology – Open Systems Interconnection – Remote Procedure Call (RPC)*.

– ISO 11664-1: in force, *Colorimetry – Part 1: CIE standard colorimetric observers*.

– ISO 12232: in force, *Photography – Digital still cameras – Determination of exposure index, ISO speed ratings, standard output sensitivity, and recommended exposure index*.

– IETF RFC 1321 (in force), *The MD5 Message-Digest Algorithm*.

– IETF RFC 5646 (in force), *Tags for Identifying Languages.*

– ISO/IEC 23001-11 (in force), *Information Technology – MPEG Systems technologies – Part 11: Energy-efficient media consumption (green metadata).*

–– IETF RFC 5646 (2009), *Tags for Identifying Languages.*

*Replace D.1.1 with the following:*

**D.1.1 General SEI message syntax**

|  |  |  |
| --- | --- | --- |
| sei\_payload( payloadType, payloadSize ) { | C | Descriptor |
| if( payloadType = = 0 ) |  |  |
| buffering\_period( payloadSize ) | 5 |  |
| else if( payloadType = = 1 ) |  |  |
| pic\_timing( payloadSize ) | 5 |  |
| else if( payloadType = = 2 ) |  |  |
| pan\_scan\_rect( payloadSize ) | 5 |  |
| else if( payloadType = = 3 ) |  |  |
| filler\_payload( payloadSize ) | 5 |  |
| else if( payloadType = = 4 ) |  |  |
| user\_data\_registered\_itu\_t\_t35( payloadSize ) | 5 |  |
| else if( payloadType = = 5 ) |  |  |
| user\_data\_unregistered( payloadSize ) | 5 |  |
| else if( payloadType = = 6 ) |  |  |
| recovery\_point( payloadSize ) | 5 |  |
| else if( payloadType = = 7 ) |  |  |
| dec\_ref\_pic\_marking\_repetition( payloadSize ) | 5 |  |
| else if( payloadType = = 8 ) |  |  |
| spare\_pic( payloadSize ) | 5 |  |
| else if( payloadType = = 9 ) |  |  |
| scene\_info( payloadSize ) | 5 |  |
| else if( payloadType = = 10 ) |  |  |
| sub\_seq\_info( payloadSize ) | 5 |  |
| else if( payloadType = = 11 ) |  |  |
| sub\_seq\_layer\_characteristics( payloadSize ) | 5 |  |
| else if( payloadType = = 12 ) |  |  |
| sub\_seq\_characteristics( payloadSize ) | 5 |  |
| else if( payloadType = = 13 ) |  |  |
| full\_frame\_freeze( payloadSize ) | 5 |  |
| else if( payloadType = = 14 ) |  |  |
| full\_frame\_freeze\_release( payloadSize ) | 5 |  |
| else if( payloadType = = 15 ) |  |  |
| full\_frame\_snapshot( payloadSize ) | 5 |  |
| else if( payloadType = = 16 ) |  |  |
| progressive\_refinement\_segment\_start( payloadSize ) | 5 |  |
| else if( payloadType = = 17 ) |  |  |
| progressive\_refinement\_segment\_end( payloadSize ) | 5 |  |
| else if( payloadType = = 18 ) |  |  |
| motion\_constrained\_slice\_group\_set( payloadSize ) | 5 |  |
| else if( payloadType = = 19 ) |  |  |
| film\_grain\_characteristics( payloadSize ) | 5 |  |
| else if( payloadType = = 20 ) |  |  |
| deblocking\_filter\_display\_preference( payloadSize ) | 5 |  |
| else if( payloadType = = 21 ) |  |  |
| stereo\_video\_info( payloadSize ) | 5 |  |
| else if( payloadType = = 22 ) |  |  |
