1. * + - 1. **Sequence parameter set data syntax**

|  |  |  |
| --- | --- | --- |
| seq\_parameter\_set\_data( ) { | **C** | **Descriptor** |
| **profile\_idc** | 0 | u(8) |
| **constraint\_set0\_flag** | 0 | u(1) |
| **constraint\_set1\_flag** | 0 | u(1) |
| **constraint\_set2\_flag** | 0 | u(1) |
| **constraint\_set3\_flag** | 0 | u(1) |
| **constraint\_set4\_flag** | 0 | u(1) |
| **constraint\_set5\_flag** | 0 | u(1) |
| **reserved\_zero\_2bits** /\* equal to 0 \***/** | 0 | u(2) |
| **level\_idc** | 0 | u(8) |
| **seq\_parameter\_set\_id** | 0 | ue(v) |
| if( profile\_idc = = 100 | | profile\_idc = = 110 | |  profile\_idc = = 122 | | profile\_idc = = 244 | | profile\_idc = = 44 | |  profile\_idc = = 83 | | profile\_idc = = 86 | | profile\_idc = = 118 | |  profile\_idc = = 128 | | profile\_idc = = 138 | | profile\_idc = = 139 | |  profile\_idc = = 134 | | profile\_idc = = 135 ) { |  |  |
| **chroma\_format\_idc** | 0 | ue(v) |
| if( chroma\_format\_idc = = 3 ) |  |  |
| **separate\_colour\_plane\_flag** | 0 | u(1) |
| **bit\_depth\_luma\_minus8** | 0 | ue(v) |
| **bit\_depth\_chroma\_minus8** | 0 | ue(v) |
| **qpprime\_y\_zero\_transform\_bypass\_flag** | 0 | u(1) |
| **seq\_scaling\_matrix\_present\_flag** | 0 | u(1) |
| if( seq\_scaling\_matrix\_present\_flag ) |  |  |
| for( i = 0; i < ( ( chroma\_format\_idc  !=  3 ) ? 8 : 12 ); i++ ) { |  |  |
| **seq\_scaling\_list\_present\_flag[** i **]** | 0 | u(1) |
| if( seq\_scaling\_list\_present\_flag[ i ] ) |  |  |
| if( i < 6 ) |  |  |
| scaling\_list( ScalingList4x4[ i ], 16,   UseDefaultScalingMatrix4x4Flag[ i ]) | 0 |  |
| else |  |  |
| scaling\_list( ScalingList8x8[ i − 6 ], 64,  UseDefaultScalingMatrix8x8Flag[ i − 6 ] ) | 0 |  |
| } |  |  |
| } |  |  |
| **log2\_max\_frame\_num\_minus4** | 0 | ue(v) |
| **pic\_order\_cnt\_type** | 0 | ue(v) |
| if( pic\_order\_cnt\_type = = 0 ) |  |  |
| **log2\_max\_pic\_order\_cnt\_lsb\_minus4** | 0 | ue(v) |
| else if( pic\_order\_cnt\_type = = 1 ) { |  |  |
| **delta\_pic\_order\_always\_zero\_flag** | 0 | u(1) |
| **offset\_for\_non\_ref\_pic** | 0 | se(v) |
| **offset\_for\_top\_to\_bottom\_field** | 0 | se(v) |
| **num\_ref\_frames\_in\_pic\_order\_cnt\_cycle** | 0 | ue(v) |
| for( i = 0; i < num\_ref\_frames\_in\_pic\_order\_cnt\_cycle; i++ ) |  |  |
| **offset\_for\_ref\_frame[** i **]** | 0 | se(v) |
| } |  |  |
| **max\_num\_ref\_frames** | 0 | ue(v) |
| **gaps\_in\_frame\_num\_value\_allowed\_flag** | 0 | u(1) |
| **pic\_width\_in\_mbs\_minus1** | 0 | ue(v) |
| **pic\_height\_in\_map\_units\_minus1** | 0 | ue(v) |
| **frame\_mbs\_only\_flag** | 0 | u(1) |
| if( !frame\_mbs\_only\_flag ) |  |  |
| **mb\_adaptive\_frame\_field\_flag** | 0 | u(1) |
| **direct\_8x8\_inference\_flag** | 0 | u(1) |
| **frame\_cropping\_flag** | 0 | u(1) |
| if( frame\_cropping\_flag ) { |  |  |
| **frame\_crop\_left\_offset** | 0 | ue(v) |
| **frame\_crop\_right\_offset** | 0 | ue(v) |
| **frame\_crop\_top\_offset** | 0 | ue(v) |
| **frame\_crop\_bottom\_offset** | 0 | ue(v) |
| } |  |  |
| **vui\_parameters\_present\_flag** | 0 | u(1) |
| if( vui\_parameters\_present\_flag ) |  |  |
| vui\_parameters( ) | 0 |  |
| } |  |  |

##### Subset sequence parameter set RBSP syntax

|  |  |  |
| --- | --- | --- |
| subset\_seq\_parameter\_set\_rbsp( ) { | **C** | **Descriptor** |
| seq\_parameter\_set\_data( ) | 0 |  |
| if( profile\_idc = = 83 | | profile\_idc = = 86 ) { |  |  |
| seq\_parameter\_set\_svc\_extension( ) /\* specified in Annex G \*/ | 0 |  |
| **svc\_vui\_parameters\_present\_flag** | 0 | u(1) |
| if( svc\_vui\_parameters\_present\_flag = = 1 ) |  |  |
| svc\_vui\_parameters\_extension( ) /\* specified in Annex G \*/ | 0 |  |
| } else if( profile\_idc = = 118 | | profile\_idc = = 128 | |  profile\_idc = = 134 ) { |  |  |
| **bit\_equal\_to\_one** /\* equal to 1 \*/ | 0 | f(1) |
| seq\_parameter\_set\_mvc\_extension( ) /\* specified in Annex H \*/ | 0 |  |
| **mvc\_vui\_parameters\_present\_flag** | 0 | u(1) |
| if( mvc\_vui\_parameters\_present\_flag = = 1 ) |  |  |
| mvc\_vui\_parameters\_extension( ) /\* specified in Annex H \*/ | 0 |  |
| } else if( profile\_idc = = 138 | | profile\_idc = = 135) { |  |  |
| **bit\_equal\_to\_one** /\* equal to 1 \*/ | 0 | f(1) |
| seq\_parameter\_set\_mvcd\_extension( ) /\* specified in Annex I \*/ |  |  |
| } else if( profile\_idc = = 139 ) { |  |  |
| **bit\_equal\_to\_one** /\* equal to 1 \*/ | 0 | f(1) |
| seq\_parameter\_set\_mvcd\_extension( ) /\* specified in Annex I \*/ |  |  |
| seq\_parameter\_set\_3dvc\_extension( ) /\* specified in Annex J \*/ | 0 |  |
| } |  |  |
| **additional\_extension2\_flag** | 0 | u(1) |
| if( additional\_extension2\_flag = = 1 ) |  |  |
| while( more\_rbsp\_data( ) ) |  |  |
| **additional\_extension2\_data\_flag** | 0 | u(1) |
| rbsp\_trailing\_bits( ) | 0 |  |
| **}** |  |  |

1. Annex I  
     
   Multiview and depth video coding

(This annex forms an integral part of this Recommendation | International Standard.)

This annex specifies multiview video coding with depth information, referred to as MVCD.

* 1. Scope

Bitstreams and decoders conforming to the profile specified in this annex are completely specified in this annex with reference made to clauses 2-9 and Annexes A-H.

* 1. Normative references

The specifications in clause 2 apply.

* 1. Definitions

For the purpose of this annex, the following definitions apply in addition to the definitions in clause H.3. These definitions are either not present in clause H.3 or replace definitions in clause H.3.

1. **depth field view**: A *depth view component* of a *field*.
2. **depth frame view**: A *depth view component* of a *frame*.
3. **depth view**: A sequence of *depth view components* associated with an identical value of view\_id.
4. **depth view component**: A *coded representation* of the depth of a view in a single *access unit*.
5. **inter-view only reference component**: A *view component*, *texture view component, or depth view component* coded with nal\_ref\_idc equal to 0 and inter\_view\_flag equal to 1. An *inter-view only reference component* contains samples that may be used for *inter-view prediction* in the *decoding process* of subsequent *view components* in *decoding order*, but are not used for *inter prediction* by any *view components*. *Inter-view only reference components* are *non-reference pictures*.
6. **inter-view reference component**: A *view component*, *texture view component, or depth view component* coded with nal\_ref\_idc greater than 0 and inter\_view\_flag equal to 1. An *inter-view reference component* contains samples that may be used for *inter prediction* of subsequent *pictures* in *decoding order* and *inter-view prediction* of subsequent *view components* in *decoding order*. *Inter-view reference components* are *reference pictures*.
7. **MVCD operation point**: An operation point for which each target output view includes a texture view or a depth view or both a texture view and a depth view.
8. **MVCD sequence parameter set**: A collective term for *sequence parameter set* or *subset sequence parameter set*.
9. **MVCD sequence parameter set RBSP**: A collective term for *sequence parameter set RBSP* or *subset sequence parameter set RBSP*.
10. **reference picture**: A *view component*, *texture view component, or depth view component* coded with nal\_ref\_idc greater than 0. A *reference picture* contains samples that may be used for *inter prediction* in the *decoding process* of subsequent *view components* in *decoding order*. A *reference picture* may be an *inter-view reference component*, in which case the samples contained in the *reference picture* may also be used for *inter-view prediction* in the *decoding process* of subsequent *view components* in *decoding order*.
11. **stereoscopic texture bitstream**: A *bitstream* containing two *texture* *views* and conforming to one of the *profiles* specified in Annex H.
12. **texture field view component**: A *texture view component* of a *field*.
13. **texture frame view component**: A *texture view component* of a *frame*.
14. **texture view**:A sequence of*texture view components* associated with an identical value of view\_id.
15. **texture view component**:A *coded representation* of the texture of a view in a single *access unit*.
16. **view**: A *texture view* and a *depth view* with the same value of view\_id, unless explicitly limited to either *texture view* or *depth view*.
17. **view component**: A *coded representation* of a *view* in a single *access unit*. A *view component* may consist of a *texture view component* and a *depth view component*.
18. **view component pair**: A *texture view component* and a *depth view component* of the same *view* within the same *access unit*.
    1. Abbreviations

The specifications in clause 4 apply.

* 1. Conventions

The specifications in clause 5 apply.

* 1. Source, coded, decoded and output data formats, scanning processes, and neighbouring relationships

The specifications in clause 6 apply with substitution of MVCD sequence parameter set for sequence parameter set.

* 1. Syntax and semantics

This clause specifies syntax and semantics for coded video sequences that conform to one or more of the profiles specified in this annex.

* + 1. Method of specifying syntax in tabular form

The specifications in clause H.7.1 apply.

* + 1. Specification of syntax functions, categories, and descriptors

The specifications in clause H.7.2 apply.

* + 1. Syntax in tabular form
       1. NAL unit syntax

The syntax table is specified in clause H.7.3.1.

* + - * 1. NAL unit header MVC extension syntax

The syntax table is specified in clause H.7.3.1.1.

* + - 1. Raw byte sequence payloads and RBSP trailing bits syntax
         1. Sequence parameter set RBSP syntax

The syntax table is specified in clause H.7.3.2.1.

Sequence parameter set data syntax

The syntax table is specified in clause H.7.3.2.1.1.

Scaling list syntax

The syntax table is specified in clause H.7.3.2.1.1.1.

Sequence parameter set extension RBSP syntax

The syntax table is specified in clause H.7.3.2.1.2.

Subset sequence parameter set RBSP syntax

The syntax table is specified in clause H.7.3.2.1.3.

Sequence parameter set MVC extension syntax

The syntax table is specified in clause H.7.3.2.1.4.

Sequence parameter set MVCD extension syntax

|  |  |  |
| --- | --- | --- |
| seq\_parameter\_set\_mvcd\_extension( ) { | **C** | **Descriptor** |
| **num\_views\_minus1** | 0 | ue(v) |
| for( i = 0, NumDepthViews = 0; i <= num\_views\_minus1; i++ ) { |  |  |
| **view\_id[**i**]** | 0 | ue(v) |
| **depth\_view\_present\_flag[**i**]** | 0 | u(1) |
| DepthViewId[ NumDepthViews ] = view\_id[ i ] |  |  |
| NumDepthViews += depth\_view\_present\_flag[ i ] |  |  |
| **texture\_view\_present\_flag[**i**]** | 0 | u(1) |
| } |  |  |
| for( i = 1; i <= num\_views\_minus1; i++ ) |  |  |
| if( depth\_view\_present\_flag[ i ] ) { |  |  |
| **num\_anchor\_refs\_l0[**i**]** | 0 | ue(v) |
| for( j = 0; j < num\_anchor\_refs\_l0[ i ]; j++ ) |  |  |
| **anchor\_ref\_l0[**i**][**j**]** | 0 | ue(v) |
| **num\_anchor\_refs\_l1[**i**]** | 0 | ue(v) |
| for( j = 0; j < num\_anchor\_refs\_l1[ i ]; j++ ) |  |  |
| **anchor\_ref\_l1[**i**][**j**]** | 0 | ue(v) |
| } |  |  |
| for( i = 1; i <= num\_views\_minus1; i++ ) |  |  |
| if( depth\_view\_present\_flag[ i ] ) { |  |  |
| **num\_non\_anchor\_refs\_l0[**i**]** | 0 | ue(v) |
| for( j = 0; j < num\_non\_anchor\_refs\_l0[ i ]; j++ ) |  |  |
| **non\_anchor\_ref\_l0[**i**][**j**]** | 0 | ue(v) |
| **num\_non\_anchor\_refs\_l1[**i**]** | 0 | ue(v) |
| for( j = 0; j < num\_non\_anchor\_refs\_l1[ i ]; j++ ) |  |  |
| **non\_anchor\_ref\_l1[**i**][**j**]** | 0 | ue(v) |
| } |  |  |
| **num\_level\_values\_signalled\_minus1** | 0 | ue(v) |
| for( i = 0; i <= num\_level\_values\_signalled\_minus1; i++ ) { |  |  |
| **level\_idc[**i**]** | 0 | u(8) |
| **num\_applicable\_ops\_minus1[**i**]** | 0 | ue(v) |
| for( j = 0; j <= num\_applicable\_ops\_minus1[ i ]; j++ ) { |  |  |
| **applicable\_op\_temporal\_id[**i**][**j**]** | 0 | u(3) |
| **applicable\_op\_num\_target\_views\_minus1[**i**][**j**]** | 0 | ue(v) |
| for( k = 0; k <= applicable\_op\_num\_target\_views\_minus1[ i ][ j ];  k++ ) { |  |  |
| **applicable\_op\_target\_view\_id[**i**][**j**][**k**]** | 0 | ue(v) |
| **applicable\_op\_depth\_flag[**i**][**j**][**k**]** | 0 | u(1) |
| **applicable\_op\_texture\_flag[**i**][**j**][**k**]** | 0 | u(1) |
| } |  |  |
| **applicable\_op\_num\_texture\_views\_minus1[**i**][**j**]** | 0 | ue(v) |
| **applicable\_op\_num\_depth\_views[**i**][**j**]** | 0 | ue(v) |
| } |  |  |
| } |  |  |
| **mvcd\_vui\_parameters\_present\_flag** | 0 | u(1) |
| if( mvcd\_vui\_parameters\_present\_flag = = 1 ) |  |  |
| mvcd\_vui\_parameters\_extension( ) |  |  |
| **texture\_vui\_parameters\_present\_flag** | 0 | u(1) |
| if( texture\_vui\_parameters\_present\_flag = = 1 ) |  |  |
| mvc\_vui\_parameters\_extension( ) | 0 |  |
| } |  |  |

* + - * 1. Picture parameter set RBSP syntax

The syntax table is specified in clause H.7.3.2.2.

* + - * 1. Supplemental enhancement information RBSP syntax

The syntax table is specified in clause H.7.3.2.3.

Supplemental enhancement information message syntax

The syntax table is specified in clause H.7.3.2.3.1.

* + - * 1. Access unit delimiter RBSP syntax

The syntax table is specified in clause H.7.3.2.4.

* + - * 1. End of sequence RBSP syntax

The syntax table is specified in clause H.7.3.2.5.

* + - * 1. End of stream RBSP syntax

The syntax table is specified in clause H.7.3.2.6.

* + - * 1. Filler data RBSP syntax

The syntax table is specified in clause H.7.3.2.7.

* + - * 1. Slice layer without partitioning RBSP syntax

The syntax table is specified in clause H.7.3.2.8.

* + - * 1. Slice data partition RBSP syntax

Slice data partition syntax is not present in coded video sequences conforming to one or more of the profiles specified in this annex.

* + - * 1. RBSP slice trailing bits syntax

The syntax table is specified in clause H.7.3.2.10.

* + - * 1. RBSP trailing bits syntax

The syntax table is specified in clause H.7.3.2.11.

* + - * 1. Prefix NAL unit RBSP syntax

The syntax table is specified in clause H.7.3.2.12.

* + - * 1. Slice layer extension RBSP syntax

The syntax table is specified in clause H.7.3.2.13.

* + - 1. Slice header syntax

The syntax table is specified in clause H.7.3.3.

* + - * 1. Reference picture list modification syntax

The syntax table is specified in clause H.7.3.3.1.

Reference picture list MVC modification syntax

The syntax table is specified in clause H.7.3.3.1.1

* + - * 1. Prediction weight table syntax

The syntax table is specified in clause H.7.3.3.2.

* + - * 1. Decoded reference picture marking syntax

The syntax table is specified in clause H.7.3.3.3.

* + - 1. Slice data syntax

The syntax table is specified in clause H.7.3.4.

* + - 1. Macroblock layer syntax

The syntax table is specified in clause H.7.3.5.

* + - * 1. Macroblock prediction syntax

The syntax table is specified in clause H.7.3.5.1.

* + - * 1. Sub-macroblock prediction syntax

The syntax table is specified in clause H.7.3.5.2.

* + - * 1. Residual data syntax

The syntax table is specified in clause H.7.3.5.3.

Residual luma syntax

The syntax table is specified in clause H.7.3.5.3.1.

Residual block CAVLC syntax

The syntax table is specified in clause H.7.3.5.3.2.

Residual block CABAC syntax

The syntax table is specified in clause H.7.3.5.3.3.

* + 1. Semantics

Semantics associated with the syntax structures and syntax elements within these structures (in clause I.7.3 and in clause H.7.3 by reference in clause I.7.3) are specified in this clause and by reference to clause I.7.4. When the semantics of a syntax element are specified using a table or a set of tables, any values that are not specified in the table(s) shall not be present in the bitstream unless otherwise specified in this Recommendation | International Standard.