| post\_filter\_hint( payloadSize ) | 5 |  |
| else if( payloadType = = 23 ) |  |  |
| tone\_mapping\_info( payloadSize ) | 5 |  |
| else if( payloadType = = 24 ) |  |  |
| scalability\_info( payloadSize ) /\* specified in Annex ‎G \*/ | 5 |  |
| else if( payloadType = = 25 ) |  |  |
| sub\_pic\_scalable\_layer( payloadSize ) /\* specified in Annex ‎G \*/ | 5 |  |
| else if( payloadType = = 26 ) |  |  |
| non\_required\_layer\_rep( payloadSize ) /\* specified in Annex ‎G \*/ | 5 |  |
| else if( payloadType = = 27 ) |  |  |
| priority\_layer\_info( payloadSize ) /\* specified in Annex ‎G \*/ | 5 |  |
| else if( payloadType = = 28 ) |  |  |
| layers\_not\_present( payloadSize ) /\* specified in Annex ‎G \*/ | 5 |  |
| else if( payloadType = = 29 ) |  |  |
| layer\_dependency\_change( payloadSize ) /\* specified in Annex ‎G \*/ | 5 |  |
| else if( payloadType = = 30 ) |  |  |
| scalable\_nesting( payloadSize ) /\* specified in Annex ‎G \*/ | 5 |  |
| else if( payloadType = = 31 ) |  |  |
| base\_layer\_temporal\_hrd( payloadSize ) /\* specified in Annex ‎G \*/ | 5 |  |
| else if( payloadType = = 32 ) |  |  |
| quality\_layer\_integrity\_check( payloadSize ) /\* specified in Annex ‎G \*/ | 5 |  |
| else if( payloadType = = 33 ) |  |  |
| redundant\_pic\_property( payloadSize ) /\* specified in Annex ‎G \*/ | 5 |  |
| else if( payloadType = = 34 ) |  |  |
| tl0\_dep\_rep\_index( payloadSize ) /\* specified in Annex ‎G \*/ | 5 |  |
| else if( payloadType = = 35 ) |  |  |
| tl\_switching\_point( payloadSize ) /\* specified in Annex ‎G \*/ | 5 |  |
| else if( payloadType = = 36 ) |  |  |
| parallel\_decoding\_info( payloadSize ) /\* specified in Annex ‎H \*/ | 5 |  |
| else if( payloadType = = 37 ) |  |  |
| mvc\_scalable\_nesting( payloadSize ) /\* specified in Annex ‎H \*/ | 5 |  |
| else if( payloadType = = 38 ) |  |  |
| view\_scalability\_info( payloadSize ) /\* specified in Annex ‎H \*/ | 5 |  |
| else if( payloadType = = 39 ) |  |  |
| multiview\_scene\_info( payloadSize ) /\* specified in Annex ‎H \*/ | 5 |  |
| else if( payloadType = = 40 ) |  |  |
| multiview\_acquisition\_info( payloadSize ) /\* specified in Annex ‎H \*/ | 5 |  |
| else if( payloadType = = 41 ) |  |  |
| non\_required\_view\_component( payloadSize ) /\* specified in Annex ‎H \*/ | 5 |  |
| else if( payloadType = = 42 ) |  |  |
| view\_dependency\_change( payloadSize ) /\* specified in Annex ‎H \*/ | 5 |  |
| else if( payloadType = = 43 ) |  |  |
| operation\_points\_not\_present( payloadSize ) /\* specified in Annex ‎H \*/ | 5 |  |
| else if( payloadType = = 44 ) |  |  |
| base\_view\_temporal\_hrd( payloadSize ) /\* specified in Annex ‎H \*/ | 5 |  |
| else if( payloadType = = 45 ) |  |  |
| frame\_packing\_arrangement( payloadSize ) | 5 |  |
| else if( payloadType = = 46 ) |  |  |
| multiview\_view\_position( payloadSize ) /\* specified in Annex ‎H \*/ | 5 |  |