* + - 1. NAL unit semantics

The semantics for the syntax elements in clause I.7.3.1 are specified in clause H.7.3.1.

* + - * 1. NAL unit header MVC extension semantics

The semantics for the syntax elements in clause I.7.3.1.1 are specified in clause H.7.3.1.1.

* + - * 1. Order of NAL units and association to coded pictures, access units, and video sequences

This clause specifies constraints on the order of NAL units in the bitstream. Any order of NAL units in the bitstream obeying these constraints is referred to in the text as the decoding order of NAL units. Within a NAL unit, the syntax in clauses 7.3, D.1, E.1, H.7.3, H.13.1, H.14.1, I.13.1 and I.14.1 specifies the decoding order of syntax elements. Decoders shall be capable of receiving NAL units and their syntax elements in decoding order.

Order of MVCD sequence parameter set RBSPs and picture parameter set RBSPs and their activation

NOTE 1 – The sequence and picture parameter set mechanism decouples the transmission of infrequently changing information from the transmission of coded macroblock data. Sequence and picture parameter sets may, in some applications, be conveyed "out-of-band" using a reliable transport mechanism.

A picture parameter set RBSP includes parameters that can be referred to by the coded slice NAL units of one or more texture view or depth view components of one or more coded pictures.

Each picture parameter set RBSP is initially considered not active at the start of the operation of the decoding process. At most one picture parameter set RBSP is considered as the active picture parameter set RBSP at any given moment during the operation of the decoding process, and when any particular picture parameter set RBSP becomes the active picture parameter set RBSP, the previously-active picture parameter set RBSP (if any) is deactivated.

In addition to the active picture parameter set RBSP, zero or more picture parameter set RBSPs may be specifically active for texture view components (with a particular value of VOIdx less than or equal to VOIdxMax) that belong to the target output views or that may be referred to through inter-view prediction in decoding texture view components belonging to the target output views. Such a picture parameter set RBSP is referred to as the active texture picture parameter set RBSP for the particular value of VOIdx. The restrictions on active picture parameter set RBSPs also apply to active texture picture parameter set RBSPs for a particular value of VOIdx.

Furthermore, zero or more picture parameter set RBSPs may be specifically active for depth view components (with a particular value of VOIdx less than VOIdxMax) that belong to the target output views or that may be referred to through inter-view prediction in decoding depth view components belonging to the target output views. Such a picture parameter set RBSP is referred to as the active depth picture parameter set RBSP for the particular value of VOIdx. The restrictions on active picture parameter set RBSPs also apply to active depth picture parameter set RBSPs for a particular value of VOIdx less than VOIdxMax.

When a picture parameter set RBSP (with a particular value of pic\_parameter\_set\_id) is not the active picture parameter set RBSP and it is referred to by a coded slice NAL unit belonging to a depth view component (i.e., with nal\_unit\_type equal to 21) and with VOIdx equal to VOIdxMax (using that value of pic\_parameter\_set\_id), it is activated. This picture parameter set RBSP is called the active picture parameter set RBSP until it is deactivated when another picture parameter set RBSP becomes the active picture parameter set RBSP. A picture parameter set RBSP, with that particular value of pic\_parameter\_set\_id, shall be available to the decoding process prior to its activation.

When a picture parameter set RBSP (with a particular value of pic\_parameter\_set\_id) is not the active depth picture parameter set for a particular value of VOIdx less than VOIdxMax and it is referred to by a coded slice NAL unit belonging to a depth view component (i.e., with nal\_unit\_type equal to 21) and with the particular value of VOIdx (using that value of pic\_parameter\_set\_id), it is activated for view components with the particular value of VOIdx. This picture parameter set RBSP is called the active depth picture parameter set RBSP for the particular value of VOIdx until it is deactivated when another picture parameter set RBSP becomes the active depth picture parameter set RBSP for the particular value of VOIdx. A picture parameter set RBSP, with that particular value of pic\_parameter\_set\_id, shall be available to the decoding process prior to its activation.

When a picture parameter set RBSP (with a particular value of pic\_parameter\_set\_id) is not the active texture picture parameter set for a particular value of VOIdx less than or equal to VOIdxMax and it is referred to by a coded slice NAL unit belonging to a texture view component (i.e., with nal\_unit\_type equal to 1, 5 or 20) and with the particular value of VOIdx (using that value of pic\_parameter\_set\_id), it is activated for texture view components with the particular value of VOIdx. This picture parameter set RBSP is called the active texture picture parameter set RBSP for the particular value of VOIdx until it is deactivated when another picture parameter set RBSP becomes the active texture picture parameter set RBSP for the particular value of VOIdx. A picture parameter set RBSP, with that particular value of pic\_parameter\_set\_id, shall be available to the decoding process prior to its activation.

Any picture parameter set NAL unit containing the value of pic\_parameter\_set\_id for the active picture parameter set RBSP for a coded picture shall have the same content as that of the active picture parameter set RBSP for this coded picture unless it follows the last VCL NAL unit of this coded picture and precedes the first VCL NAL unit of another coded picture. Any picture parameter set NAL unit containing the value of pic\_parameter\_set\_id for the active depth picture parameter set RBSP for a particular value of VOIdx less than VOIdxMax for a coded picture shall have the same content as that of the active view picture parameter set RBSP for the particular value of VOIdx for this coded picture unless it follows the last VCL NAL unit of this coded picture and precedes the first VCL NAL unit of another coded picture. Any picture parameter set NAL unit containing the value of pic\_parameter\_set\_id for the active texture picture parameter set RBSP for a particular value of VOIdx for a coded picture shall have the same content as that of the active texture picture parameter set RBSP for the particular value of VOIdx for this coded picture unless it follows the last VCL NAL unit of this coded picture and precedes the first VCL NAL unit of another coded picture.

A MVCD sequence parameter set RBSP includes parameters that can be referred to by one or more picture parameter set RBSPs or one or more buffering period SEI messages.

Each MVCD sequence parameter set RBSP is initially considered not active at the start of the operation of the decoding process. At most one MVCD sequence parameter set RBSP is considered as the active MVCD sequence parameter set RBSP at any given moment during the operation of the decoding process, and when any particular MVCD sequence parameter set RBSP becomes the active MVCD sequence parameter set RBSP, the previously-active MVCD sequence parameter set RBSP (if any) is deactivated.

In addition to the active MVCD sequence parameter set RBSP, zero or more MVCD sequence parameter set RBSPs may be specifically active for view components (with a particular value of VOIdx less than VOIdxMax) that belong to the target output views or that may be referred to through inter-view prediction in decoding view components belonging to the target output views. Such a MVCD sequence parameter set RBSP is referred to as the active view MVCD sequence parameter set RBSP for the particular value of VOIdx. The restrictions on active MVCD sequence parameter set RBSPs also apply to active view MVCD sequence parameter set RBSPs for a particular value of VOIdx less than VOIdxMax.

Furthermore, zero or more MVCD sequence parameter set RBSPs may be specifically active for texture view components (with a particular value of VOIdx less than or equal to VOIdxMax) that belong to the target output views or that may be referred to through inter-view prediction in decoding texture view components belonging to the target output views. Such a MVCD sequence parameter set RBSP is referred to as the active texture MVCD sequence parameter set RBSP for the particular value of VOIdx. The restrictions on active MVCD sequence parameter set RBSPs also apply to active texture MVCD sequence parameter set RBSPs for a particular value of VOIdx.

For the following specification, the activating buffering period SEI message is specified as follows.

– If VOIdxMax is equal to VOIdxMin and the access unit contains a buffering period SEI message not included in an MVC scalable nesting SEI message and not included in a MVCD scalable nesting SEI message, this buffering period SEI message is the activating buffering period SEI message.

– Otherwise if VOIdxMax is not equal to VOIdxMin and the access unit contains a buffering period SEI message included in a MVCD scalable nesting SEI message and associated with the operation point being decoded, this buffering period SEI message is the activating buffering period SEI message.

– Otherwise, the access unit does not contain an activating buffering period SEI message.

When a sequence parameter set RBSP (nal\_unit\_type is equal to 7) with a particular value of seq\_parameter\_set\_id is not already the active MVCD sequence parameter set RBSP and it is referred to by activation of a picture parameter set RBSP (using that value of seq\_parameter\_set\_id) and the picture parameter set RBSP is activated by a coded slice NAL unit with nal\_unit\_type equal to 1 or 5 (the picture parameter set RBSP becomes the active picture parameter set RBSP and VOIdxMax is equal to VOIdxMin and there is no depth view component in any access unit) and the access unit does not contain an activating buffering period SEI message, it is activated. This sequence parameter set RBSP is called the active MVCD sequence parameter set RBSP until it is deactivated when another MVCD sequence parameter set RBSP becomes the active MVCD sequence parameter set RBSP. A sequence parameter set RBSP, with that particular value of seq\_parameter\_set\_id, shall be available to the decoding process prior to its activation.

When a sequence parameter set RBSP (nal\_unit\_type is equal to 7) with a particular value of seq\_parameter\_set\_id is not already the active MVCD sequence parameter set RBSP and it is referred to by an activating buffering period SEI message (using that value of seq\_parameter\_set\_id) that is not included in a MVCD scalable nesting SEI message and VOIdxMax is equal to VOIdxMin and there is no depth view component in the access unit, it is activated. This sequence parameter set RBSP is called the active MVCD sequence parameter set RBSP until it is deactivated when another MVCD sequence parameter set RBSP becomes the active MVCD sequence parameter set RBSP. A sequence parameter set RBSP, with that particular value of seq\_parameter\_set\_id, shall be available to the decoding process prior to its activation.

When a subset sequence parameter set RBSP (nal\_unit\_type is equal to 15) with a particular value of seq\_parameter\_set\_id is not already the active MVCD sequence parameter set RBSP and it is referred to by activation of a picture parameter set RBSP (using that value of seq\_parameter\_set\_id) and the picture parameter set RBSP is activated by a coded slice depth extension NAL unit with nal\_unit\_type equal to 21 and with VOIdx equal to VOIdxMax (the picture parameter set RBSP becomes the active picture parameter set RBSP) and the access unit does not contain an activating buffering period SEI message, it is activated. This subset sequence parameter set RBSP is called the active MVCD sequence parameter set RBSP until it is deactivated when another MVCD sequence parameter set RBSP becomes the active MVCD sequence parameter set RBSP. A subset sequence parameter set RBSP, with that particular value of seq\_parameter\_set\_id, shall be available to the decoding process prior to its activation.

When a subset sequence parameter set RBSP (nal\_unit\_type is equal to 15) with a particular value of seq\_parameter\_set\_id is not already the active MVCD sequence parameter set RBSP and it is referred to by an activating buffering period SEI message (using that value of seq\_parameter\_set\_id) that is included in a MVCD scalable nesting SEI message, it is activated. This subset sequence parameter set RBSP is called the active MVCD sequence parameter set RBSP until it is deactivated when another MVCD sequence parameter set RBSP becomes the active MVCD sequence parameter set RBSP. A subset sequence parameter set RBSP, with that particular value of seq\_parameter\_set\_id, shall be available to the decoding process prior to its activation.

NOTE 2 – The active MVCD sequence parameter set RBSP is either a sequence parameter set RBSP or a subset sequence parameter set RBSP. Sequence parameter set RBSPs are activated by coded slice NAL units with nal\_unit\_type equal to 1 or 5 or buffering period SEI messages that are not included in an MVC scalable nesting SEI message or a MVCD scalable nesting SEI message. Subset sequence parameter sets are activated by coded slice depth extension NAL units (nal\_unit\_type equal to 21) or buffering period SEI messages that are included in a MVCD scalable nesting SEI message. A sequence parameter set RBSP and a subset sequence parameter set RBSP may have the same value of seq\_parameter\_set\_id.

For the following specification, the activating texture buffering period SEI message for a particular value of VOIdx is specified as follows.

– If the access unit contains one or more than one buffering period SEI message included in an MVC scalable nesting SEI message and associated with an operation point for which the greatest VOIdx in the associated bitstream subset is equal to the particular value of VOIdx, the first of these buffering period SEI messages, in decoding order, is the activating texture buffering period SEI message for the particular value of VOIdx.

– Otherwise, if the access unit contains a buffering period SEI message not included in an MVC scalable nesting SEI message or a MVCD scalable nesting SEI message, this buffering period SEI message is the activating texture buffering period SEI message for the particular value of VOIdx equal to VOIdxMin.

– Otherwise, the access unit does not contain an activating texture buffering period SEI message for the particular value of VOIdx.

When a sequence parameter set RBSP (nal\_unit\_type is equal to 7) with a particular value of seq\_parameter\_set\_id is not already the active texture MVCD sequence parameter set RBSP for VOIdx equal to VOIdxMin and it is referred to by activation of a picture parameter set RBSP (using that value of seq\_parameter\_set\_id) and the picture parameter set RBSP is activated by a coded slice NAL unit with nal\_unit\_type equal to 1 or 5 (the picture parameter set RBSP becomes the active texture picture parameter set RBSP for VOIdx equal to VOIdxMin), it is activated for texture view components with VOIdx equal to VOIdxMin. This sequence parameter set RBSP is called the active texture MVCD sequence parameter set RBSP for VOIdx equal to VOIdxMin until it is deactivated when another MVCD sequence parameter set RBSP becomes the active texture MVCD sequence parameter set RBSP for VOIdx equal to VOIdxMin. A sequence parameter set RBSP, with that particular value of seq\_parameter\_set\_id, shall be available to the decoding process prior to its activation.

When a sequence parameter set RBSP (nal\_unit\_type is equal to 7) with a particular value of seq\_parameter\_set\_id is not already the active texture MVCD sequence parameter set RBSP for VOIdx equal to VOIdxMin and it is referred to by an activating texture buffering period SEI message (using that value of seq\_parameter\_set\_id) that is not included in an MVC scalable nesting SEI message or a MVCD scalable nesting SEI message, the sequence parameter set RBSP is activated for texture view components with VOIdx equal to VOIdxMin. This sequence parameter set RBSP is called the active texture MVCD sequence parameter set RBSP for VOIdx equal to VOIdxMin until it is deactivated when another MVCD sequence parameter set RBSP becomes the active texture MVCD sequence parameter set RBSP for VOIdx equal to. A sequence parameter set RBSP, with that particular value of seq\_parameter\_set\_id, shall be available to the decoding process prior to its activation.

When a subset sequence parameter set RBSP (nal\_unit\_type is equal to 15) with a particular value of seq\_parameter\_set\_id is not already the active texture MVCD sequence parameter set RBSP for a particular value of VOIdx less than or equal to VOIdxMax and it is referred to by activation of a picture parameter set RBSP (using that value of seq\_parameter\_set\_id) and the picture parameter set RBSP is activated by a coded slice MVC extension NAL unit (nal\_unit\_type equal to 20) with the particular value of VOIdx (the picture parameter set RBSP becomes the active texture picture parameter set RBSP for the particular value of VOIdx), it is activated for texture view components with the particular value of VOIdx. This subset sequence parameter set RBSP is called the active texture MVCD sequence parameter set RBSP for the particular value of VOIdx until it is deactivated when another MVCD sequence parameter set RBSP becomes the active texture MVCD sequence parameter set RBSP for the particular value of VOIdx. A subset sequence parameter set RBSP, with that particular value of seq\_parameter\_set\_id, shall be available to the decoding process prior to its activation.

When a subset sequence parameter set RBSP (nal\_unit\_type is equal to 15) with a particular value of seq\_parameter\_set\_id is not already the active texture MVCD sequence parameter set RBSP for a particular value of VOIdx less than or equal to VOIdxMax and it is referred to by an activating texture buffering period SEI message (using that value of seq\_parameter\_set\_id) that is included in an MVC scalable nesting SEI message and associated with the particular value of VOIdx, this subset sequence parameter set RBSP is activated for texture view components with the particular value of VOIdx. This subset sequence parameter set RBSP is called the active texture MVCD sequence parameter set RBSP for the particular value of VOIdx until it is deactivated when another MVCD sequence parameter set RBSP becomes the active texture MVCD sequence parameter set RBSP for the particular value of VOIdx. A subset sequence parameter set RBSP, with that particular value of seq\_parameter\_set\_id, shall be available to the decoding process prior to its activation.

For the following specification, the activating view buffering period SEI message for a particular value of VOIdx is specified as follows.

– If the access unit contains one or more than one buffering period SEI message included in a MVCD scalable nesting SEI message and associated with an operation point for which the greatest VOIdx in the associated bitstream subset is equal to the particular value of VOIdx, the first of these buffering period SEI messages, in decoding order, is the activating view buffering period SEI message for the particular value of VOIdx.

– Otherwise, the access unit does not contain an activating view buffering period SEI message for the particular value of VOIdx.

When a subset sequence parameter set RBSP (nal\_unit\_type is equal to 15) with a particular value of seq\_parameter\_set\_id is not already the active view MVCD sequence parameter set RBSP for a particular value of VOIdx less than VOIdxMax and it is referred to by activation of a picture parameter set RBSP (using that value of seq\_parameter\_set\_id) and the picture parameter set RBSP is activated by a coded slice NAL unit with nal\_unit\_type equal to 21 and with the particular value of VOIdx (the picture parameter set RBSP becomes the active view picture parameter set RBSP for the particular value of VOIdx), it is activated for view components with the particular value of VOIdx. This subset sequence parameter set RBSP is called the active view MVCD sequence parameter set RBSP for the particular value of VOIdx until it is deactivated when another MVCD sequence parameter set RBSP becomes the active view MVCD sequence parameter set RBSP for the particular value of VOIdx or when decoding an access unit with VOIdxMax less than or equal to the particular value of VOIdx. A subset sequence parameter set RBSP, with that particular value of seq\_parameter\_set\_id, shall be available to the decoding process prior to its activation.

When a subset sequence parameter set RBSP (nal\_unit\_type is equal to 15) with a particular value of seq\_parameter\_set\_id is not already the active view MVCD sequence parameter set RBSP for a particular value of VOIdx less than VOIdxMax and it is referred to by an activating view buffering period SEI message (using that value of seq\_parameter\_set\_id) that is included in a MVCD scalable nesting SEI message and associated with the particular value of VOIdx, this subset sequence parameter set RBSP is activated for view components with the particular value of VOIdx. This subset sequence parameter set RBSP is called the active view MVCD sequence parameter set RBSP for the particular value of VOIdx until it is deactivated when another MVCD sequence parameter set RBSP becomes the active view MVCD sequence parameter set RBSP for the particular value of VOIdx or when decoding an access unit with VOIdxMax less than or equal to the particular value of VOIdx. A subset sequence parameter set RBSP, with that particular value of seq\_parameter\_set\_id, shall be available to the decoding process prior to its activation.

A MVCD sequence parameter set RBSP that includes a value of profile\_idc not specified in Annex A or Annex H or Annex I shall not be referred to by activation of a picture parameter set RBSP as the active picture parameter set RBSP or as active view picture parameter set RBSP or as active texture picture parameter set RBSP (using that value of seq\_parameter\_set\_id) or referred to by a buffering period SEI message (using that value of seq\_parameter\_set\_id). A MVCD sequence parameter set RBSP including a value of profile\_idc not specified in Annex A or Annex H or Annex I is ignored in the decoding for profiles specified in Annex A or Annex H or Annex I.

It is a requirement of bitstream conformance that the following constraints are obeyed:

– For each particular value of VOIdx, all coded slice NAL units (with nal\_unit\_type equal to 1, 5, 20, or 21) of a coded video sequence shall refer to the same value of seq\_parameter\_set\_id (via the picture parameter set RBSP that is referred to by the value of pic\_parameter\_set\_id).

– The value of seq\_parameter\_set\_id in a buffering period SEI message that is not included in an MVC scalable nesting SEI message shall be identical to the value of seq\_parameter\_set\_id in the picture parameter set RBSP that is referred to by coded slice NAL units with nal\_unit\_type equal to 1 or 5 (via the value of pic\_parameter\_set\_id) in the same access unit.

– The value of seq\_parameter\_set\_id in a buffering period SEI message that is included in an MVC scalable nesting SEI message and is associated with a particular value of VOIdx shall be identical to the value of seq\_parameter\_set\_id in the picture parameter set RBSP that is referred to by coded slice NAL units with nal\_unit\_type equal to 1, 5 or 20 with the particular value of VOIdx (via the value of pic\_parameter\_set\_id) in the same access unit.

– The value of seq\_parameter\_set\_id in a buffering period SEI message that is included in a MVCD scalable nesting SEI message and is associated with a particular value of VOIdx shall be identical to the value of seq\_parameter\_set\_id in the picture parameter set RBSP that is referred to by coded slice NAL units with nal\_unit\_type equal to 21 with the particular value of VOIdx (via the value of pic\_parameter\_set\_id) in the same access unit.

The active view MVCD sequence parameter set RBSPs for different values of VOIdx may be the same MVCD sequence parameter set RBSP. The active MVCD sequence parameter set RBSP and an active view MVCD sequence parameter set RBSP for a particular value of VOIdx may be the same MVCD sequence parameter set RBSP.

The active texture MVCD sequence parameter set RBSPs for different values of VOIdx may be the same MVCD sequence parameter set RBSP. The active MVCD sequence parameter set RBSP and an active texture MVCD sequence parameter set RBSP for a particular value of VOIdx may be the same MVCD sequence parameter set RBSP..

When the active MVCD sequence parameter set RBSP for a coded picture is a sequence parameter set RBSP, any sequence parameter set RBSP in the coded video sequence containing this coded picture and with the value of seq\_parameter\_set\_id for the active MVCD sequence parameter set RBSP shall have the same content as that of the active MVCD sequence parameter set RBSP.

When the active MVCD sequence parameter set RBSP for a coded picture is a subset sequence parameter set RBSP, any subset sequence parameter set RBSP in the coded video sequence containing this coded picture and with the value of seq\_parameter\_set\_id for the active MVCD sequence parameter set RBSP shall have the same content as that of the active MVCD sequence parameter set RBSP.

For each particular value of VOIdx, the following applies:

– When the active texture MVCD sequence parameter set RBSP for a coded picture is a sequence parameter set RBSP, any sequence parameter set RBSP in the coded video sequence containing this coded picture and with the value of seq\_parameter\_set\_id for the active texture MVCD sequence parameter set RBSP shall have the same content as that of the active texture MVCD sequence parameter set RBSP.

– When the active texture MVCD sequence parameter set RBSP for a coded picture is a subset sequence parameter set RBSP, any subset sequence parameter set RBSP in the coded video sequence containing this coded picture and with the value of seq\_parameter\_set\_id for the active texture MVCD sequence parameter set RBSP shall have the same content as that of the active texture MVCD sequence parameter set RBSP.

– The active view MVCD sequence parameter set RBSP for a coded picture is a subset sequence parameter set RBSP, and any subset sequence parameter set RBSP in the coded video sequence containing this coded picture and with the value of seq\_parameter\_set\_id for the active view MVCD sequence parameter set RBSP shall have the same content as that of the active view MVCD sequence parameter set RBSP.

NOTE 3 – If picture parameter set RBSPs or MVCD sequence parameter set RBSPs are conveyed within the bitstream, these constraints impose an order constraint on the NAL units that contain the picture parameter set RBSPs or MVCD sequence parameter set RBSPs, respectively. Otherwise (picture parameter set RBSPs or MVCD sequence parameter set RBSPs are conveyed by other means not specified in this Recommendation | International Standard), they must be available to the decoding process in a timely fashion such that these constraints are obeyed.

When present, a sequence parameter set extension RBSP includes parameters having a similar function to those of a sequence parameter set RBSP. For purposes of establishing constraints on the syntax elements of the sequence parameter set extension RBSP and for purposes of determining activation of a sequence parameter set extension RBSP, the sequence parameter set extension RBSP shall be considered part of the preceding sequence parameter set RBSP with the same value of seq\_parameter\_set\_id. When a sequence parameter set RBSP is present that is not followed by a sequence parameter set extension RBSP with the same value of seq\_parameter\_set\_id prior to the activation of the sequence parameter set RBSP, the sequence parameter set extension RBSP and its syntax elements shall be considered not present for the active MVCD sequence parameter set RBSP. The contents of sequence parameter set extension RBSPs only apply when the base texture view, which conforms to one or more of the profiles specified in Annex A, of a coded video sequence conforming to one or more profiles specified in Annex I is decoded. Subset sequence parameter set RBSPs shall not be followed by a sequence parameter set extension RBSP.

NOTE 4 – Sequence parameter sets extension RBSPs are not considered to be part of a subset sequence parameter set RBSP and subset sequence parameter set RBSPs must not be followed by a sequence parameter set extension RBSP.

For view components with VOIdx equal to VOIdxMax, all constraints that are expressed on the relationship between the values of the syntax elements (and the values of variables derived from those syntax elements) in MVCD sequence parameter sets and picture parameter sets and other syntax elements are expressions of constraints that apply only to the active MVCD sequence parameter set and the active picture parameter set. For view components with a particular value of VOIdx less than VOIdxMax, all constraints that are expressed on the relationship between the values of the syntax elements (and the values of variables derived from those syntax elements) in MVCD sequence parameter sets and picture parameter sets and other syntax elements are expressions of constraints that apply only to the active view MVCD sequence parameter set and the active view picture parameter set for the particular value of VOIdx. If any MVCD sequence parameter set RBSP having profile\_idc equal to the value of one of the profile\_idc values specified in Annex A or Annex H or Annex I is present that is never activated in the bitstream (i.e., it never becomes the active MVCD sequence parameter set or an active view MVCD sequence parameter set), its syntax elements shall have values that would conform to the specified constraints if it were activated by reference in an otherwise-conforming bitstream. If any picture parameter set RBSP is present that is never activated in the bitstream (i.e., it never becomes the active picture parameter set or an active view picture parameter set), its syntax elements shall have values that would conform to the specified constraints if it were activated by reference in an otherwise-conforming bitstream.

During operation of the decoding process (see clause I.8), for view components with VOIdx equal to VOIdxMax, the values of parameters of the active picture parameter set and the active MVCD sequence parameter set shall be considered in effect. For view components with a particular value of VOIdx less than VOIdxMax, the values of the parameters of the active view picture parameter set and the active view MVCD sequence parameter set for the particular value of VOIdx shall be considered in effect. For interpretation of SEI messages that apply to the entire access unit or the view component with VOIdx equal to VOIdxMax, the values of the parameters of the active picture parameter set and the active MVCD sequence parameter set for the same access unit shall be considered in effect unless otherwise specified in the SEI message semantics. For interpretation of SEI messages that apply to view components with a particular value of VOIdx less than VOIdxMax, the values of the parameters of the active view picture parameter set and the active view MVCD sequence parameter set for the particular value of VOIdx for the same access unit shall be considered in effect unless otherwise specified in the SEI message semantics.

For any active MVCD sequence parameter set or active view MVCD sequence parameter set, part of the syntax elements in the MVC sequence parameter set extension applies only to the depth views referring to this sequence parameter set, while the some other parts of the syntax elements in the MVCD sequence parameter set extension collectively apply to both the depth views referring to this sequence parameter set and the corresponding texture views. More specifically, the view dependency information of the MVCD sequence parameter set extension applies only to the depth views, and the level definitions collectively apply to operation points, each of which contains both depth views and their corresponding texture views. Moreover, the mvcd\_vui\_parameters\_extension( ) applies collectively to both the depth views referring to this MVCD sequence parameter set and the corresponding texture views. The vui\_parameters( ) included in the sequence parameter set data syntax structure, if present, apply collectively to both the depth views referring to this sequence parameter set and the corresponding texture views, except for the aspect ratio information and the bitstream restriction information, if present, which apply only to the depth views referring to this MVCD sequence parameter set. The aspect ratio information and the bitstream restriction information for the texture views may be present in the vui\_parameters( ) syntax structure included in an MVC sequence parameter set.

Order of access units and association to coded video sequences

The specification of clause H.7.4.1.2.2 apply.

Order of NAL units and coded pictures and association to access units

The specification of clause H.7.4.1.2.3 applies with the following modifications.

NOTE – Some bitstreams that conform to one or more profiles specified in this annex do not conform to any profile specified in Annex A (prior to operation of the base view extraction process specified in clause I.8.5.4). As specified in clauses 7.4.1 and 7.4.1.2.3 for the profiles specified in Annex A, NAL units with nal\_unit\_type equal to 21 are classified as non-VCL NAL units that must be preceded within each access unit by at least one NAL unit with nal\_unit\_type in the range of 1 to 5, inclusive. For this reason, any bitstream that conforms to one or more profiles specified in this annex does not conform to any profile specified in Annex A when it contains any of the following:

– any access unit that does not contain any NAL units with nal\_unit\_type equal to 1 or 5, but contains one or more NAL units with nal\_unit\_type equal to 6, 7, 8, 9, or 15

– any access unit in which one or more NAL units with nal\_unit\_type equal to 7, 8, or 15 is present after the last NAL unit in the access unit with nal\_unit\_type equal to 1 or 5.

The association of VCL NAL units to primary or redundant coded pictures is specified in clause I.7.4.1.2.5.

The constraints for the detection of the first VCL NAL unit of a primary coded picture are specified in clause I.7.4.1.2.4.

The constraint expressed in clause H.7.4.1.2.3 on the order of a buffering period SEI message is replaced by the following constraints.

– When an SEI NAL unit containing a buffering period SEI message is present, the following applies:

– If the buffering period SEI message is the only buffering period SEI message in the access unit and it is not included in an MVC scalable nesting SEI message or a MVCD scalable nesting SEI message, the buffering period SEI message shall be the first SEI message payload of the first SEI NAL unit in the access unit.

– Otherwise (the buffering period SEI message is not the only buffering period SEI message in the access unit or it is included in an MVC scalable nesting SEI message or it is included in a MVCD scalable nesting SEI message), the following constraints are specified:

– When a buffering period SEI message that is not included in either an MVC scalable nesting SEI message or a MVCD scalable nesting SEI message is present, this buffering period SEI message shall be the only SEI message payload of the first SEI NAL unit in the access unit.

– An MVC scalable nesting SEI message that includes a buffering period SEI message shall not include any other SEI messages and shall be the only SEI message inside the SEI NAL unit.

– A MVCD scalable nesting SEI message that includes a buffering period SEI message shall not include any other SEI messages and shall be the only SEI message inside the SEI NAL unit.

– All SEI NAL units that precede an SEI NAL unit that contains an MVC scalable nesting SEI message with a buffering period SEI message as payload, or a MVCD scalable nesting SEI message with a buffering period SEI message as payload in an access unit shall only contain buffering period SEI messages or MVC scalable nesting SEI messages with a buffering period SEI message as payload, or MVCD scalable nesting SEI messages with a buffering period SEI message.

Detection of the first VCL NAL unit of a primary coded picture

The specification of clause H.7.4.1.2.4 applies.

Order of VCL NAL units and association to coded pictures

The specification of clause H.7.4.1.2.5 applies with following modifications.

Each VCL NAL unit is part of a coded picture.

Let voIdx be the value of VOIdx of any particular VCL NAL unit. The order of the VCL NAL units within a coded picture is constrained as follows:

* For all VCL NAL units following this particular VCL NAL unit, the value of VOIdx shall be greater than or equal to voIdx.
* All VCL NAL units for a depth view component, if present, shall follow any VCL NAL unit of a texture view component with a same value of VOIdx.

For each set of VCL NAL units within a texture or depth view component, the following applies:

– If arbitrary slice order, as specified in Annex A, clause H.10 or clause I.10, is allowed, coded slice NAL units of a view component may have any order relative to each other.

* Otherwise (arbitrary slice order is not allowed), coded slice NAL units of a slice group shall not be interleaved with coded slice NAL units of another slice group and the order of coded slice NAL units within a slice group shall be in the order of increasing macroblock address for the first macroblock of each coded slice NAL unit of the same slice group.

The following applies:

– If a coded texture view component with a particular view\_id is the first field view component of a complementary field pair, the depth view component with the same view\_id value, if present in the access unit, shall be a coded frame view component or the first field view component of a complementary field pair.

– Otherwise, if a coded texture view component with a particular view\_id is the second field view component of a complementary field pair, the depth view component with the same view\_id value, if present in the access unit, shall be the second field view component of a complementary field pair.

– Otherwise, if a coded texture view component with a particular view\_id is a non-paired field, the depth view component with the same view\_id value, if present in the access unit, shall be a coded frame view component or a non-paired field.

– Otherwise (a coded texture view component with a particular view\_id is a coded frame), the depth view component with the same view\_id value, if present in the access unit, shall be a coded frame view component.

NAL units having nal\_unit\_type equal to 12 may be present in the access unit but shall not precede the first VCL NAL unit of the primary coded picture within the access unit.

NAL units having nal\_unit\_type equal to 0 or in the range of 24 to 31, inclusive, which are unspecified, may be present in the access unit but shall not precede the first VCL NAL unit of the primary coded picture within the access unit.

NAL units having nal\_unit\_type in the range of 22 to 23, inclusive, which are reserved, shall not precede the first VCL NAL unit of the primary coded picture within the access unit (when specified in the future by ITU-T | ISO/IEC).

* + - 1. Raw byte sequence payloads and RBSP trailing bits semantics
         1. Sequence parameter set RBSP semantics

The semantics specified in clause 7.4.2.1 apply.

Sequence parameter set data semantics

The semantics specified in clause H.7.4.2.1.1 apply with the substitution of MVCD sequence parameter set for MVC sequence parameter set. All constraints specified in clause H.7.4.2.1.1 apply only to the texture view components for which the MVCD sequence parameter set is the active texture MVC sequence parameter set or to the depth view components for which the MVCD sequence parameter set is the active view MVC sequence parameter set as specified in clause I.7.4.1.2.1.

Scaling list semantics

The semantics specified in clause H.7.4.2.1.1.1 apply.

Sequence parameter set extension RBSP semantics

The semantics specified in clause 7.4.2.1.2 apply. Additionally, the following applies.

Sequence parameter set extension RBSPs can only follow sequence parameter set RBSPs in decoding order. Subset sequence parameter set RBSPs shall not be followed by a sequence parameter set extension RBSP. The contents of sequence parameter set extension RBSPs only apply when the base view, which conforms to one or more of the profiles specified in Annex A, of a coded video sequence conforming to one or more profiles specified in Annex I is decoded.

Subset sequence parameter set RBSP semantics

The semantics specified in clause 7.4.2.1.3 apply with the following additions.

**mvcd\_vui\_parameters\_present\_flag** equal to 0 specifies that the syntax structure mvc\_vui\_parameters\_extension( ) corresponding to MVCD VUI parameters extension is not present. mvcd\_vui\_parameters\_present\_flag equal to 1 specifies that the syntax structure mvc\_vui\_parameters\_extension( ) is present and referred to as MVCD VUI parameters extension.

**texture\_vui\_parameters\_present\_flag** equal to 0 specifies that the syntax structure mvc\_vui\_parameters\_extension( ) corresponding to MVCD texture sub-bitstream VUI parameters extension is not present. texture\_vui\_parameters\_present\_flag equal to 1 specifies that the syntax structure mvc\_vui\_parameters\_extension( ) is present and referred to as MVCD texture sub-bitstream VUI parameters extension.

Sequence parameter set MVCD extension semantics

The semantics specified in clause H.7.4.2.1.4 apply with the substitution of texture view component or depth view component for view component and with the following additions:

**depth\_view\_present\_flag[**i**]** equal to 0 specifies that there is no depth view having a view\_id equal to view\_id[ i ] and VOIdx equal to i. depth\_view\_present\_flag[ i ] equal to 1 specifies that there is a depth view having a view\_id equal to view\_id[ i ].

**texture\_view\_present\_flag[**i**]** equal to 0 specifies that there is no texture view having a view\_id equal to view\_id[ i ] and VOIdx equal to i. texture\_view\_present\_flag[ i ] equal to 1 specifies that there is a texture view having a view\_id equal to view\_id[ i ] and VOIdx equal to i. When depth\_view\_present\_flag[ i ] is equal to 0, texture\_view\_present\_flag[ i ] shall be equal to 1.

num\_anchor\_refs\_l0[ i ], anchor\_ref\_l0[ i ][ j ], num\_anchor\_refs\_l1[ i ], anchor\_ref\_l1[ i ][ j ], num\_non\_anchor\_refs\_l0[ i ], non\_anchor\_ref\_l0[ i ][ j ], num\_non\_anchor\_refs\_l1[ i ], and non\_anchor\_ref\_l1[ i ][ j ] apply to depth view components.

**applicable\_op\_depth\_flag[**i**][**j**][**k**]** equal to 0 indicates that the depth view with view\_id equal to applicable\_op\_target\_view\_id[ i ][ j ][ k ] is not included in the j-th operation point. applicable\_op\_depth\_flag[ i ][ j ][ k ] equal to 1 indicates that the depth view with view\_id equal to applicable\_op\_target\_view\_id[ i ][ j ][ k ] is included in the j-th operation point.