| else if( payloadType = = 47 ) |  |  |
| display\_orientation( payloadSize ) | 5 |  |
| else if( payloadType = = 48 ) |  |  |
| mvcd\_scalable\_nesting( payloadSize ) /\* specified in Annex ‎I \*/ | 5 |  |
| else if( payloadType = = 49 ) |  |  |
| mvcd\_view\_scalability\_info( payloadSize ) /\* specified in Annex ‎I \*/ | 5 |  |
| else if( payloadType = = 50 ) |  |  |
| depth\_representation\_info( payloadSize ) /\* specified in Annex ‎I \*/ | 5 |  |
| else if( payloadType = = 51 ) |  |  |
| three\_dimensional\_reference\_displays\_info( payloadSize )  /\* specified in Annex ‎I \*/ | 5 |  |
| else if( payloadType = = 52 ) |  |  |
| depth\_timing( payloadSize ) /\* specified in Annex ‎I \*/ | 5 |  |
| else if( payloadType = = 53 ) |  |  |
| depth\_sampling\_info( payloadSize ) /\* specified in Annex ‎I \*/ | 5 |  |
| else if( payloadType = = 54 ) |  |  |
| constrained\_depth\_parameter\_set\_identifier( payloadSize )   /\* specified in Annex ‎J \*/ | 5 |  |
| else if( payloadType = = 56 ) |  |  |
| green\_metadata( payloadSize ) /\* specified in ISO/IEC 23001-11 \*/ | 5 |  |
| else if( payloadType = = 137 ) |  |  |
| mastering\_display\_colour\_volume( payloadSize ) | 5 |  |
| else if( payloadType = = 142 ) |  |  |
| colour\_remapping\_info( payloadSize ) | 5 |  |
| else if( payloadType = = 144 ) |  |  |
| content\_light\_level\_info( payloadSize ) | 5 |  |
| else if( payloadType = = 147 ) |  |  |
| alternative\_transfer\_characteristics( payloadSize ) | 5 |  |
| else if( payloadType = = 148 ) |  |  |
| ambient\_viewing\_environment( payloadSize ) | 5 |  |
| else if( payloadType = = 149 ) |  |  |
| content\_colour\_volume( payloadSize ) | 5 |  |
| else if( payloadType = = 150 ) |  |  |
| equirectangular\_projection( payloadSize ) | 5 |  |
| else if( payloadType = = 151 ) |  |  |
| cubemap\_projection( payloadSize ) | 5 |  |
| else if( payloadType = = 154 ) |  |  |
| sphere\_rotation( payloadSize ) | 5 |  |
| else if( payloadType = = 155 ) |  |  |
| regionwise\_packing( payloadSize ) | 5 |  |
| else if( payloadType = = 156 ) |  |  |
| omni\_viewport( payloadSize ) | 5 |  |
| else if( payloadType = = 181 ) |  |  |
| alternative\_depth\_info( payloadSize ) /\* specified in Annex ‎I \*/ | 5 |  |
| else if( payloadType = = 200 ) |  |  |
| sei\_manifest( payloadSize ) | 5 |  |
| else if( payloadType = = 201 ) |  |  |
| sei\_prefix\_indication( payloadSize ) | 5 |  |
| else if( payloadType = = 202) |  |  |
| annotated\_regions( payloadSize ) | 5 |  |
| else if( payloadType = = 203 ) |  |  |
| shutter\_interval\_info( payloadSize ) | 5 |  |
| else |  |  |
| reserved\_sei\_message( payloadSize ) | 5 |  |
| if( !byte\_aligned( ) ) { |  |  |
| **bit\_equal\_to\_one** /\* equal to 1 \*/ | 5 | f(1) |
| while( !byte\_aligned( ) ) |  |  |
| **bit\_equal\_to\_zero** /\* equal to 0 \*/ | 5 | f(1) |
| } |  |  |
| } |  |  |