**applicable\_op\_texture\_flag[**i**][**j**][**k**]** equal to 0 indicates that the texture view with view\_id equal to applicable\_op\_target\_view\_id[ i ][ j ][ k ] is not included in the j-th operation point. applicable\_op\_texture\_flag[ i ][ j ][ k ] equal to 1 indicates that the texture view with view\_id equal to applicable\_op\_target\_view\_id[ i ][ j ][ k ] is included in the j-th operation point. When applicable\_op\_depth\_flag[ i ][ j ][ k ] is equal to 0, applicable\_op\_texture\_flag[ i ][ j ][ k ] shall be equal to 1.

**applicable\_op\_num\_texture\_views\_minus1[**i**][**j**]** plus 1 specifies the number of texture views required for decoding the target output views corresponding to the j-th operation point to which the level indicated by level\_idc[ i ] applies. The number of texture views specified by applicable\_op\_num\_views\_minus1 includes the texture views of the target output views and the texture views that the target output views depend on. The value of applicable\_op\_num\_texture\_views\_minus1[ i ][ j ] shall be in the range of 0 to 1023, inclusive.

**applicable\_op\_num\_depth\_views[** i **][** j **]** specifies the number of depth views required for decoding the target output views corresponding to the j-th operation point to which the level indicated by level\_idc[ i ] applies. The number of depth views specified by applicable\_op\_num\_depth\_views\_minus1 includes the depth views of the target output views and the depth views that the depth views of the target output views depend on. The value ofapplicable\_op\_num\_depth\_views\_minus1[ i ][ j ] shall be in the range of 0 to 1023, inclusive.

All sequence parameter set MVCD extensions that are included in the active view MVCD sequence parameter set RBSPs of one coded video sequence shall be identical.

* + - * 1. Picture parameter set RBSP semantics

The semantics specified in clause H.7.4.2.2 apply with substituting MVCD sequence parameter set for MVC sequence parameter set. All constraints specified in clause H.7.4.2.2 apply only to the texture or depth view components for which the picture parameter set is the active picture parameter set or the active view picture parameter set or the active texture picture parameter set as specified in clause I.7.4.1.2.1.

* + - * 1. Supplemental enhancement information RBSP semantics

The semantics specified in clause H.7.4.2.3 apply.

Supplemental enhancement information message semantics

The semantics specified in clause H.7.4.2.3.1 apply.

* + - * 1. Access unit delimiter RBSP semantics

The semantics specified in clause H.7.4.2.4 apply.

NOTE – The value of primary\_pic\_type applies to the slice\_type values in all slice headers of the primary coded picture, including the slice\_type syntax elements in all NAL units with nal\_unit\_type equal to 1, 5, 20 or 21. NAL units with nal\_unit\_type equal to 2 are not present in bitstreams conforming to any of the profiles specified in this annex.

* + - * 1. End of sequence RBSP semantics

The semantics specified in clause H.7.4.2.5 apply.

* + - * 1. End of stream RBSP semantics

The semantics specified in clause H.7.4.2.6 apply.

* + - * 1. Filler data RBSP semantics

The semantics specified in clause H.7.4.2.7 apply with the following modifications.

Filler data NAL units shall be considered to contain the syntax elements priority\_id, view\_id, and temporal\_id with values that are inferred as follows:

1. Let prevMvcNalUnit be the most recent NAL unit in decoding order that has nal\_unit\_type equal to 14, 20 or 21.

NOTE – The most recent NAL unit in decoding order with nal\_unit\_type equal to 14, 20 or 21 always belongs to the same access unit as the filler data NAL unit.

1. The values of priority\_id, view\_id, and temporal\_id for the filler data NAL unit are inferred to be equal to the values of priority\_id, view\_id, and temporal\_id, respectively, of the NAL unit prevMvcNalUnit.
   * + - 1. Slice layer without partitioning RBSP semantics

The semantics specified in clause H.7.4.2.8 apply.

* + - * 1. Slice data partition RBSP semantics

Slice data partition syntax is not present in bitstreams conforming to one or more of the profiles specified in Annex I.

* + - * 1. RBSP slice trailing bits semantics

The semantics specified in H.7.4.2.10 apply.

* + - * 1. RBSP trailing bits semantics

The semantics specified in clause H.7.4.2.11 apply.

* + - * 1. Prefix NAL unit RBSP semantics

The semantics specified in clause H.7.4.2.12 apply.

* + - * 1. Slice layer extension RBSP semantics

The semantics specified in clause H.7.4.2.13 apply.

* + - 1. Slice header semantics

The semantics specified in clause H.7.4.3 apply with the substitution of texture view component (for nal\_unit\_type equal to 1, 5, and 20) or depth view component (for nal\_unit\_type equal to 21) for view component and with the following modifications.

When nal\_unit\_type is equal to 1, 5, or 20, all constraints specified in clause H.7.4.3 apply only to the texture view components with the same value of VOIdx. When nal\_unit\_type is equal to 21, all constraints specified in clause H.7.4.3 apply only to the depth view components with the same value of VOIdx.

The value of the following MVCD sequence parameter set syntax elements shall be the same across all coded slice NAL units of nal\_unit\_type equal to 1, 5, and 20 of an access unit: chroma\_format\_idc.

The value of the following slice header syntax elements shall be the same across all coded slice NAL units of nal\_unit\_type equal to 1, 5, and 20 of an access unit: field\_pic\_flag and bottom\_field\_flag.

The value of the following slice header syntax elements shall be the same across all coded slice NAL units of nal\_unit\_type equal to 21 of an access unit: field\_pic\_flag and bottom\_field\_flag.

* + - * 1. Reference picture list modification semantics

The semantics specified in clause H.7.4.3.1 apply.

Reference picture list MVC modification semantics

The semantics specified in clause H.7.4.3.1 apply**.**

* + - * 1. Prediction weight table semantics

The semantics specified in clause H.7.4.3.2 apply.

* + - * 1. Decoded reference picture marking semantics

The semantics specified in clause 7.4.3.3 apply to each view independently, with "sequence parameter set" being replaced by "MVCD sequence parameter set", and "primary coded picture" being replaced by "texture view component" for nal\_unit\_type equal to 1, 5, and 20, and by "depth view component" for nal\_unit\_type equal to 21.

* + - 1. Slice data semantics

The semantics specified in clause H.7.4.4 apply.

* + - 1. Macroblock layer semantics

The semantics specified in clause H.7.4.5 apply.

* + - * 1. Macroblock prediction semantics

The semantics specified in clause H.7.4.5.1 apply.

* + - * 1. Sub-macroblock prediction semantics

The semantics specified in clause H.7.4.5.2 apply.

* + - * 1. Residual data semantics

The semantics specified in clause H.7.4.5.3 apply.

Residual luma semantics

The semantics specified in clause H.7.4.5.3.1 apply.

Residual block CAVLC semantics

The semantics specified in clause H.7.4.5.3.2 apply.

Residual block CABAC semantics

The semantics specified in clause H.7.4.5.3.3 apply.

* 1. MVCD decoding process

This clause specifies the decoding process for an access unit of a coded video sequence conforming to one or more of the profiles specified in Annex I. Specifically, this clause specifies how the decoded picture with multiple texture view components and multiple depth view components is derived from syntax elements and global variables that are derived from NAL units in an access unit when the decoder is decoding the operation point identified by the target temporal level and the target output texture and depth views.

The decoding process is specified such that all decoders shall produce numerically identical results for the target output texture and depth views. Any decoding process that produces identical results for the target output texture and depth views to the process described here conforms to the decoding process requirements of this Recommendation | International Standard.

Unless stated otherwise, the syntax elements and derived upper-case variables that are referred to by the decoding process specified in this clause and all child processes invoked from the process specified in this clause are the syntax elements and derived upper-case variables for the current access unit.

The target output texture and depth views are either specified by external means not specified in this Specification, or, when not specified by external means, there shall be one target output texture view which is the base texture view.

NOTE – The association of VOIdx values to view\_id values according to the decoding process of clause I.8 may differ from that of the decoding process of clause H.8.

A target output view may include only a texture view, only a depth view, or both the texture view and the depth view, which have the same view\_id value.

All sub-bitstreams that can be derived using the sub-bitstream extraction process with depthPresentFlagTarget equal to 0 or 1, pIdTarget equal to any value in the range of 0 to 63, inclusive, tIdTarget equal to any value in the range of 0 to 7, inclusive, viewIdTargetList consisting of any one or more viewIdTarget's identifying the views in the bitstream as inputs as specified in clause I.8.5 shall result in a set of coded video sequences, with each coded video sequence conforming to one or more of the profiles specified in Annex A, Annex H and Annex I.

Let vOIdxList be a list of integer values specifying the VOIdx values of the view components of the access unit. The variable VOIdxMax is set equal to the maximum value of the entries in the list vOIdxList, and the variable vOIdxMin is set to the minimum value of the entries in the list vOIdxList. When the current access unit is an anchor access unit, the variable VOIdxMin is set to vOIdxMin.

The MVCD video decoding process specified in this clause is repeatedly invoked for each texture and depth view component with VOIdx from vOIdxMin to VOIdxMax, inclusive, which is present in the list vOIdxList, in increasing order of VOIdx and in decoding order of texture or depth view components as specified in clause I.7.4.1.2.5.

Outputs of the MVCD video decoding process are decoded samples of the current primary coded picture including all decoded texture and depth view components of the target output texture and depth views.

For each texture view component and each depth view component, the specifications in clause H.8 apply, with the decoding processes for picture order count, reference picture lists construction and decoded reference picture marking being modified in clauses I.8.1, I.8.2, I.8.3, and I.8.4, respectively. The MVCD inter prediction and inter-view prediction process is specified in clause I.8.4.

* + 1. MVCD decoding process for picture order count

The specifications in clause 8.2.18.2.1 apply independently for each texture view or depth view.

* + 1. MVC decoding process for reference picture lists construction

The specification of clause H.8.2 apply with substituting "view component" as either "texture view component" or "depth view component", and "frame view component"" as either "depth frame view component" or "texture frame view component ", and "field view component" as "texture field view component" or "depth field view component".

Additionally, an inter-view reference component or the inter-view only reference component is identified by the view\_id and a depth view component when the current slice is a part of a coded depth view component or a texture view component if the current slice is a part of a coded texture view component.

* + - 1. Initialisation process for reference picture list for inter-view prediction references

The specifications of clause H.8.2.1 apply.

* + - 1. Modification process for reference picture lists

The specifications of clause H.8.2.2 apply.

* + - * 1. Modification process of reference picture lists for short-term reference pictures for inter prediction

The specifications of clause H.8.2.2.1 apply.

* + - * 1. Modification process of reference picture lists for long-term reference pictures for inter prediction

The specifications of clause H.8.2.2.2 apply.

* + - * 1. Modification process for reference picture lists for inter-view prediction references

The specifications of clause H.8.2.2.3 apply.

* + 1. MVCD decoded reference picture marking process

The specifications of clause H.8.3 apply. Additionally, the following applies.

The process specified in this clause is invoked for a particular texture view or depth view with view order index VOIdx. The specifications in clause H.8.3 apply with "view component" being replaced by either "texture view component" or "depth view component", "frame view component" being replaced by either "texture frame view component" or "depth frame view component", and "field view component" being replaced by either "texture field view component" or "depth field view component". During the invocation of the process for a particular texture view, only texture view components of the particular view are considered. During the invocation of the process for a particular depth view, only depth view components of the particular view are considered. The marking of view components of other views is not changed.

NOTE – A texture view component of a picture may have a different marking status than other texture view components of the same picture. A depth view component of a picture may have a different marking status than other depth view components of the same picture. A texture view component of a picture may have a different marking status than a depth view component.

* + 1. MVCD inter prediction and inter-view prediction process

The specifications of clause H.8.4 apply.

* + 1. Specification of bitstream subsets

The specifications of clause H.8.5 apply.

* + - 1. Derivation process for required anchor view components

The specification of clause H.8.5.1 apply with substituting "view component" with "depth view component" and "view" with "depth view" or and substituting "required for anchor" with "required for anchor depth".

The specification of clause H.8.5.1 apply with substituting "view component" with "texture view component" and "view" with "texture view" and substituting "required for anchor" with "required for anchor texture".

* + - 1. Derivation process for required non-anchor view components

The specification of clause H.8.5.2 apply with substituting "view component" with "depth view component" and "view" with "depth view" and substituting "required for anchor" with "required for anchor depth".

The specification of clause H.8.5.2 apply with substituting "view component" with "depth view component" with either "depth view", and substituting "required for anchor" with "required for anchor texture".

* + - 1. Sub-bitstream extraction process

It is requirement of bitstream conformance that any sub-bitstream that is the output of the process specified in this clause with depthPresentFlagTarget equal to 0 or 1, pIdTarget equal to any value in the range of 0 to 63, inclusive, tIdTarget equal to any value in the range of 0 to 7, inclusive, viewIdTargetList consisting of any one or more values of viewIdTarget identifying the views in the bitstream, shall be conforming to this Recommendation | International Standard.

NOTE 1 – A conforming bitstream contains one or more coded slice NAL units with priority\_id equal to 0 and temporal\_id equal to 0.

NOTE 2 – It is possible that not all operation points of sub-bitstreams resulting from the sub-bitstream extraction process have an applicable level\_idc or level\_idc[ i ]. In this case, each coded video sequence in a sub-bitstream must still conform to one or more of the profiles specified in Annex A, Annex H and Annex I, but may not satisfy the level constraints specified in clauses A.3, H.10.2 and I.10.2, respectively.

Inputs to this process are:

– a variable depthPresentFlagTarget (when present),

– a variable pIdTarget (when present),

– a variable tIdTarget (when present),

– a list viewIdTargetList consisting of one or more values of viewIdTarget (when present).

– a list viewIdDepthTargetList consisting of one or more value of viewIdDepthTarget (when present).

Outputs of this process are a sub-bitstream and a list of VOIdx values VOIdxList.

When depthPresentFlagTarget is not present as input to this clause, depthPresentFlagTarget is inferred to be equal to 0.

When pIdTarget is not present as input to this clause, pIdTarget is inferred to be equal to 63.

When tIdTarget is not present as input to this clause, tIdTarget is inferred to be equal to 7.

When viewIdTargetList is not present as input to this clause, there shall be one value of viewIdTarget inferred in viewIdTargetList and the value of viewIdTarget is inferred to be equal to view\_id of the base view.

When viewIdDepthTargetList is not present as input to this clause, the viewIdDepthTargetList is inferred to be identical to viewIdTargetList. viewIdDepthTargetList shall not be present as input if depthPresentFlagTarget is equal to 0.

The sub-bitstream is derived by applying the following operations in sequential order:

1. Let VOIdxList be empty and minVOIdx be the VOIdx value of the base view.
2. For each value of viewIdTarget included in viewIdTargetList, invoke the process specified in clause I.8.5.1 for texture views with the viewIdTarget as input.
3. If depthPresentFlagTarget is equal to 1, for each value of viewIdTarget included in viewIdDepthTargetList, invoke the process specified in clause I.8.5.1 for depth views with the viewIdTarget as input.
4. For each value of viewIdTarget included in viewIdTargetList, invoke the process specified in clause I.8.5.2 for texture views with the value of viewIdTarget as input.
5. If depthPresentFlagTarget is equal to 1, for each value of viewIdTarget included in viewIdDepthTargetList, invoke the process specified in clause I.8.5.2 for depth views with the viewIdTarget as input.
6. Mark all VCL NAL units of texture view components and filler data NAL units for which any of the following conditions are true as "to be removed from the bitstream":

– priority\_id is greater than pIdTarget,

– temporal\_id is greater than tIdTarget,

– anchor\_pic\_flag is equal to 1, nal\_unit\_type is not equal to 21 and view\_id is not marked as "required for anchor texture",

– anchor\_pic\_flag is equal to 0, nal\_unit\_type is not equal to 21 and view\_id is not marked as "required for non-anchor texture",

– anchor\_pic\_flag is equal to 1, nal\_unit\_type is equal to 21 and view\_id is not marked as "required for anchor depth",

– anchor\_pic\_flag is equal to 0, nal\_unit\_type is equal to 21 and view\_id is not marked as "required for non-anchor depth",

– nal\_unit\_type is not equal to 21, nal\_ref\_idc is equal to 0 and inter\_view\_flag is equal to 0 and view\_id is not equal to any value in the list viewIdTargetList.

– nal\_unit\_type is equal to 21, nal\_ref\_idc is equal to 0 and inter\_view\_flag is equal to 0 and view\_id is not equal to any value in the list viewIdDepthTargetList.

– nal\_unit\_type is equal to 21 and depthPresentFlagTarget is equal to 0.

1. Remove all access units for which all VCL NAL units are marked as "to be removed from the bitstream".
2. Remove all VCL NAL units and filler data NAL units that are marked as "to be removed from the bitstream".
3. When VOIdxList contains only one value of VOIdx that is equal to minVOIdx, remove the following NAL units:

– all NAL units with nal\_unit\_type equal to 14 or 15,

– all NAL units with nal\_unit\_type equal to 6 in which the first SEI message has payloadType in the range of 36 to 44, inclusive.

NOTE 3 – When VOIdxList contains only one value of VOIdx equal to minVOIdx, the sub-bitstream contains only the base view or only a temporal subset of the base view.

1. Remove all NAL units with nal\_unit\_type equal to 6 in which the first SEI message has payloadType equal to 0 or 1, or the first SEI message has payloadType equal to equal to 37 (MVC scalable nesting SEI message) and operation\_point\_flag in the first SEI message is equal to 1.

NOTE 4 – The buffering period SEI and picture timing SEI messages, when not nested or nested in the MVC scalable nesting SEI message, apply for a sub-bitstream obtained with the sub-bitstream extraction process of clause H.8.5.3, which does not process NAL units of nal\_unit\_type equal to 21.

1. When depthPresentFlagTarget is equal to 0, the following applies in sequential order.

– Replace all NAL units with nal\_unit\_type equal to 6 in which payloadType indicates a MVCD texture scalable nesting SEI message with sei\_op\_texture\_only\_flag equal to 0 with a MVC scalable nesting SEI message with the same values of num\_view\_components\_op\_minus1, sei\_op\_view\_id[ i ] and sei\_op\_temporal\_id and the same nested SEI messages.

– Remove all NAL units with nal\_unit\_type equal to 6 in which payloadType indicates a MVCD texture scalable nesting SEI message.

– The following applies for each active texture MVCD sequence parameter set RBSP.