*Renumber clause D.1.38 (Reserved SEI message syntax) as D.1.40.*

*Add clauses D.1.38 and D.1.39, as follows:*

**D.1.38 Annotated regions SEI message syntax**

|  |  |  |
| --- | --- | --- |
| annotated\_regions( payloadSize ) { | **C** | **Descriptor** |
| **ar\_cancel\_flag** | 5 | u(1) |
| if(!ar\_cancel\_flag) { |  |  |
| **ar\_not\_optimized\_for\_viewing\_flag** | 5 | u(1) |
| **ar\_true\_motion\_flag** | 5 | u(1) |
| **ar\_occluded\_object\_flag** | 5 | u(1) |
| **ar\_partial\_object\_flag\_present\_flag** | 5 | u(1) |
| **ar\_object\_label\_present\_flag** | 5 | u(1) |
| **ar\_object\_confidence\_info\_present\_flag** | 5 | u(1) |
| if( ar\_object\_confidence\_info\_present\_flag ) |  |  |
| **ar\_object\_confidence\_length\_minus1** | 5 | u(4) |
| if( ar\_object\_label\_present\_flag ) { |  |  |
| **ar\_object\_label\_language\_present\_flag** | 5 | u(1) |
| if( ar\_object\_label\_language\_present\_flag ) { |  |  |
| while( !byte\_aligned( ) ) |  |  |
| **ar\_bit\_equal\_to\_zero** /\* equal to 0 \*/ | 5 | f(1) |
| **ar\_object\_label\_language** | 5 | st(v) |
| } |  |  |
| **ar\_num\_label\_updates** | 5 | ue(v) |
| for( i = 0; i < ar\_num\_ label\_updates; i++ ) { |  |  |
| **ar\_label\_idx**[ i ] | 5 | ue(v) |
| **ar\_label\_cancel\_flag** | 5 | u(1) |
| LabelAssigned[ ar\_label\_idx[ i ] ] = !ar\_label\_cancel\_flag |  |  |
| if( !ar\_label\_cancel\_flag ) { |  |  |
| while( !byte\_aligned( ) ) |  |  |
| **ar\_bit\_equal\_to\_zero** /\* equal to 0 \*/ | 5 | f(1) |
| **ar\_label**[ ar\_label\_idx[ i ] ] | 5 | st(v) |
| } |  |  |
| } |  |  |
| } |  |  |
| **ar\_num\_object\_updates** | 5 | ue(v) |
| for( i = 0; i  <=  ar\_num\_object\_updates; i++ ) { |  |  |
| **ar\_object\_idx**[ i ] | 5 | ue(v) |
| **ar\_object\_cancel\_flag** | 5 | u(1) |
| ObjectTracked[ ar\_object\_idx[ i ] ] = !ar\_object\_cancel\_flag |  |  |
| if( !ar\_object\_cancel\_flag ) { |  |  |
| if( ar\_object\_label\_present\_flag ) { |  |  |
| **ar\_object\_label\_update\_flag** | 5 | u(1) |
| if( ar\_object\_label\_update\_flag ) |  |  |
| **ar\_object\_label\_idx**[ ar\_object\_idx[ i ] ] | 5 | ue(v) |
| } |  |  |
| **ar\_bounding\_box\_update\_flag** | 5 | u(1) |
| if( ar\_bounding\_box\_update\_flag ) { |  |  |
| **ar\_bounding\_box\_cancel\_flag** | 5 | u(1) |
| ObjectBoundingBoxAvail[ ar\_object\_idx[ i ] ] = !ar\_bounding\_box\_cancel\_flag |  |  |
| if( !ar\_bounding\_box\_cancel\_flag ) { |  |  |
| **ar\_bounding\_box\_top[** ar\_object\_idx[ i ] ] | 5 | u(16) |
| **ar\_bounding\_box\_left**[ ar\_object\_idx[ i ] ] | 5 | u(16) |
| **ar\_bounding\_box\_width**[ ar\_object\_idx[ i ] ] | 5 | u(16) |
| **ar\_bounding\_box\_height**[ ar\_object\_idx[ i ] ] | 5 | u(16) |
| if( ar\_partial\_object\_flag\_present\_flag ) |  |  |
| **ar\_partial\_object\_flag**[ ar\_object\_idx[ i ] ] | 5 | u(1) |
| if( ar\_object\_confidence\_info\_present\_flag ) |  |  |
| **ar\_object\_confidence**[ ar\_object\_idx[ i ] ] | 5 | u(v) |
| } |  |  |
| } |  |  |
| } |  |  |
| } |  |  |
| } |  |  |
| } |  |  |

**D.1.39 Shutter interval information SEI message syntax**

|  |  |  |
| --- | --- | --- |
| shutter\_interval\_info( payloadSize ) { | **C** | **Descriptor** |
| **sii\_sub\_layer\_idx** | 5 | ue(v) |
| if( sii\_sub\_layer\_idx = = 0 ) |  |  |
| **shutter\_interval\_info\_present\_flag** | 5 | u(1) |
| if( shutter\_interval\_info\_present\_flag ) |  |  |
| **sii\_time\_scale** | 5 | u(32) |
| **fixed\_shutter\_interval\_within\_cvs\_flag** | 5 | u(1) |
| if( fixed\_shutter\_interval\_within\_cvs\_flag ) |  |  |
| **sii\_num\_units\_in\_shutter\_interval** | 5 | u(32) |
| else { |  |  |
| **sii\_max\_sub\_layers\_minus1** | 5 | u(3) |
| for( i = 0; i <= sii\_max\_sub\_layers\_minus1; i++ ) |  |  |
| **sub\_layer\_num\_units\_in\_shutter\_interval**[ i ] | 5 | u(32) |
| } |  |  |
| } |  |  |
| } |  |  |
| } |  |  |

*Renumber clause D.2.38 (Reserved SEI message semantics) as D.2.40.*

*Add clauses D.2.38 and D.2.39, as follows:*

D.2.38 Annotated regions SEI message semantics

The annotated regions SEI message carries parameters that identify annotated regions using bounding boxes representing the size and location of identified objects.

**ar\_cancel\_flag** equal to 1 indicates that the annotated regions SEI message cancels the persistence of any previous annotated regions SEI message that is associated with one or more layers to which the annotated regions SEI message applies. ar\_cancel\_flag equal to 0 indicates that annotated regions information follows.