– Replace mvc\_vui\_parameters\_extension( ) syntax structure in an active texture MVCD sequence parameter set RBSPs with the mvc\_vui\_parameters\_extension( ) syntax structure of the MVCD texture sub-bitstream VUI parameters extension, if both mvc\_vui\_parameters\_extension( ) syntax structures apply to the same views.

– Otherwise, remove mvc\_vui\_parameters\_extension( ) syntax structure in an active texture MVCD sequence parameter set RBSP.

– When depthPresentFlagTarget is equal to 0, remove all NAL units with nal\_unit\_type equal to 6 in which the first SEI message has payloadType in the range of 48 to 52, inclusive.

1. Let maxTId be the maximum temporal\_id of all the remaining VCL NAL units. Remove all NAL units with nal\_unit\_type equal to 6 that only contain SEI messages that are part of an MVC scalable nesting SEI message or MVCD scalable nesting SEI message with any of the following properties:

– operation\_point\_flag is equal to 0 and all\_view\_components\_in\_au\_flag is equal to 0 and none of sei\_view\_id[ i ] for all i in the range of 0 to num\_view\_components\_minus1, inclusive, corresponds to a VOIdx value included in VOIdxList,

– operation\_point\_flag is equal to 1 and either sei\_op\_temporal\_id is greater than maxTId or the list of sei\_op\_view\_id[ i ] for all i in the range of 0 to num\_view\_components\_op\_minus1, inclusive, is not a subset of viewIdTargetList (i.e., it is not true that sei\_op\_view\_id[ i ] for any i in the range of 0 to num\_view\_components\_op\_minus1, inclusive, is equal to a value in viewIdTargetList).

1. Remove each view scalability information SEI message and each operation point not present SEI message, when present.
2. When VOIdxList does not contain a value of VOIdx equal to minVOIdx, the view with VOIdx equal to the minimum VOIdx value included in VOIdxList is converted to the base view of the extracted sub-bitstream.

NOTE 5 – When VOIdxList does not contain a value of VOIdx equal to minVOIdx, the resulting sub-bitstream according to the operation steps 1-9 above does not contain a base view that conforms to one or more profiles specified in Annex A. In this case, by this operation step, the remaining view with the new minimum VOIdx value is converted to be the new base view that conforms to one or more profiles specified in Annex A and Annex H.

* + - 1. Specification of the base view bitstream

A bitstream that conforms to one or more profiles as specified in Annex I shall contain a base view bitstream that conforms to one or more of the profiles specified in Annex A. This base view bitstream is derived by invoking the sub‑bitstream extraction process as specified in clause I.8.5.3 with no input and the base view bitstream being the output.

NOTE – Although all multiview bitstreams that conform to one or more of the profiles specified in this annex contain a base view bitstream that conforms to one or more of the profiles specified in Annex A, the complete multiview bitstream (prior to operation of the base view extraction process specified in this clause) may not conform to any profile specified in Annex A.

* + - 1. Specification of the stereoscopic texture bitstream

A bitstream that conforms to a profile as specified in Annex I shall contain at least one sub-bitstream that conforms to one or more of the profiles specified in Annex H with number of views equal to 2. This stereoscopic texture bitstream is derived by invoking the sub‑bitstream extraction process as specified in clause I.8.5.3 with depthPresentFlagTarget equal to 0 and viewIdTargetList containing the view\_id values of the base view and a non-base view, the texture of which does not depend on any other non-base view for decoding.

* 1. Parsing process

The specifications in clause 9 apply.

* 1. Profiles and levels

The specifications in Annex H apply. Additional profiles and specific values of profile\_idc are specified in the following.

The profiles that are specified in clause I.10.1 are also referred to as the profiles specified in Annex I.

* + 1. Profiles

All constraints for picture parameter sets that are specified in the following are constraints for picture parameter sets that become the active picture parameter set or an active view picture parameter set inside the bitstream. All constraints for MVCD sequence parameter sets that are specified in the following are constraints for MVCD sequence parameter sets that become the active MVCD sequence parameter set or an active view MVCD sequence parameter set inside the bitstream.

* + - 1. Multiview Depth High Profile

Bitstreams conforming to the Multiview Depth High profile shall obey the following constraints:

– The base view bitstream as specified in clause I.8.5.4 shall obey all constraints of the High profile specified in clause A.2.4 and all active sequence parameter sets shall fulfill one of the following conditions:

– profile\_idc is equal to 77 or constraint\_set1\_flag is equal to 1,

– profile\_idc is equal to 100.

– The sub-bitstream of stereoscopic texture bitstream as specified in clause I.8.5.5 shall obey all constraints of the Stereo High profile specified in clause H.10.2 and all active MVC sequence parameter sets shall fulfill one of the following conditions:

– profile\_idc is equal to 128,

– profile\_idc is equal to 118 and constraint\_set5\_flag is equal to 1,

– profile\_idc is equal to 100,

– profile\_idc is equal to 77 or constraint\_set1\_flag is equal to 1.

– Only I, P, and B slice types may be present.

– NAL unit streams shall not contain nal\_unit\_type values in the range of 2 to 4, inclusive.

– Arbitrary slice order is not allowed.

– Picture parameter sets shall have num\_slice\_groups\_minus1 equal to 0 only.

– Picture parameter sets shall have redundant\_pic\_cnt\_present\_flag equal to 0 only.

– When frame\_mbs\_only\_flag is equal to 1 in an active sequence parameter set for a texture view, frame\_mbs\_only\_flag shall be equal to 1 in the active sequence parameter set for the depth view having the same view\_id.

– When frame\_mbs\_only\_flag is equal to 0 in an active sequence parameter set for a depth view, mb\_adaptive\_frame\_field\_flag shall be equal to 0.

– MVCD sequence parameter sets for the depth views shall have chroma\_format\_idc equal to 0 only.

– MVCD sequence parameter sets shall have bit\_depth\_luma\_minus8 equal to 0 only.

– MVCD sequence parameter sets shall have bit\_depth\_chroma\_minus8 equal to 0 only.

– MVCD sequence parameter sets shall have qpprime\_y\_zero\_transform\_bypass\_flag equal to 0 only.

– For each access unit, the value of level\_idc for all active view MVCD sequence parameter set RBSPs shall be the same as the value of level\_idc for the active MVCD sequence parameter set RBSP.

– The level constraints specified for the Multiview Depth High profile in clause I.10.2 shall be fulfilled.

Conformance of a bitstream to the Multiview Depth High profile is indicated by profile\_idc being equal to 138.

Decoders conforming to the Multiview Depth High profile at a specific level shall be capable of decoding all bitstreams in which both of the following conditions are true:

1. All active MVCD sequence parameter sets have one or more of the following conditions fulfilled:

– profile\_idc is equal to 138,

– profile\_idc is equal to 128,

– profile\_idc is equal to 118 and constraint\_set5\_flag is equal to 1,

– profile\_idc is equal to 100,

– profile\_idc is equal to 77 or constraint\_set1\_flag is equal to 1.

1. All active MVCD sequence parameter sets have one or more of the following conditions fulfilled:

– level\_idc or (level\_idc and constraint\_set3\_flag) represent a level less than or equal to the specific level,

– level\_idc[ i ] or (level\_idc[ i ] and constraint\_set3\_flag) represent a level less than or equal to the specific level.

* + - 1. MFC Depth High Profile

Bitstreams conforming to the Multiview Depth High profile shall obey the following constraints:

– The base view bitstream as specified in clause I.8.5.4 shall obey all constraints of the High profile specified in clause A.2.4 and all active sequence parameter sets shall fulfill one of the following conditions:

– profile\_idc is equal to 77 or constraint\_set1\_flag is equal to 1,

– profile\_idc is equal to 100.

– The sub-bitstream of stereoscopic texture bitstream as specified in clause I.8.5.5 shall obey all constraints of the MFC High profile specified in clause H.10.1.3 and all active MVC sequence parameter sets shall fulfill one of the following conditions:

– profile\_idc is equal to 134,

– profile\_idc is equal to 100,

– profile\_idc is equal to 77 or constraint\_set1\_flag is equal to 1.

– Only I, P, and B slice types may be present.

– NAL unit streams shall not contain nal\_unit\_type values in the range of 2 to 4, inclusive.

– Arbitrary slice order is not allowed.

– Picture parameter sets shall have num\_slice\_groups\_minus1 equal to 0 only.

– Picture parameter sets shall have redundant\_pic\_cnt\_present\_flag equal to 0 only.

– When frame\_mbs\_only\_flag is equal to 1 in an active sequence parameter set for a texture view, frame\_mbs\_only\_flag shall be equal to 1 in the active sequence parameter set for the depth view having the same view\_id.

– When frame\_mbs\_only\_flag is equal to 0 in an active sequence parameter set for a depth view, mb\_adaptive\_frame\_field\_flag shall be equal to 0.

– MVCD sequence parameter sets for the depth views shall have chroma\_format\_idc equal to 0 only.

– MVCD sequence parameter sets shall have bit\_depth\_luma\_minus8 equal to 0 only.

– MVCD sequence parameter sets shall have bit\_depth\_chroma\_minus8 equal to 0 only.

– MVCD sequence parameter sets shall have qpprime\_y\_zero\_transform\_bypass\_flag equal to 0 only.

– For each access unit, the value of level\_idc for all active view MVCD sequence parameter set RBSPs shall be the same as the value of level\_idc for the active MVCD sequence parameter set RBSP.

– The level constraints specified for the MFC Depth High profile in clause I.10.2 shall be fulfilled.

Conformance of a bitstream to the MFC Depth High profile is indicated by profile\_idc being equal to 135.

Decoders conforming to the MFC Depth High profile at a specific level shall be capable of decoding all bitstreams in which both of the following conditions are true:

1. All active MVCD sequence parameter sets have one or more of the following conditions fulfilled:

– profile\_idc is equal to 135,

– profile\_idc is equal to 138,

– profile\_idc is equal to 134,

– profile\_idc is equal to 128,

– profile\_idc is equal to 118 and constraint\_set5\_flag is equal to 1,

– profile\_idc is equal to 100,

– profile\_idc is equal to 77 or constraint\_set1\_flag is equal to 1.

1. All active MVCD sequence parameter sets have one or more of the following conditions fulfilled:

– level\_idc or (level\_idc and constraint\_set3\_flag) represent a level less than or equal to the specific level,

– level\_idc[ i ] or (level\_idc[ i ] and constraint\_set3\_flag) represent a level less than or equal to the specific level.

* + 1. Levels

The following is specified for expressing the constraints in this clause:

– Let access unit n be the n-th access unit in decoding order with the first access unit being access unit 0.

– Let picture n be the primary coded picture or the corresponding decoded picture of access unit n.

Let the variable fR be derived as follows:

– If picture n is a frame, fR is set equal to 1 ÷ 172.

– Otherwise (picture n is a field), fR is set equal to 1 ÷ (172 \* 2).

The value of mvcScaleFactor is set equal to 2.

The value of mvcdScaleFactor is set equal to 2.5.

The value of NumViews indicates the number of views, including texture views and depth views, which are required for decoding the target output views corresponding to the j-th operation point for level\_idc[ i ] as signalled in the subset sequence parameter set, and is set equal to applicable\_op\_num\_depth\_views\_minus1[ i ][ j ] + applicable\_op\_num\_depth\_views\_minus1[ i ][ j ] + 2.

The value of PicWidthInMbs and FrameHeightInMbs refer to the width and height of each view component, while the value of TotalPicSizeInMbs indicates the total number of macroblocks in the texture view components and depth view components of a picture.

* + - 1. Level limits common to Multiview Depth High profiles

Bitstreams conforming to the Multiview Depth High profile at a specified level shall obey the following constraints:

1. The nominal removal time of access unit n (with n > 0) from the CPB as specified in clause C.1.2, satisfies the constraint that tr,n( n ) − tr( n − 1 ) is greater than or equal to Max( TotalPicSizeInMbs ÷ ( mvcdScaleFactor \* MaxMBPS ), fR ), where MaxMBPS is the value specified in Table A‑1 that applies to picture n − 1, and TotalPicSizeInMbs is the total number of macroblocks in the texture view components and depth view components of picture n − 1.
2. The difference between consecutive output times of pictures from the DPB as specified in clause C.2.2, satisfies the constraint that Δto,dpb( n ) >= Max( TotalPicSizeInMbs ÷ ( mvcdScaleFactor \* MaxMBPS ), fR ), where MaxMBPS is the value specified in Table A‑1 for picture n, and TotalPicSizeInMbs is the total number of macroblocks in the texture view components and depth view components of picture n, provided that picture n is a picture that is output and is not the last picture of the bitstream that is output.
3. PicWidthInMbs \* FrameHeightInMbs <= MaxFS, where MaxFS is specified in Table A‑1.
4. PicWidthInMbs <= Sqrt( MaxFS \* 8 ), where MaxFS is specified in Table A‑1.
5. FrameHeightInMbs <= Sqrt( MaxFS \* 8 ), where MaxFS is specified in Table A‑1.
6. max\_dec\_frame\_buffering <= MaxDpbFrames, where MaxDpbFrames is equal to Min( mvcdScaleFactor \* MaxDpbMbs / ( TotalPicSizeInMbs / NumViews ) ), Max( 1, Ceil( log2( NumViews ) ) ) \* 16 ) and MaxDpbMbs is specified in Table A‑1.
7. The vertical motion vector component range does not exceed MaxVmvR in units of luma frame samples, where MaxVmvR is specified in Table A‑1.
8. The horizontal motion vector range does not exceed the range of −2048 to 2047.75, inclusive, in units of luma samples.
9. Let setOf2Mb be the set of unsorted pairs of macroblocks that contains the unsorted pairs of macroblocks (mbA, mbB) of a coded video sequence for which any of the following conditions are true:

– mbA and mbB are macroblocks that belong to the same slice and are consecutive in decoding order,

– separate\_colour\_plane\_flag is equal to 0, mbA is the last macroblock (in decoding order) of a slice, and mbB is the first macroblock (in decoding order) of the next slice in decoding order,

– separate\_colour\_plane\_flag is equal to 1, mbA is the last macroblock (in decoding order) of a slice with a particular value of colour\_plane\_id, and mbB is the first macroblock (in decoding order) of the next slice with the same value of colour\_plane\_id in decoding order.

NOTE 1 – In the two above conditions, the macroblocks mbA and mbB can belong to different pictures.

For each unsorted pair of macroblocks (mbA, mbB) of the set setOf2Mb, the total number of motion vectors (given by the sum of the number of motion vectors for macroblock mbA and the number of motion vectors for macroblock mbB) does not exceed MaxMvsPer2Mb, where MaxMvsPer2Mb is specified in Table A‑1. The number of motion vectors for each macroblock is the value of the variable MvCnt after the completion of the intra or inter prediction process for the macroblock.

NOTE 2 – When separate\_colour\_plane\_flag is equal to 0, the constraint specifies that the total number of motion vectors for two consecutive macroblocks in decoding order must not exceed MaxMvsPer2Mb. When separate\_colour\_plane\_flag is equal to 1, the constraint specifies that the total number of motion vectors for two consecutive macroblocks with the same value of colour\_plane\_id in decoding order must not exceed MaxMvsPer2Mb. For macroblocks that are consecutive in decoding order but are associated with a different value of colour\_plane\_id, no constraint for the total number of motion vectors is specified.

1. The number of bits of macroblock\_layer( ) data for any macroblock is not greater than 128 + RawMbBits. Depending on entropy\_coding\_mode\_flag, the bits of macroblock\_layer( ) data are counted as follows:

– If entropy\_coding\_mode\_flag is equal to 0, the number of bits of macroblock\_layer( ) data is given by the number of bits in the macroblock\_layer( ) syntax structure for a macroblock.

– Otherwise (entropy\_coding\_mode\_flag is equal to 1), the number of bits of macroblock\_layer( ) data for a macroblock is given by the number of times read\_bits( 1 ) is called in clauses 9.3.3.2.2 and 9.3.3.2.3 when parsing the macroblock\_layer( ) associated with the macroblock.

1. The removal time of access unit 0 shall satisfy the constraint that the number of slices in picture 0 is less than or equal to mvcdScaleFactor \* ( Max( PicSizeInMbs, fR \* MaxMBPS ) + MaxMBPS \* ( tr( 0 ) − tr,n( 0 ) ) ) ÷ SliceRate, where MaxMBPS and SliceRate are the values specified in Tables A‑1 and A‑4, respectively, that apply to picture 0 and PicSizeInMbs is the number of macroblocks in a single texture view component of picture 0.
2. The removal time of access unit 0 shall satisfy the constraint that the number of slices in each view component of picture 0 is less than or equal to ( Max( PicSizeInMbs, fR \* MaxMBPS ) + MaxMBPS \* ( tr( 0 ) − tr,n( 0 ) ) ) ÷ SliceRate, where MaxMBPS and SliceRate are the values specified in Tables A‑1 and A‑4, respectively, that apply to picture 0 and PicSizeInMbs is the number of macroblocks in a single view component of picture 0.
3. The difference between consecutive removal times of access units n and n − 1 with n > 0 shall satisfy the constraint that the number of slices in picture n is less than or equal to mvcdScaleFactor \* MaxMBPS \* ( tr( n ) − tr( n − 1 ) ) ÷ SliceRate, where SliceRate is the value specified in Table A‑4 that applies to picture n.
4. The difference between consecutive removal times of access units n and n − 1 with n > 0 shall satisfy the constraint that the number of slices in each view component of picture n is less than or equal to MaxMBPS \* ( tr( n ) − tr( n − 1 ) ) ÷ SliceRate, where SliceRate is the value specified in Table A‑4 that applies to picture n.
5. MVCD sequence parameter sets shall have direct\_8x8\_inference\_flag equal to 1 for the levels specified in Table A‑4.
6. The value of sub\_mb\_type[ mbPartIdx ] with mbPartIdx = 0..3 in B macroblocks with mb\_type equal to B\_8x8 shall not be equal to B\_Bi\_8x4, B\_Bi\_4x8, or B\_Bi\_4x4 for the levels in which MinLumaBiPredSize is shown as 8x8 in Table A‑4.
7. For the VCL HRD parameters, BitRate[ SchedSelIdx ] <= cpbBrVclFactor \* MaxBR and CpbSize[ SchedSelIdx ] <= cpbBrVclFactor \*MaxCPB for at least one value of SchedSelIdx, where cpbBrVclFactor is equal to 1250. With vui\_mvc\_vcl\_hrd\_parameters\_present\_flag[ i ] being the syntax element, in the MVCD VUI parameters extension of the active MVCD sequence parameter set, that is associated with the VCL HRD parameters that are used for conformance checking (as specified in Annex C), BitRate[ SchedSelIdx ] and CpbSize[ SchedSelIdx ] are given as follows:

– If vui\_mvc\_vcl\_hrd\_parameters\_present\_flag equal to 1, BitRate[ SchedSelIdx ] and CpbSize[ SchedSelIdx ] are given by Equations E-37 and E-38, respectively, using the syntax elements of the hrd\_parameters( ) syntax structure that immediately follows vui\_mvc\_vcl\_hrd\_parameters\_present\_flag.