When ar\_cancel\_flag equal to 1 or a new CVS of the current layer begins, the variables LabelAssigned[ i ], ObjectTracked[ i ], and ObjectBoundingBoxAvail are set equal to 0 for i in the range of 0 to 255, inclusive.

**ar\_not\_optimized\_for\_viewing\_flag** equal to 1 indicates that the decoded pictures that the annotated regions SEI message applies to are not optimized for user viewing, but rather are optimized for some other purpose such as algorithmic object classification performance. ar\_not\_optimized\_for\_viewing\_flagequal to 0 indicates that the decoded pictures that the annotated regions SEI message applies to may or may not be optimized for user viewing.

**ar\_true\_motion\_flag** equal to 1 indicates that the motion information in the coded pictures that the annotated regions SEI message applies to was selected with a goal of accurately representing object motion for objects in the annotated regions. ar\_true\_motion\_flag equal to 0 indicates that the motion information in the coded pictures that the annotated regions SEI message applies to may or may not be selected with a goal of accurately representing object motion for objects in the annotated regions.

**ar\_occluded\_object\_flag** equal to 1 indicates that the ar\_bounding\_box\_top[ ar\_object\_idx[ i ] ], ar\_bounding\_box\_left[ ar\_object\_idx[ i ] ], ar\_bounding\_box\_width[ ar\_object\_idx[ i ] ], and ar\_bounding\_box\_height[ ar\_object\_idx[ i ] ] syntax elements each represent the size and location of an object or a portion of an object that may not be visible or may be only partially visible within the cropped decoded picture. ar\_occluded\_object\_flagequal to 0 indicates that the ar\_bounding\_box\_top[ ar\_object\_idx[ i ] ], ar\_bounding\_box\_left[ ar\_object\_idx[ i ] ], ar\_bounding\_box\_width[ ar\_object\_idx[ i ] ], and ar\_bounding\_box\_height[ ar\_object\_idx[ i ] ] syntax elements represent the size and location of an object that is entirely visible within the cropped decoded picture. It is a requirement of bitstream conformance that the value of ar\_occluded\_object\_flag shall be the same for all annotated\_regions( ) syntax structures within a CVS.

**ar\_partial\_object\_flag\_present\_flag** equal to 1 indicates that ar\_partial\_object\_flag[ ar\_object\_idx[ i ] ] syntax elements are present. ar\_partial\_object\_flag\_present\_flag equal to 0 indicates that ar\_partial\_object\_flag[ ar\_object\_idx[ i ] ] syntax elements are not present. It is a requirement of bitstream conformance that the value of ar\_partial\_object\_flag\_present\_flag shall be the same for all annotated\_regions( ) syntax structures within a CVS.

**ar\_object\_label\_present\_flag** equal to 1 indicates that label information corresponding to objects in the annotated regions is present. ar\_object\_label\_present\_flag equal to 0 indicates that label information corresponding to the objects in the annotated regions is not present.

**ar\_object\_confidence\_info\_present\_flag** equal to 1 indicates that ar\_object\_confidence[ ar\_object\_idx[ i ] ] syntax elements are present. ar\_object\_confidence\_info\_present\_flag equal to 0 indicates that ar\_object\_confidence[ ar\_object\_idx[ i ] ] syntax elements are not present. It is a requirement of bitstream conformance that the value of ar\_object\_confidence\_present\_flag shall be the same for all annotated\_regions( ) syntax structures within a CVS.

**ar\_object\_confidence\_length\_minus1** + 1 specifies the length, in bits, of the ar\_object\_confidence[ ar\_object\_idx[ i ] ] syntax elements. It is a requirement of bitstream conformance that the value of ar\_object\_confidence\_length\_minus1 shall be the same for all annotated\_regions( ) syntax structures within a CVS.

**ar\_object\_label\_language\_present\_flag** equal to 1 indicates that the ar\_object\_label\_language syntax element is present. ar\_object\_label\_language\_present\_flag equal to 0 indicates that the ar\_object\_label\_language syntax element is not present.

**ar\_bit\_equal\_to\_zero** shall be equal to zero.

**ar\_object\_label\_language** contains a language tag as specified by IETF RFC 5646 followed by a null termination byte equal to 0x00. The length of the ar\_object\_label\_language syntax element shall be less than or equal to 255 bytes, not including the null termination byte. When not present, the language of the label is unspecified.

**ar\_num\_label\_updates** indicates the total number of labels associated with the annotated regions that will be signalled. The value of ar\_num\_label\_updates shall be in the range of 0 to 255, inclusive.