– Otherwise (vui\_mvc\_vcl\_hrd\_parameters\_present\_flag equal to 0), BitRate[ SchedSelIdx ] and CpbSize[ SchedSelIdx ] are inferred as specified in clause E.2.2 for VCL HRD parameters.

MaxBR and MaxCPB are specified in Table A‑1 in units of cpbBrVclFactor bits/s and cpbBrVclFactor bits, respectively. The bitstream shall satisfy these conditions for at least one value of SchedSelIdx in the range 0 to cpb\_cnt\_minus1, inclusive.

1. For the NAL HRD parameters, BitRate[ SchedSelIdx ] <= cpbBrNalFactor \* MaxBR and CpbSize[ SchedSelIdx ] <= cpbBrNalFactor \*MaxCPB for at least one value of SchedSelIdx, where cpbBrNalFactor is equal to 1500. With vui\_mvc\_nal\_hrd\_parameters\_present\_flag[ i ] being the syntax element, in the MVCD VUI parameters extension of the active MVCD sequence parameter set, that is associated with the NAL HRD parameters that are used for conformance checking (as specified in Annex C), BitRate[ SchedSelIdx ] and CpbSize[ SchedSelIdx ] are given as follows:

– If vui\_mvc\_nal\_hrd\_parameters\_present\_flag equal to 1, BitRate[ SchedSelIdx ] and CpbSize[ SchedSelIdx ] are given by Equations E-37 and E-38, respectively, using the syntax elements of the hrd\_parameters( ) syntax structure that immediately follows vui\_mvc\_nal\_hrd\_parameters\_present\_flag.

– Otherwise (vui\_mvc\_nal\_hrd\_parameters\_present\_flag equal to 0), BitRate[ SchedSelIdx ] and CpbSize[ SchedSelIdx ] are inferred as specified in clause E.2.2 for NAL HRD parameters.

MaxBR and MaxCPB are specified in Table A‑1 in units of cpbBrNalFactor bits/s and cpbBrNalFactor bits, respectively. The bitstream shall satisfy these conditions for at least one value of SchedSelIdx in the range 0 to cpb\_cnt\_minus1, inclusive.

1. The sum of the NumBytesInNALunit variables for access unit 0 is less than or equal to 384 \* mvcdScaleFactor \* ( Max( PicSizeInMbs, fR \* MaxMBPS ) + MaxMBPS \* ( tr( 0 ) − tr,n( 0 ) ) ) ÷ MinCR, where MaxMBPS and MinCR are the values specified in Table A‑1 that apply to picture 0 and PicSizeInMbs is the number of macroblocks in a single texture view component of picture 0.
2. The sum of the NumBytesInNALunit variables for the VCL NAL units of each view component of access unit 0 is less than or equal to 384 \* ( Max( PicSizeInMbs, fR \* MaxMBPS ) + MaxMBPS \* ( tr( 0 ) − tr,n( 0 ) ) ) ÷ MinCR, where MaxMBPS and MinCR are the values specified in Table A‑1 that apply to picture 0 and PicSizeInMbs is the number of macroblocks in a single view component of picture 0.
3. The sum of the NumBytesInNALunit variables for access unit n with n > 0 is less than or equal to 384 \* mvcdScaleFactor \* MaxMBPS \* ( tr( n ) − tr( n − 1 ) ) ÷ MinCR, where MaxMBPS and MinCR are the values specified in Table A‑1 that apply to picture n.
4. The sum of the NumBytesInNALunit variables for the VCL NAL units of each view component of access unit n with n > 0 is less than or equal to 384 \* MaxMBPS \* ( tr( n ) − tr( n − 1 ) ) ÷ MinCR, where MaxMBPS and MinCR are the values specified in Table A‑1 that apply to picture n.
5. When PicSizeInMbs is greater than 1620, the number of macroblocks in any coded slice shall not exceed MaxFS / 4, where MaxFS is specified in Table A‑1.
6. max\_num\_ref\_frames shall be less than or equal to MaxDpbFrames / mvcScaleFactor for each texture view component, where MaxDpbFrames is specified in item f).
7. MVCD sequence parameter sets shall have frame\_mbs\_only\_flag equal to 1 for the levels specified in Table A‑4.

Table A‑1 specifies the limits for each level. A definition of all levels identified in the "Level number" column of Table A‑1 is specified for the Multiview Depth High profile. Table A‑4 specifies limits for each level that are specific to bitstreams conforming to the Multiview Depth High profile. Each entry in Table A‑1 and A‑4 indicates, for the level corresponding to the row of the table, the absence or value of a limit that is imposed by the variable corresponding to the column of the table, as follows:

– If the table entry is marked as "-", no limit is imposed by the value of the variable as a requirement of bitstream conformance to the profile at the specified level.

– Otherwise, the table entry specifies the value of the variable for the associated limit that is imposed as a requirement of bitstream conformance to the profile at the specified level.

For coded video sequences conforming to the Multiview Depth High profile, the level\_idc value is specified as follows:

– If level\_idc is not equal to 0, level\_idc indicates the level that applies to the coded video sequence operating with all the views being target output views.

NOTE 3 – A level\_idc value that is not equal to zero may indicate a higher level than necessary to decode the coded video sequence operating with all the views being target output views. This may occur when a subset of views or temporal subsets are removed from a coded video sequence according to the sub-bitstream extraction process specified in clause I.8.5.3, and the level\_idc value is not updated accordingly.

– Otherwise (level\_idc is equal to 0), the level that applies to the coded video sequence operating with all the views being target output views is unspecified.

NOTE 4 – When profile\_idc is equal to 118 or 128 and level\_idc is equal to 0, there may exist a level indicated by level\_idc[ i ] that is applicable to the coded video sequence operating with all the views being target output views. This may occur when a subset of views or temporal subsets are removed from a coded video sequence according to the sub-bitstream extraction process specified in clause I.8.5.3, and a particular value of level\_idc[ i ] corresponds to the resulting coded video sequence.

In bitstreams conforming to the Multiview Depth High profile, the conformance of the bitstream to a specified level is indicated by the syntax element level\_idc or level\_idc[ i ] as follows:

– If level\_idc or level\_idc[ i ] is equal to 9, the indicated level is level 1b.

– Otherwise (level\_idc or level\_idc[ i ] is not equal to 9), level\_idc or level\_idc[ i ] is equal to a value of ten times the level number (of the indicated level) specified in Table A‑1.

* + - 1. Profile specific level limits

1. In bitstreams conforming to the Multiview Depth High profile, MVCD sequence parameter sets shall have frame\_mbs\_only\_flag equal to 1 for the levels specified in Table A‑4.
   1. Byte stream format

The specifications in Annex B apply.

* 1. MVCD hypothetical reference decoder

The specifications in Annex C apply with substituting MVCD sequence parameter set for MVC sequence parameter set.

* 1. MVCD SEI messages

The specifications in Annex D together with the extensions and modifications specified in this clause apply.

* + 1. SEI message syntax
       1. MVCD view scalability information SEI message syntax

|  |  |  |
| --- | --- | --- |
| mvcd\_view\_scalability\_info( payloadSize ) { | C | Descriptor |
| **num\_operation\_points\_minus1** | 5 | ue(v) |
| for( i = 0; i <= num\_operation\_points\_minus1; i++ ) { |  |  |
| **operation\_point\_id[** i **]** | 5 | ue(v) |
| **priority\_id[**i**]** | 5 | u(5) |
| **temporal\_id[** i **]** | 5 | u(3) |
| **num\_target\_output\_views\_minus1[** i **]** | 5 | ue(v) |
| for( j = 0; j <= num\_target\_output\_views\_minus1[ i ]; j++ ) { |  |  |
| **view\_id[** i **][** j **]** | 5 | ue(v) |
| mvcd\_op\_view\_info( ) |  |  |
| } |  |  |
| **profile\_level\_info\_present\_flag[** i **]** | 5 | u(1) |
| **bitrate\_info\_present\_flag[** i **]** | 5 | u(1) |
| **frm\_rate\_info\_present\_flag[** i **]** | 5 | u(1) |
| if( !num\_target\_output\_views\_minus1[ i ] ) |  |  |
| **view\_dependency\_info\_present\_flag[**i**]** | 5 | u(1) |
| **parameter\_sets\_info\_present\_flag[** i **]** | 5 | u(1) |
| **bitstream\_restriction\_info\_present\_flag**[ i ] | 5 | u(1) |
| if( profile\_level\_info\_present\_flag[ i ] ) |  |  |
| **op\_profile\_level\_idc[** i **]** | 5 | u(24) |
| if( bitrate\_info\_present\_flag[ i ] ) { |  |  |
| **avg\_bitrate[** i **]** | 5 | u(16) |
| **max\_bitrate[** i **]** | 5 | u(16) |
| **max\_bitrate\_calc\_window[** i **]** | 5 | u(16) |
| } |  |  |
| if( frm\_rate\_info\_present\_flag[ i ] ) { |  |  |
| **constant\_frm\_rate\_idc[** i **]** | 5 | u(2) |
| **avg\_frm\_rate[** i **]** | 5 | u(16) |
| } |  |  |
| if( view\_dependency\_info\_present\_flag[ i ] ) { |  |  |
| **num\_directly\_dependent\_views[**i**]** | 5 | ue(v) |
| for( j = 0; j < num\_directly\_dependent\_views[ i ]; j++ ) { |  |  |
| **directly\_dependent\_view\_id[**i**][**j**]** | 5 | ue(v) |
| mvcd\_op\_view\_info( ) |  |  |
| } |  |  |
| } else |  |  |
| **view\_dependency\_info\_src\_op\_id[** i **]** | 5 | ue(v) |
| if( parameter\_sets\_info\_present\_flag[ i ] ) { |  |  |
| **num\_seq\_parameter\_set\_minus1[** i **]** | 5 | ue(v) |
| for( j = 0; j <= num\_seq\_parameter\_set\_minus1[ i ]; j++ ) |  |  |
| **seq\_parameter\_set\_id\_delta[** i **][** j **]** | 5 | ue(v) |
| **num\_subset\_seq\_parameter\_set\_minus1[** i **]** | 5 | ue(v) |
| for( j = 0; j <= num\_subset\_seq\_parameter\_set\_minus1[ i ]; j++ ) |  |  |
| **subset\_seq\_parameter\_set\_id\_delta[** i **][** j **]** | 5 | ue(v) |
| **num\_pic\_parameter\_set\_minus1[** i **]** | 5 | ue(v) |
| for( j = 0; j <= num\_init\_pic\_parameter\_set\_minus1[ i ]; j++ ) |  |  |
| **pic\_parameter\_set\_id\_delta[** i **][** j **]** | 5 | ue(v) |
| } else |  |  |
| **parameter\_sets\_info\_src\_op\_id[** i **]** | 5 | ue(v) |
| if( bitstream\_restriction\_info\_present\_flag[ i ] ) { |  |  |
| **motion\_vectors\_over\_pic\_boundaries\_flag[** i **]** | 5 | u(1) |
| **max\_bytes\_per\_pic\_denom[** i **]** | 5 | ue(v) |
| **max\_bits\_per\_mb\_denom[** i **]** | 5 | ue(v) |
| **log2\_max\_mv\_length\_horizontal[** i **]** | 5 | ue(v) |
| **log2\_max\_mv\_length\_vertical[** i **]** | 5 | ue(v) |
| **num\_reorder\_frames[** i **]** | 5 | ue(v) |
| **max\_dec\_frame\_buffering[** i **]** | 5 | ue(v) |
| } |  |  |
| } |  |  |
| } |  |  |

* + - * 1. MVCD operation point view information syntax

|  |  |  |
| --- | --- | --- |
| mvcd\_op\_view\_info( ) { | **C** | **Descriptor** |
| **view\_info\_depth\_view\_present\_flag** | 5 | u(1) |
| if( view\_info\_depth\_view\_present\_flag ) |  |  |
| **reserved\_depth\_view\_confirmation\_flag** | 5 | u(1) |
| **view\_info\_texture\_view\_present\_flag** | 5 | u(1) |
| if( view\_info\_texture\_view\_present\_flag ) |  |  |
| **reserved\_texture\_view\_confirmation\_flag** | 5 | u(1) |
| } |  |  |

* + - 1. MVCD scalable nesting SEI message syntax

|  |  |  |
| --- | --- | --- |
| mvcd\_scalable\_nesting( payloadSize ) { | C | Descriptor |
| **operation\_point\_flag** | 5 | u(1) |
| if( !operation\_point\_flag ) { |  |  |
| **all\_view\_components\_in\_au\_flag** | 5 | u(1) |
| if( !all\_view\_components\_in\_au\_flag ) { |  |  |
| **num\_view\_components\_minus1** | 5 | ue(v) |
| for( i = 0; i <= num\_view\_components\_minus1; i++ ) { |  |  |
| **sei\_view\_id[** i **]** | 5 | u(10) |
| **sei\_view\_applicability\_flag[** i **]** | 5 | u(1) |
| } |  |  |
| } |  |  |
| } else { |  |  |
| **sei\_op\_texture\_only\_flag** | 5 | u(1) |
| **num\_view\_components\_op\_minus1** | 5 | ue(v) |
| for( i = 0; i <= num\_view\_components\_op\_minus1; i++ ) { |  |  |
| **sei\_op\_view\_id[** i **]** | 5 | u(10) |
| if( !sei\_op\_texture\_only\_flag ) { |  |  |
| **sei\_op\_depth\_flag[** i **]** |  |  |
| **sei\_op\_texture\_flag[** i **]** |  |  |
| } |  |  |
| } |  |  |
| **sei\_op\_temporal\_id** | 5 | u(3) |
| } |  |  |
| while( !byte\_aligned( ) ) |  |  |
| **sei\_nesting\_zero\_bit** /\* equal to 0 \*/ | 5 | f(1) |
| sei\_message( ) | 5 |  |
| } |  |  |

* + - 1. Depth representation information SEI message syntax

|  |  |  |
| --- | --- | --- |
| depth\_representation\_info( payloadSize ) { | **C** | Descriptor |
| **all\_views\_equal\_flag** | 5 | u(1) |
| if( all\_views\_equal\_flag  = =  0 ) { |  |  |
| **num\_views\_minus1** | 5 | ue(v) |
| numViews = num\_views\_minus1 + 1 |  |  |
| } else |  |  |
| numViews = 1 |  |  |
| **z\_near\_flag** | 5 | u(1) |
| **z\_far\_flag** | 5 | u(1) |
| if( z\_near\_flag | | z\_far\_flag ) { |  |  |
| **z\_axis\_equal\_flag** | 5 | u(1) |
| if( z\_axis\_equal\_flag ) |  |  |
| **common\_z\_axis\_reference\_view** | 5 | ue(v) |
| } |  |  |
| **d\_min\_flag** | 5 | u(1) |
| **d\_max\_flag** | 5 | u(1) |
| **depth\_representation\_type** | 5 | ue(v) |
| for( i = 0; i < numViews; i++ ) { |  |  |
| **depth\_info\_view\_id**[ i ] | 5 | ue(v) |
| if( ( z\_near\_flag | | z\_far\_flag ) && ( z\_axis\_equal\_flag = = 0 ) ) |  |  |
| **z\_axis\_reference\_view**[ i ] | 5 | ue(v) |
| if( d\_min\_flag | | d\_max\_flag ) |  |  |
| **disparity\_reference\_view**[ i ] | 5 | ue(v) |
| if( z\_near\_flag ) |  |  |
| depth\_representation\_sei\_element( ZNearSign, ZNearExp,  ZNearMantissa, ZNearManLen ) |  |  |
| if( z\_far\_flag ) |  |  |
| depth\_representation\_sei\_element( ZFarSign, ZFarExp,  ZFarMantissa, ZFarManLen ) |  |  |
| if( d\_min\_flag ) |  |  |
| depth\_representation\_sei\_element( DMinSign, DMinExp,  DMinMantissa, DMinManLen ) |  |  |
| if( d\_max\_flag ) |  |  |
| depth\_representation\_sei\_element( DMaxSign, DMaxExp,  DMaxMantissa, DMaxManLen ) |  |  |
| } |  |  |
| if( depth\_representation\_type = = 3 ) { |  |  |
| **depth\_nonlinear\_representation\_num\_minus1** | 5 | ue(v) |
| for( i = 1; i <= depth\_nonlinear\_representation\_num\_minus1 + 1; i++ ) |  |  |
| **depth\_nonlinear\_representation\_model**[ i ] | 5 | ue(v) |
| } |  |  |
| } |  |  |

* + - 1. Depth representation SEI element syntax

|  |  |  |
| --- | --- | --- |
| depth\_representation\_sei\_element( OutSign, OutExp, OutMantissa,  OutManLen ) { | **C** | Descriptor |
| **da\_sign\_flag** | 5 | u(1) |
| **da\_exponent** | 5 | u(7) |
| **da\_matissa\_len\_minus1** | 5 | u(5) |
| **da\_mantissa** | 5 | u(v) |
| } |  |  |

* + - 1. 3D reference displays information SEI message syntax

|  |  |  |
| --- | --- | --- |
| three\_dimensional\_reference\_displays\_info( payloadSize ) { | **C** | **Descriptor** |
| **prec\_ref\_baseline** | 5 | ue(v) |
| **prec\_ref\_display\_width** | 5 | ue(v) |
| **ref\_viewing\_distance\_flag** | 5 | u(1) |
| if( ref\_viewing\_distance\_flag ) |  |  |
| **prec\_ref\_viewing\_dist** | 5 | ue(v) |
| **num\_ref\_displays\_minus1** | 5 | ue(v) |
| numRefDisplays = num\_ref\_displays\_minus1 + 1 |  |  |
| for( i = 0; i < numRefDisplays; i++ ) { |  |  |
| **exponent\_ref\_baseline[**i**]** | 5 | u(6) |
| **mantissa\_ref\_baseline[**i**]** | 5 | u(v) |
| **exponent\_ref\_display\_width[**i**]** | 5 | u(6) |
| **mantissa\_ref\_display\_width[**i**]** | 5 | u(v) |
| if( ref\_viewing\_distance\_flag ) { |  |  |
| **exponent\_ref\_viewing\_distance[**i**]** | 5 | u(6) |
| **mantissa\_ref\_viewing\_distance[**i**]** | 5 | u(v) |
| **}** |  |  |
| **additional\_shift\_present\_flag**[ i ] | 5 | u(1) |
| if( additional\_shift\_present[ i ] ) |  |  |
| **num\_sample\_shift\_plus512**[ i ] | 5 | u(10) |
| } |  |  |
| **three\_dimensional\_reference\_displays\_extension\_flag** | 5 | u(1) |
| } |  |  |

* + - 1. Depth timing SEI message syntax

|  |  |  |
| --- | --- | --- |
| depth\_timing( payloadSize ) { | C | Descriptor |
| **per\_view\_depth\_timing\_flag** | 5 | u(1) |
| if( per\_view\_depth\_timing\_flag ) |  |  |
| for( i = 0; i < NumDepthViews; i++ ) |  |  |
| depth\_timing\_offset( ) |  |  |
| else |  |  |
| depth\_timing\_offset( ) |  |  |
| } |  |  |

* + - * 1. Depth timing offset syntax

|  |  |  |
| --- | --- | --- |
| depth\_timing\_offset( ) { | **C** | **Descriptor** |
| **offset\_len\_minus1** | 5 | u(5) |
| **depth\_disp\_delay\_offset\_fp** | 5 | u(v) |
| **depth\_disp\_delay\_offset\_dp** | 5 | u(6) |
| } |  |  |

* + - 1. Depth sampling information SEI message syntax

|  |  |  |
| --- | --- | --- |
| depth\_sampling\_info( payloadSize ) { | **C** | **Descriptor** |
| **dttsr\_x\_mul** | 5 | u(16) |
| **dttsr\_x\_dp** | 5 | u(4) |
| **dttsr\_y\_mul** | 5 | u(16) |
| **dttsr\_y\_dp** | 5 | u(4) |
| **per\_view\_depth\_grid\_pos\_flag** | 5 | u(1) |
| if( per\_view\_depth\_grid\_pos\_flag ) { |  |  |
| **num\_video\_plus\_depth\_views\_minus1** | 5 | ue(v) |
| for( i = 0; i <= num\_video\_plus\_depth\_views\_minus1; i++ ) { |  |  |
| **depth\_grid\_view\_id**[ i ] | 5 | ue(v) |
| depth\_grid\_position( ) |  |  |
| } |  |  |
| } else |  |  |
| depth\_grid\_position( ) |  |  |
| } |  |  |

* + - * 1. Depth grid position syntax

|  |  |  |
| --- | --- | --- |
| depth\_grid\_position( ) { | **C** | **Descriptor** |
| **depth\_grid\_pos\_x\_fp** | 5 | u(20) |
| **depth\_grid\_pos\_x\_dp** | 5 | u(4) |
| **depth\_grid\_pos\_x\_sign\_flag** | 5 | u(1) |
| **depth\_grid\_pos\_y\_fp** | 5 | u(20) |
| **depth\_grid\_pos\_y\_dp** | 5 | u(4) |
| **depth\_grid\_pos\_y\_sign\_flag** | 5 | u(1) |
| } |  |  |

* + 1. SEI message semantics

Depending on payloadType, the corresponding SEI message semantics are extended as follows:

– If payloadType is equal to 2, 3, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 45 or 47, the following applies:

– If the SEI message is not included in an MVC scalable nesting SEI message or a MVCD scalable nesting SEI message, it applies to the texture view component of the current access unit with VOIdx equal to VOIdxMin.

– Otherwise, if included in an MVC scalable nesting SEI message and not included in a MVCD scalable nesting SEI message, it applies to all texture view components of the current access unit when all\_view\_components\_in\_au\_flag is equal to 1, or it applies to all texture view components of the current access unit with view\_id equal to sei\_view\_id[ i ] for any i in the range of 0 to num\_view\_components\_minus1, inclusive, when all\_view\_components\_in\_au\_flag is equal to 0. When payloadType is equal to 10 for the SEI message that is included in an MVC scalable nesting SEI message, the semantics for sub\_seq\_layer\_num of the sub-sequence information SEI message is modified as follows:

**sub\_seq\_layer\_num** specifies the sub-sequence layer number of the current picture. When the current picture resides in a sub-sequence for which the first picture in decoding order is an IDR picture, the value of sub\_seq\_layer\_num shall be equal to 0. For a non-paired reference field, the value of sub\_seq\_layer\_num shall be equal to 0. sub\_seq\_layer\_num shall be in the range of 0 to 255, inclusive.

– Otherwise, if not included in an MVC scalable nesting SEI message and included in an MVCD scalable nesting SEI message, it applies to all depth view components or view component pairs of the current access unit when all\_view\_components\_in\_au\_flag is equal to 1, or it applies to all depth view components or view component pairs of the current access unit with view\_id equal to sei\_view\_id[ i ] for any i in the range of 0 to num\_view\_components\_minus1, inclusive, when all\_view\_components\_in\_au\_flag is equal to 0. When payloadType is equal to 10 for the SEI message that is included in an MVCD scalable nesting SEI message, the semantics for sub\_seq\_layer\_num of the sub-sequence information SEI message is modified as follows:

**sub\_seq\_layer\_num** specifies the sub-sequence layer number of the current picture. When the current picture resides in a sub-sequence for which the first picture in decoding order is an IDR picture, the value of sub\_seq\_layer\_num shall be equal to 0. For a non-paired reference field, the value of sub\_seq\_layer\_num shall be equal to 0. sub\_seq\_layer\_num shall be in the range of 0 to 255, inclusive.

– Otherwise, if payloadType is equal to 41, 42 or 43, the following applies:

– If the SEI message is not included in MVCD scalable nesting SEI message, it applies to texture views only and NAL units having nal\_unit\_type equal to 21 are non-VCL NAL units.

– Otherwise (the SEI message is included in MVCD scalable nesting SEI message), the SEI message applies to depth views, to texture views or both texture views all depth views, depending on the values of the syntax elements of the MVCD scalable nesting SEI message.

– Otherwise, if payloadType is equal to 0 or 1, the following applies:

– If the SEI message is not included in an MVC scalable nesting SEI message or a MVCD scalable nesting SEI message or a MVCD texture sub-bitstream HRD nesting SEI message, the following applies. When the SEI message and all other SEI messages with payloadType equal to 0 or 1 not included in an MVC scalable nesting SEI message or a MVCD scalable nesting SEI message or a MVCD texture sub-bitstream HRD nesting SEI message are used as the buffering period and picture timing SEI messages for checking the bitstream conformance according to Annex C and the decoding process specified in clauses 2-9 is used, the bitstream shall be conforming to this Recommendation | International Standard.

– Otherwise, if the SEI message is included in an MVC scalable nesting SEI message and not included in a MVCD scalable nesting SEI message or a MVCD texture sub-bitstream HRD nesting SEI message, the following applies. When the SEI message and all other SEI messages with payloadType equal to 0 or 1 included in an MVC scalable nesting SEI message with identical values of sei\_op\_temporal\_id and sei\_op\_view\_id[ i ] for all i in the range of 0 to num\_view\_components\_op\_minus1, inclusive, are used as the buffering period and picture timing SEI messages for checking the bitstream conformance according to Annex C, the bitstream that would be obtained by invoking the bitstream extraction process as specified in clause H.8.3 with tIdTarget equal to sei\_op\_temporal\_id and viewIdTargetList equal to sei\_op\_view\_id[ i ] for all i in the range of 0 to num\_view\_components\_op\_minus1, inclusive, shall be conforming to this Recommendation | International Standard.

In the semantics of clauses D.2.1 and D.2.2, the syntax elements num\_units\_in\_tick, time\_scale, fixed\_frame\_rate\_flag, nal\_hrd\_parameters\_present\_flag, vcl\_hrd\_parameters\_present\_flag, low\_delay\_hrd\_flag, and pic\_struct\_present\_flag and the derived variables NalHrdBpPresentFlag, VclHrdBpPresentFlag, and CpbDpbDelaysPresentFlag are substituted with the syntax elements vui\_mvc\_num\_units\_in\_tick[ i ], vui\_mvc\_time\_scale[ i ], vui\_mvc\_fixed\_frame\_rate\_flag[ i ], vui\_mvc\_nal\_hrd\_parameters\_present\_flag[ i ], vui\_mvc\_vcl\_hrd\_parameters\_present\_flag[ i ], vui\_mvc\_low\_delay\_hrd\_flag[ i ], and vui\_mvc\_pic\_struct\_present\_flag[ i ] and the derived variables VuiMvcNalHrdBpPresentFlag[ i ], VuiMvcVclHrdBpPresentFlag[ i ], and VuiMvcCpbDpbDelaysPresentFlag[ i ].

The values of seq\_parameter\_set\_id's in all buffering period SEI messages included in MVC scalable nesting SEI messages and associated with operation points for which the greatest VOIdx values in the associated bitstream subsets are identical shall be identical.

– Otherwise, if the SEI message is included in a MVCD scalable nesting SEI message and not included in an MVC scalable nesting SEI message or a MVCD texture sub-bitstream HRD nesting SEI message, the following applies. When the SEI message and all other SEI messages with payloadType equal to 0 or 1 included in a MVCD scalable nesting SEI message with identical values of sei\_op\_temporal\_id and sei\_op\_view\_id[ i ] for all i in the range of 0 to num\_view\_components\_op\_minus1, inclusive, are used as the buffering period and picture timing SEI messages for checking the bitstream conformance according to Annex C, the bitstream that would be obtained by invoking the bitstream extraction process as specified in clause I.8.5 with depthPresentTargetFlag equal to 1, tIdTarget equal to sei\_op\_temporal\_id and viewIdTargetList equal to sei\_op\_view\_id[ i ] for all i in the range of 0 to num\_view\_components\_op\_minus1, inclusive, shall be conforming to this Recommendation | International Standard.

In the semantics of clauses D.2.1 and D.2.2, the syntax elements num\_units\_in\_tick, time\_scale, fixed\_frame\_rate\_flag, nal\_hrd\_parameters\_present\_flag, vcl\_hrd\_parameters\_present\_flag, low\_delay\_hrd\_flag, and pic\_struct\_present\_flag and the derived variables NalHrdBpPresentFlag, VclHrdBpPresentFlag, and CpbDpbDelaysPresentFlag are substituted with the syntax elements vui\_mvc\_num\_units\_in\_tick[ i ], vui\_mvc\_time\_scale[ i ], vui\_mvc\_fixed\_frame\_rate\_flag[ i ], vui\_mvc\_nal\_hrd\_parameters\_present\_flag[ i ], vui\_mvc\_vcl\_hrd\_parameters\_present\_flag[ i ], vui\_mvc\_low\_delay\_hrd\_flag[ i ], and vui\_mvc\_pic\_struct\_present\_flag[ i ] and the derived variables VuiMvcNalHrdBpPresentFlag[ i ], VuiMvcVclHrdBpPresentFlag[ i ], and VuiMvcCpbDpbDelaysPresentFlag[ i ] for the MVCD VUI parameters extension.

The values of seq\_parameter\_set\_id's in all buffering period SEI messages included in MVCD scalable nesting SEI messages and not included in either MVC scalable nesting SEI messages or MVCD texture sub-bitstream HRD nesting SEI messages and associated with operation points for which the greatest VOIdx values in the associated bitstream subsets are identical shall be identical.

– Otherwise, if the SEI message is included in a MVCD texture sub-bitstream HRD nesting SEI message, the following applies. When the SEI message and all other SEI messages included in a MVCD texture sub-bitstream HRD nesting SEI message with identical values of texture\_subbitstream\_temporal\_id and texture\_subbitstream\_view\_id[ i ] for all i in the range of 0 to num\_texture\_subbitstream\_view\_components\_minus1, inclusive, are used as the buffering period and picture timing SEI messages for checking the bitstream conformance according to Annex C, the bitstream that would be obtained by invoking the bitstream extraction process as specified in clause I.8.5 with depthPresentTargetFlag equal to 0, tIdTarget equal to texture\_subbitstream\_temporal\_id and viewIdTargetList equal to texture\_subbitstream\_view\_id[ i ] for all i in the range of 0 to num\_texture\_subbitstream\_view\_components\_minus1, inclusive, shall be conforming to this Recommendation | International Standard.

In the semantics of clauses D.2.1 and D.2.2, the syntax elements num\_units\_in\_tick, time\_scale, fixed\_frame\_rate\_flag, nal\_hrd\_parameters\_present\_flag, vcl\_hrd\_parameters\_present\_flag, low\_delay\_hrd\_flag, and pic\_struct\_present\_flag and the derived variables NalHrdBpPresentFlag, VclHrdBpPresentFlag, and CpbDpbDelaysPresentFlag are substituted with the syntax elements vui\_mvc\_num\_units\_in\_tick[ i ], vui\_mvc\_time\_scale[ i ], vui\_mvc\_fixed\_frame\_rate\_flag[ i ], vui\_mvc\_nal\_hrd\_parameters\_present\_flag[ i ], vui\_mvc\_vcl\_hrd\_parameters\_present\_flag[ i ], vui\_mvc\_low\_delay\_hrd\_flag[ i ], and vui\_mvc\_pic\_struct\_present\_flag[ i ] and the derived variables VuiMvcNalHrdBpPresentFlag[ i ], VuiMvcVclHrdBpPresentFlag[ i ], and VuiMvcCpbDpbDelaysPresentFlag[ i ] for the MVCD texture sub-bitstream VUI parameters extension.

The values of seq\_parameter\_set\_id's in all buffering period SEI messages included in MVCD texture sub-bitstream HRD nesting SEI messages and associated with operation points for which the greatest VOIdx values in the associated bitstream subsets are identical shall be identical.

– Otherwise (all remaining payloadType values), the corresponding SEI message semantics are not extended.

For the semantics of SEI messages with payloadType in the range of 0 to 23, inclusive, or equal to 45 or 47, which are specified in clause D.2, MVCD sequence parameter set is substituted for sequence parameter set; the parameters of MVCD sequence parameter set RBSP and picture parameter set RBSP that are in effect are specified in clauses I.7.4.2.1and I.7.4.2.2, respectively.

Coded video sequences conforming to one or more of the profiles specified in Annex I shall not include SEI NAL units that contain SEI messages with payloadType in the range of 24 to 35, inclusive.

When an SEI NAL unit contains an SEI message with payloadType in the range of 36 to 44, inclusive, or equal to 46, or in the range of 48 to 53, inclusive, it shall not contain any SEI messages with payloadType less than 36 and the first SEI message in the SEI NAL unit shall have payloadType in the range of 36 to 44, inclusive, or equal to 46, or in the range of  48 to 53, inclusive.

When an MVC scalable nesting SEI message (payloadType equal to 37), a view scalability information SEI message (payloadType equal to 38), or an operation point not present SEI message (payloadType equal to 43), an MVCD scalable nesting SEI message (payloadType equal to 48), or an MVCD view scalability information SEI messages (payloadType equal to 49) is present in an SEI NAL unit, it shall be the only SEI message in the SEI NAL unit.

* + - 1. MVCD view scalability information SEI message semantics

The syntax elements in the MVCD view scalability information SEI message that have the same names as those in the view scalability information SEI message specified in Annex H, except num\_directly\_dependent\_depth\_views[ i ] and directly\_dependent\_depth\_view\_id[ i ][ j ], have the same semantics as the corresponding syntax elements in the view scalability information SEI message, but apply to operation points that may potentially contain depth view components as well as texture view components.

**num\_directly\_dependent\_views[** i **]** and **directly\_dependent\_view\_id[** i **][** j **]** apply only to the texture view components of an operation point if the operation point contains both texture and depth, and otherwise have the same semantics as the corresponding syntax elements in the view scalability information SEI message.

* + - * 1. MVCD operation point view information semantics

**view\_info\_depth\_view\_present\_flag** equal to 0 specifies that the depth view is not included in the operation point for the view for which the mvcd\_op\_view\_info( ) syntax structure is present. view\_info\_depth\_view\_present\_flag equal to 1 specifies that the depth view is included in the operation point for the view for which the mvcd\_op\_view\_info( ) syntax structure is present.

**reserved\_depth\_view\_confirmation\_flag** shall be equal to 1 in bitstreams conforming to this version of this Specification. The value 0 for reserved\_depth\_view\_confirmation\_flag is reserved for future specification by ITU-T | ISO/IEC. Decoders conforming to this version of this Specification shall interpret the value 0 for reserved\_depth\_view\_confirmation\_flag as being equivalent to having view\_info\_depth\_view\_present\_flag equal to 0.

**view\_info\_texture\_view\_present\_flag** equal to 0 specifies that the texture view is not included in the operation point for the view for which the mvcd\_op\_view\_info( ) syntax structure is present. view\_info\_depth\_view\_present\_flag equal to 1 specifies that the texture view is included in the operation point for the view for which the mvcd\_op\_view\_info( ) syntax structure is present. When view\_info\_depth\_view\_present\_flag is equal to 0, view\_info\_texture\_view\_present\_flag shall be equal to 1.

**reserved\_texture\_view\_confirmation\_flag** shall be equal to 1 in bitstreams conforming to this version of this Specification. The value 0 for reserved\_texture\_view\_confirmation\_flag is reserved for future specification by ITU-T | ISO/IEC. Decoders shall ignore the value of reserved\_depth\_view\_confirmation\_flag. Decoders conforming to this version of this Specification shall interpret the value 0 for reserved\_texture\_view\_confirmation\_flag as being equivalent to having view\_info\_texture\_view\_present\_flag equal to 0.

* + - 1. MVCD scalable nesting SEI message semantics

The syntax elements in the MVCD scalable nesting SEI message have the same semantics as the ones with the same names and present in the MVC scalable nesting SEI message in Annex H.

**sei\_view\_applicability\_flag[** i **]** equal to 1 indicates that the nested SEI message applies to both the texture view component and the depth view component of the view with view\_id equal to sei\_view\_id[ i ]. sei\_view\_applicability\_flag[ i ] equal to 0 indicates that the nested SEI message applies only to the depth view component of the view with view\_id equal to sei\_view\_id[ i ].

**sei\_op\_texture\_only\_flag** equal to 0 specifies that the semantics of sei\_op\_view\_id[ i ] and sei\_op\_temporal\_id apply to both texture and depth views, if present. sei\_op\_texture\_only\_flag equal to 1 specifies that the nested SEI message as well as the semantics of sei\_op\_view\_id[ i ] and sei\_op\_temporal\_id apply to the sub-bitstream obtained by the sub-bitstream extraction process of clause I.8.5.3 with depthPresentFlagTarget equal to 0, tIdTarget equal to sei\_op\_temporal\_id, and viewIdTargetList equal to sei\_op\_view\_id[ i ] for all values of i in the range of 0 to num\_view\_components\_op\_minus1, inclusive, as inputs.