**ar\_label\_idx**[ i ] indicates the index of the signalled label . The value of ar\_label\_idx[ i ]shall be in the range of 0 to 255, inclusive.

**ar\_label\_cancel\_flag** equal to 1 cancels the persistence scope of the ar\_label\_idx[ i ]-th label. ar\_label\_cancel\_flag equal to 0 indicates that the ar\_label\_idx[ i ]-th label will be assigned a signalled value.

**ar\_label**[ ar\_label\_idx[ i ] ] specifies the contents of the ar\_label\_idx[ i ] –th label. The length of the ar\_label[ ar\_label\_idx[ i ] ] syntax element shall be less than or equal to 255 bytes, not including the null termination byte.

**ar\_num\_object\_updates** indicates the number of object updates to be signalled. ar\_num\_object\_updates shall be in the range of 0 to 255, inclusive.

**ar\_object\_idx**[ i ] is the index of the object parameters to be signalled. ar\_object\_idx[ i ]shall be in the range of 0 to 255, inclusive.

**ar\_object\_cancel\_flag** equal to 1 cancels the persistence scope of the ar\_object\_idx[ i ]-th object. ar\_object\_cancel\_flag equal to 0 indicates that parameters associated with the ar\_object\_idx[ i ]-th object tracked object will be signalled.

**ar\_object\_label\_update\_flag** equal to 1 indicates that an object label will be signalled. ar\_object\_label\_update\_flag equal to 0 indicates that an object label will not will not be signalled.

**ar\_object\_label\_idx**[ ar\_object\_idx[ i ] ] indicates the index of the label corresponding to the ar\_object\_idx[ i ]-th object. When ar\_object\_label\_idx[ ar\_object\_idx[ i ] ] is not present, its value is inferred from a previous annotated regions SEI messages in output order in the same CVS, if any.

**ar\_bounding\_box\_update\_flag** equal to 1 indicates that object bounding box parameters will be signalled. ar\_bounding\_box\_update\_flag equal to 0 indicates that object bounding box parameters will not be signalled.

**ar\_bounding\_box\_cancel\_flag** equal to 1 cancels the persistence scope of the ar\_bounding\_box\_top[ ar\_object\_idx[ i ] ], ar\_bounding\_box\_left[ ar\_object\_idx[ i ] ], ar\_bounding\_box\_width[ ar\_object\_idx[ i ] ], ar\_bounding\_box\_height[ ar\_object\_idx[ i ] ]. ar\_partial\_object\_flag[ ar\_object\_idx[ i ] ], and ar\_object\_confidence[ ar\_object\_idx[ i ] ]. ar\_bounding\_box\_cancel\_flag equal to 0 indicates that ar\_bounding\_box\_top[ ar\_object\_idx[ i ] ], ar\_bounding\_box\_left[ ar\_object\_idx[ i ] ], ar\_bounding\_box\_width[ ar\_object\_idx[ i ] ] ar\_bounding\_box\_height[ ar\_object\_idx[ i ] ] ar\_partial\_object\_flag[ ar\_object\_idx[ i ] ], and ar\_object\_confidence[ ar\_object\_idx[ i ] ] syntax elements will be signalled.

**ar\_bounding\_box\_top**[ ar\_object\_idx[ i ] ], **ar\_bounding\_box\_left**[ ar\_object\_idx[ i ] ], **ar\_bounding\_box\_width**[ ar\_object\_idx[ i ] ], and **ar\_bounding\_box\_height**[ ar\_object\_idx[ i ] ] specify the coordinates of the top-left corner and the width and height, respectively, of the bounding box of the ar\_object\_idx[ i ]-th object in the cropped decoded picture, relative to the conformance cropping window specified by the active SPS.

Let croppedWidth and croppedHeight be the width and height, respectively, of the cropped decoded picture in units of luma samples, as specified by Equations D‑28 and D‑29.

The value of ar\_bounding\_box\_left[ ar\_object\_idx[ i ] ] shall be in the range of 0 to croppedWidth / SubWidthC − 1, inclusive.

The value of ar\_bounding\_box\_top[ ar\_object\_idx[ i ] ] shall be in the range of 0 to croppedHeight / SubHeightC − 1, inclusive.

The value of ar\_bounding\_box\_width[ ar\_object\_idx[ i ] ] shall be in the range of 0 to croppedWidth / SubWidthtC − ar\_bounding\_box\_left[ ar\_object\_idx[ i ] ], inclusive.

The value of ar\_bounding\_box\_height[ ar\_object\_idx[ i ] ] shall be in the range of 0 to croppedHeight / SubHeightC − ar\_bounding\_box\_top[ ar\_object\_idx[ i ] ], inclusive.

The identified object rectangle contains the luma samples with horizontal picture coordinates from SubWidthC \* ( conf\_win\_left\_offset + ar\_bounding\_box\_left[ ar\_object\_idx[ i ] ] ) to SubWidthC \* ( conf\_win\_left\_offset + ar\_bounding\_box\_left[ ar\_object\_idx[ i ] ] + ar\_bounding\_box\_width[ ar\_object\_idx[ i ] ] ) − 1, inclusive, and vertical picture coordinates from SubHeightC \* ( conf\_win\_top\_offset + ar\_bounding\_box\_top[ ar\_object\_idx[ i ] ] ) to SubHeightC \* ( conf\_win\_top\_offset + ar\_bounding\_box\_top[ ar\_object\_idx[ i ] ] + ar\_bounding\_box\_height[ ar\_object\_idx[ i ] ] ) − 1, inclusive.

The values of ar\_bounding\_box\_top[ ar\_object\_idx[ i ] ], ar\_bounding\_box\_left[ ar\_object\_idx[ i ] ], ar\_bounding\_box\_width[ ar\_object\_idx[ i ] ] and ar\_bounding\_box\_height[ ar\_object\_idx[ i ] ] persist in output order within the CVS for each value of ar\_object\_idx[ i ]. When not present, the values of ar\_bounding\_box\_top[ ar\_object\_idx[ i ] ], ar\_bounding\_box\_left[ ar\_object\_idx[ i ] ], ar\_bounding\_box\_width[ ar\_object\_idx[ i ] ] or ar\_bounding\_box\_height[ ar\_object\_idx[ i ] ] are inferred from a previous annotated regions SEI message in output order in the CVS, if any.

**ar\_partial\_object\_flag**[ ar\_object\_idx[ i ] ] equal to 1 indicates that the ar\_bounding\_box\_top[ ar\_object\_idx[ i ] ], ar\_bounding\_box\_left[ ar\_object\_idx[ i ] ], ar\_bounding\_box\_width[ ar\_object\_idx[ i ] ] and ar\_bounding\_box\_height[ ar\_object\_idx[ i ] ] syntax elements represent the size and location of an object that is only partially visible within the cropped decoded picture. ar\_partial\_object\_flag[ ar\_object\_idx[ i ] ] equal to 0 indicates that the ar\_bounding\_box\_top[ ar\_object\_idx[ i ] ], ar\_bounding\_box\_left[ ar\_object\_idx[ i ] ], ar\_bounding\_box\_width[ ar\_object\_idx[ i ] ] and ar\_bounding\_box\_height[ ar\_object\_idx[ i ] ] syntax elements represent the size and location of an object that may or may not be only partially visible within the cropped decoded picture. When not present, the value of ar\_partial\_object\_flag[ ar\_object\_idx[ i ] ] is inferred from a previous annotated regions SEI message in output order in the CVS, if any.

**ar\_object\_confidence**[ ar\_object\_idx[ i ] ] indicates the degree of confidence associated with the ar\_object\_idx[ i ]-th object, in units of 2−( ar\_object\_confidence\_length\_minus1 + 1 ), such that a higher value of ar\_object\_confidence[ ar\_object\_idx[ i ] ] indicates a higher degree of confidence. The length of the ar\_object\_confidence[ ar\_object\_idx[ i ] ] syntax element is ar\_object\_confidence\_length\_minus1 + 1 bits. When not present, the value of\_object\_confidence[ ar\_object\_idx[ i ] ] is inferred from a previous annotated regions SEI message in output order in the CVS, if any.

**D.2.39 Shutter interval information SEI message semantics**

The shutter interval information SEI message indicates the shutter interval for the associated video source pictures prior to encoding and display, e.g., for camera-captured content, the shutter interval is the amount of time that an image sensor is exposed to produce each source picture.

**sii\_sub\_layer\_idx** specifies the shutter interval temporal sub-layer index of the current picture. The value of sii\_sub\_layer\_idx shall be equal to 0 when the current access unit is the first access unit of the CVS. When fixed\_shutter\_interval\_within\_cvs\_flag is equal to 1, the value of sii\_sub\_layer\_idx shall be equal to 0. Otherwise, fixed\_shutter\_interval\_within\_cvs\_flag is equal to 0, the value of sii\_sub\_layer\_idx shall be less than or equal to the value of sii\_max\_sub\_layers\_minus1.

**shutter\_interval\_info\_present\_flag** equal to 1 indicates that the syntax elements sii\_time\_scale, fixed\_shutter\_interval\_within\_cvs\_flag, and either sii\_num\_units\_in\_shutter\_interval or sii\_max\_sub\_layers\_minus1 and sub\_layer\_num\_units\_in\_shutter\_interval[ i ] are present. shutter\_interval\_info\_present\_flag equal to 0 indicates that the syntax elements sii\_time\_scale, fixed\_shutter\_interval\_within\_cvs\_flag, sii\_num\_units\_in\_shutter\_interval, sii\_max\_sub\_layers\_minus1, and sub\_layer\_num\_units\_in\_shutter\_interval[ i ] are not present. The value of shutter\_interval\_info\_present\_flag shall be equal to 1 when the current access unit is the first access unit of the CVS. Otherwise, the current access unit is not the first access unit of the CVS, the value of shutter\_interval\_info\_present\_flag shall be equal to 0.

**sii\_time\_scale** specifies the number of time units that pass in one second. The value of sii\_time\_scale shall be greater than 0. For example, a time coordinate system that measures time using a 27 MHz clock has an sii\_time\_scale of 27 000 000.

**fixed\_shutter\_interval\_within\_cvs\_flag** equal to 1 specifies that the indicated shutter interval is the same for all pictures in the CVS. fixed\_shutter\_interval\_within\_cvs\_flagequal to 0 specifies that the indicated shutter interval may not be the same for all pictures in the CVS.

**sii\_num\_units\_in\_shutter\_interval**, when fixed\_shutter\_interval\_within\_cvs\_flag is equal to 1, specifies the number of time units of a clock operating at the frequency sii\_time\_scale Hz that corresponds to the indicated shutter interval of each picture in the CVS. The value 0 may be used to indicate that the associated video content contains screen capture content, computer generated content, or other non-camera-captured content.

The indicated shutter interval, denoted by the variable shutterInterval, in units of seconds, is equal to the quotient of sii\_num\_units\_in\_shutter\_interval divided by sii\_time\_scale. For example, to represent a shutter interval equal to 0.04 seconds, sii\_time\_scale may be equal to 27 000 000 and sii\_num\_units\_in\_shutter\_interval may be equal to 1 080 000.

**sii\_max\_sub\_layers\_minus1** plus 1 specifies the maximum number of shutter interval temporal sub-layers indexes that may be present in the CVS.

**sub\_layer\_num\_units\_in\_shutter\_interval**[ i ], when present, specifies the number of time units of a clock operating at the frequency sii\_time\_scale Hz that corresponds to the shutter interval of each picture in the CVS for which the value of sii\_sub\_layer\_idx is equal to i. The sub-layer shutter interval for each picture for which the value of sii\_sub\_layer\_idx is equal to i, denoted by the variable subLayerShutterInterval[ i ], in units of seconds, is equal to the quotient of sub\_layer\_num\_units\_in\_shutter\_interval[ i ] divided by sii\_time\_scale.

The variable subLayerShutterInterval[ i ], corresponding to the indicated shutter interval of each picture in the sub-layer representation with TemporalId equal to i in the CVS, is thus derived as follows:

if( fixed\_shutter\_interval\_within\_cvs\_flag )  
 subLayerShutterInterval[ i ] = sii\_num\_units\_in\_shutter\_interval ÷ sii\_time\_scale (D.X)  
else  
 subLayerShutterInterval[ i ] = sub\_layer\_num\_units\_in\_shutter\_interval[ i ] ÷ sii\_time\_scale

When a shutter interval information SEI message is present for any access unit in a CVS, a shutter interval information SEI message shall be present for the IDR access unit that is the first access unit of the CVS. All shutter interval information SEI messages that apply to the same access unit shall have the same content.

sii\_time\_scale and fixed\_shutter\_interval\_within\_cvs\_flag persist from the first access unit of the CVS until a new CVS begins or the bitstream ends.

When the value of fixed\_shutter\_interval\_within\_cvs\_flag is equal to 0, a shutter interval information SEI message shall be present for every picture in the CVS. When present, sii\_num\_units\_in\_shutter\_interval, sii\_max\_sub\_layers\_minus1, and sub\_layer\_num\_units\_in\_shutter\_interval[ i ], persist from the first access unit of the CVS until a new CVS begins or the bitstream ends.

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