NOTE 1 – MVC scalable nesting SEI message should be used for nesting SEI messages, when depth views may or may not be present in the bitstream, the nested SEI messages apply only to indicated texture view components and the semantics of the nested SEI messages apply when VCL and non-VCL NAL units are classified according to Annex H NAL unit type class of Table 7‑1.

NOTE 2 – MVCD scalable nesting SEI message with sei\_op\_texture\_only\_flag equal to 1 should be used when the nested SEI messages concern a sub-bitstream from which depth views have been excluded. For example, MVCD scalable nesting SEI message with sei\_op\_texture\_only\_flag equal to 1 may include buffering period and picture timing SEI messages which apply only to a sub-bitstream containing texture views from which depth views have been removed using the sub-bitstream extraction process of clause I.8.5.3 with depthPresentFlagTarget equal to 0.

**sei\_op\_depth\_flag**[ i ] equal to 0 specifies that the depth view with view\_id equal to sei\_op\_view\_id[ i ] is not included in the operation point to which the nested SEI message applies. sei\_op\_depth\_flag[ i ] equal to 1 specifies that the depth view with view\_id equal to sei\_op\_view\_id[ i ] is included in the operation point to which the nested SEI message applies. If sei\_op\_depth\_flag[ i ] is not present, it is inferred to be equal to 1.

**sei\_op\_texture\_flag**[ i ] equal to 0 specifies that the texture view with view\_id equal to sei\_op\_view\_id[ i ] is not included in the operation point to which the nested SEI message applies. sei\_op\_texture\_flag[ i ] equal to 1 specifies that the texture view with view\_id equal to sei\_op\_view\_id[ i ] is included in the operation point to which the nested SEI message applies. If sei\_op\_texture\_flag[ i ] is not present, it is inferred to be equal to 1. When sei\_op\_depth\_flag[ i ] is equal to 0, sei\_op\_texture\_flag[ i ] shall be equal to 1.

* + - 1. Depth representation information SEI message semantics

The syntax elements in the depth representation information SEI message specifies various parameters for depth views for the purpose of processing decoded texture and depth view components prior to rendering on a 3D display, such as view synthesis. Specifically, depth or disparity ranges for depth views are specified. When present, the depth representation information SEI message may be associated with any access unit. It is recommended, when present, the SEI message is associated with an IDR access unit for the purpose of random access. The information indicated in the SEI message applies to all the access units from the access unit the SEI message is associated with to the next access unit, in decoding order, containing an SEI message of the same type, exclusive, or to the end of the coded video sequence, whichever is earlier in decoding order.

NOTE – Camera parameters for depth views may be indicated by including a multiview acquisition information SEI message in a MVCD scalable nesting SEI message with operation\_point\_flag equal to 0.

**all\_views\_equal\_flag** equal to 0 specifies that depth acquisition information may not be identical to respective values for each view in target views. all\_views\_equal\_flag equal to 1 specifies that the depth acquisition information are identical to respective values for all target views.

**num\_views\_minus1** plus 1 specifies the number of views to which subsequent syntax element apply. When present, num\_views\_minus1 shall be less than or equal to NumDepthViews − 1. The value of num\_views\_minus1 shall be in the range of 0 to 1023, inclusive.

**z\_near\_flag** equal to 0 specifies that the syntax elements specifying the nearest depth value are not present in the syntax structure. z\_near\_flag equal to 1 specifies that the syntax elements specifying the nearest depth value are present in the syntax structure.

**z\_far\_flag** equal to 0 specifies that the syntax elements specifying the farthest depth value are not present in the syntax structure. z\_far\_flag equal to 1 specifies that the syntax elements specifying the farthest depth value are present in the syntax structure.

**z\_axis\_equal\_flag** equal to 0 specifies that the syntax element z\_axis\_reference\_view[ i ] is present. z\_axis\_equal\_flag equal to 1 specifies that the ZNear and ZFar values, when present, and the decoded samples of depth views, when depth\_representation\_type is equal to 0 or 2, have the same Z-axis, which is the Z-axis of the depth view indicated by the syntax element common\_z\_axis\_reference view.

**common\_z\_axis\_reference\_view** specifies the view\_id value of the depth view of the Z-axis of the ZNear and ZFar values, when present, and the decoded samples of depth views, when depth\_representation\_type is equal to 0 or 2. The value of common\_z\_axis\_reference\_view shall be in the range of 0 to 1023, inclusive.

**d\_min\_flag** equal to 0 specifies that the syntax elements specifying the minimum disparity value are not present in the syntax structure. d\_min\_flag equal to 1 specifies that the syntax elements specifying the minimum disparity value are present in the syntax structure.

**d\_max\_flag** equal to 0 specifies that the syntax elements specifying the maximum disparity value are not present in the syntax structure. d\_max\_flag equal to 1 specifies that the syntax elements specifying the maximum disparity value are present in the syntax structure.

**depth\_representation\_type** specifies the representation definition of decoded luma samples of depth views as specified in Table I‑1. In Table I‑1, disparity specifies the horizontal displacement between two texture views and Z value specifies the distance from a camera.

Table I‑1 – Definition of depth\_representation\_type

|  |  |
| --- | --- |
| depth\_representation\_type | Interpretation |
| 0 | Each decoded luma sample value of depth views represents an inverse of Z value that is uniformly quantized into the range of 0 to 255, inclusive. |
| 1 | Each decoded luma sample value of depth views represents disparity that is uniformly quantized into the range of 0 to 255, inclusive. |
| 2 | Each decoded luma sample value of depth views represents a Z value uniformly quantized into the range of 0 to 255, inclusive. |
| 3 | Each decoded luma sample value of depth views represents a nonlinearly mapped disparity, normalized in range from 0 to 255, as specified by depth\_nonlinear\_representation\_num\_minus1 and depth\_nonlinear\_representation\_model[ i ]. |
| Other values | Reserved for future use |

**depth\_info\_view\_id**[ i ] specifies the view\_id value for which subsequent syntax elements apply to. The value of depth\_info\_view\_id[ i ] shall be in the range of 0 to 1023, inclusive.

**z\_axis\_reference\_view**[ i ] specifies the view\_id value of the depth view of the Z-axis of the ZNear[ i ] and ZFar[ i ] values, when present, and the decoded samples of the depth view with view\_id equal to depth\_info\_view\_id[ i ], when depth\_representation\_type is equal to 0 or 2. The value of z\_axis\_reference\_view[ i ] shall be in the range of 0 to 1023, inclusive.

**disparity\_reference\_view**[ i ] specifies the view\_id value of the depth view used to derive the DMin[ i ] and Dmax[ i ] values, when present, and the decoded samples of the depth view with view\_id equal to depth\_info\_view\_id[ i ], when depth\_representation\_type is equal to 1 or 3. The value of disparity\_reference\_view[ i ] shall be in the range of 0 to 1023, inclusive.

The variables in the x column of Table I‑2 are derived as follows from the respective variables in the s, e, n, and v columns of Table I‑2 as follows.

– If 0 < e < 127, x = ( −1 )s \* 2e−31 \* ( 1 + n ÷ 2v ).

– Otherwise (e is equal to 0), x = ( −1 )s \* 2−( 30+v ) \* n.

NOTE – The above specification is similar to that found in IEC 60559.

Table I‑2 – Association between depth parameter variables and syntax elements

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **x** | **s** | **e** | **n** | **v** |
| ZNear[ vId ] | ZNearSign[ vId ] | ZNearExp[ vId ] | ZNearMantissa[ vId ] | ZNearManLen[ vId ] |
| ZFar[ vId ] | ZFarSign[ vId ] | ZFarExp[ vId ] | ZFarMantissa[ vId ] | ZFarManLen[ vId ] |
| DMax[ vId ] | DMaxSign[ vId ] | DMaxExp[ vId ] | DMaxMantissa[ vId ] | DMaxManLen[ vId ] |
| DMin[ vId ] | DMinSign[ vId ] | DMinExp[ vId ] | DMinMantissa[ vId ] | DMinManLen[ vId ] |

If all\_views\_equal\_flag is equal to 0, the variables x in Table I‑2 are specified as follows:

* ZNear[ vId ]: The closest depth value for view\_id equal to vId.
* ZFar[ vId ]: The farthest depth value for view\_id equal to vId.
* DMax[ vId ]: The maximum disparity value for view\_id equal to vId.
* DMin[ vId ]: The minimum disparity value for view\_id equal to vId.

Otherwise, the variables x in Table I‑2 are specified as follows:

* ZNear[ 0 ]: The closest depth value for all depth views.
* ZFar[ 0 ]: The farthest depth value for all depth views.
* DMax[ 0 ]: The maximum disparity value for all depth views.
* DMin[ 0 ]: The minimum disparity value for all depth views.

The DMin and DMax values, when present, are specified in units of a luma sample width of the texture views.

The ZNear and ZFar values, when present, are specified in units of a unit vector of the 3-dimensional coordinate system used to specify the extrinsic camera parameters as specified by the Multiview Acquisition Information SEI message associated with the respective depth views, if present. Otherwise, ZNear and ZFar values, when present, are specified in units of a unit vector of the 3-dimensional coordinate system used to specify the extrinsic camera parameters specified by the Multiview Acquisition Information SEI message associated with the respective texture views, if present. Otherwise, the units for the ZNear and ZFar values, if present, are identical but unspecified.

**depth\_nonlinear\_representation\_num\_minus1** + 2 specifies the number of piecewise linear segments for mapping of depth values to a scale that is uniformly quantized in terms of disparity.

**depth\_nonlinear\_representation\_model**[ i ] specifies the piecewise linear segments for mapping of decoded luma sample values of depth views to a scale that is uniformly quantized in terms of disparity.

NOTE – When depth\_representation\_type is equal to 3, depth view component contains nonlinearly transformed depth samples. Variable DepthLUT[ i ], as specified below, is used to transform coded depth sample values from nonlinear representation to the linear representation – disparity normalized in range from 0 to 255. The shape of this transform is defined by means of line-segment-approximation in two-dimensional linear-disparity-to-nonlinear-disparity space. The first (0, 0) and the last (255, 255) nodes of the curve are predefined. Positions of additional nodes are transmitted in form of deviations (depth\_nonlinear\_representation\_model[ i ]) from the straight-line curve. These deviations are uniformly distributed along the whole range of 0 to 255, inclusive, with spacing depending on the value of nonlinear\_depth\_representation\_num\_minus1.

Variable DepthLUT[ i ] for i in the range of 0 to 255, inclusive, is specified as follows.

depth\_nonlinear\_representation\_model[ 0 ] = 0  
depth\_nonlinear\_representation\_model[depth\_nonlinear\_representation\_num\_minus1 + 2 ] = 0  
for( k=0; k<= depth\_nonlinear\_representation\_num\_minus1 + 1; ++k ) {  
 pos1 = ( 255 \* k ) / (depth\_nonlinear\_representation\_num\_minus1 + 2 )  
 dev1 = depth\_nonlinear\_representation\_model[ k ]  
 pos2 = ( 255 \* ( k+1 ) ) / (depth\_nonlinear\_representation\_num\_minus1 + 2 ) )  
 dev2 = depth\_nonlinear\_representation\_model[ k+1 ]  
  
 x1 = pos1 − dev1  
 y1 = pos1 + dev1  
 x2 = pos2 − dev2  
 y2 = pos2 + dev2  
  
 for ( x = max( x1, 0 ); x <= min( x2, 255 ); ++x )  
 DepthLUT[ x ] = Clip3( 0, 255, Round( ( ( x - x1 ) \* ( y2 - y1 ) ) ÷ ( x2 - x1 ) + y1 ) )  
}

When depth\_representation\_type is equal to 3, DepthLUT[ dS ] for all decoded luma sample values dS of depth views in the range of 0 to 255, inclusive, represents disparity that is uniformly quantized into the range of 0 to 255, inclusive.

* + - 1. Depth representation SEI element semantics

The syntax structure specifies the value of an element in depth representation information.

The syntax structure sets the values of the OutSign, OutExp, OutMantissa, and OutManLen variables that represent a floating-point value. When the syntax structure is included in another syntax structure, the variable names OutSign, OutExp, OutMantissa, and OutManLen are to be interpreted as being replaced by the variable names used when the syntax structure is included.

**da\_sign\_flag** equal to 0 indicates that the sign of the floating-point value is positive. da\_sign\_flag equal to 1 indicates that the sign is negative. The variable OutSign is set equal to da\_sign\_flag.

**da\_exponent** specifies the exponent of the floating-point value. The value of da\_exponent shall be in the range of 0 to 27 − 2, inclusive. The value 27 − 1 is reserved for future use by ITU‑T | ISO/IEC. Decoders shall treat the value 27 − 1 as indicating an unspecified value. The variable OutExp is set equal to da\_exponent.

**da\_mantissa\_len\_minus1** + 1 specifies the number of bits in the da\_mantissa syntax element. The value of da\_mantissa\_len\_minus1 shall be in the range of 0 to 31, inclusive. The variable OutManLen is set equal to da\_mantissa\_len\_minus + 1.

**da\_mantissa** specifies the mantissa of the floating-point value. The variable OutMantissa is set equal to da\_mantissa.

* + - 1. 3D reference displays information SEI message semantics

When present, this SEI message shall be associated with an IDR access unit. A reference displays information message contains information about the reference display width(s) and reference viewing distance(s) as well as information about the corresponding baseline distance(s) and additional horizontal image shift(s), which form a stereo-pair for the reference display width and the reference viewing distance. This information enables a view renderer to produce a proper stereo-pair for the target screen width and the viewing distance. The reference display width and viewing distance values are signalled in units of centimetres. The reference baseline values shall be signalled in the same units as the x component of the translation vector in the depth acquisition information SEI message that is valid for the same access unit. When a reference displays information SEI message is present in an access unit, the depth acquisition information SEI message shall also be present in the same access unit. The baseline and shift information signalled for the reference display is valid for all access units they associated with and until the next IDR access unit or the next access unit containing depth acquisition information SEI message or reference displays information SEI message.

NOTE – The reference displays information SEI message specifies display parameters for which the 3D sequence was optimized and the corresponding reference parameters. Each reference display (i.e. a reference display width and possibly a corresponding viewing distance) is associated with one reference baseline distance.

The following formulas can be used for calculating the baseline distance and horizontal shift for the receiver's display when the ratio between the receiver's viewing distance and the reference viewing distance is the same as the ratio between the receiver screen width and the reference screen width:

baseline = ref\_baseline \* ( ref\_display\_width ÷ display\_width )

shift = ref\_shift \* ( ref\_display\_width ÷ display\_width )

In the provided formulas, the width of the visible part of the display used for showing the video sequence should be understood under "display width". The same formulas can also be used for choosing the baseline distance and horizontal shift in cases when the viewing distance is not scaled proportionally to the screen width compared to the reference display parameters. In this case, the effect of applying these formulas would be to keep the perceived depth in the same proportion to the viewing distance as in the reference setup.

When camera parameters are updated by a depth acquisition information SEI message in a following access unit and the baseline between the views used in the view synthesis process in the following access unit changes relative to that in the in the access unit which the reference displays information SEI belongs to, the baseline and the horizontal shift for the receiver's display in the following access unit should be modified accordingly. Let the scaling factor *s* be equal to the ratio of the baseline between two views in the following access unit and the baseline between the same two views in the access unit, which the reference displays information SEI message belongs to, where the two views are used in the view synthesis process. Then the baseline distance for the receiver's display in the following access unit should be modified with the scaling factor *s* relative to the baseline distance for the receiver's display in the access unit which the reference displays information SEI belongs to. The horizontal shift for the receiver's display should also be modified by scaling it with the same factor as that used to scale the baseline distance.

**prec\_ref\_baseline** specifies the exponent of the maximum allowable truncation error for ref\_baseline[ i ] as given by 2−prec\_ref\_baseline. The value of prec\_ref\_baselineshall be in the range of 0 to 31, inclusive.

**prec\_ref\_display\_width** specifies the exponent of the maximum allowable truncation error for ref\_display\_width[ i ] as given by 2−prec\_ref\_display\_width. The value of prec\_ref\_display\_width shall be in the range of 0 to 31, inclusive.

**ref\_viewing\_distance\_flag** equal to 1 indicates the presence of reference viewing distance. ref\_viewing\_distance\_flagequal to 0 indicates that the reference viewing distance is not present in the bitstream.

**prec\_ref\_viewing\_dist** specifies the exponent of the maximum allowable truncation error for ref\_viewing\_dist[ i ] as given by 2−prec\_ref\_viewing\_dist. The value of prec\_ref\_viewing\_distshall be in the range of 0 to 31, inclusive.

**num\_ref\_displays\_minus1** plus 1 specifies the number of reference displays that are signalled in the bitstream. The value of num\_ref\_displays\_minus1shall be in the range of 0 to 31, inclusive.

**exponent\_ref\_baseline[**i**]** specifies the exponent part of the reference baseline for the i-th reference display. The value of exponent\_ref\_baseline[ i ] shall be in the range of 0 to 62, inclusive. The value 63 is reserved for future use by ITU‑T | ISO/IEC. Decoders shall treat the value 63 as indicating an unspecified reference baseline.

**mantissa\_ref\_baseline[**i**]** specifies the mantissa part of the reference baseline for the i-th reference display. The length of the mantissa\_ref\_baseline[ i ] syntax element is variable and determined as follows.

– If exponent\_ref\_baseline[ i ] = = 0, the length is Max( 0, prec\_ref\_baseline− 30 ).

– Otherwise (0 < exponent\_ref\_baseline[ i ] < 63), the length is Max( 0, exponent\_ref\_baseline[ i ] + prec\_ref\_baseline− 31 ).

**exponent\_ref\_display\_width[**i**]** specifies the exponent part of the reference display width of the i-th reference display. The value of exponent\_ref\_display\_width[ i ] shall be in the range of 0 to 62, inclusive. The value 63 is reserved for future use by ITU‑T | ISO/IEC. Decoders shall treat the value 63 as indicating an unspecified reference display width.

**mantissa\_ref\_display\_width[**i**]** specifies the mantissa part of the reference display width of the i-th reference display. The length of the mantissa\_ref\_display\_width[ i ] syntax element is variable and determined as follows.

– If exponent\_ref\_display\_width[ i ] = = 0, the length is Max( 0, prec\_ref\_display\_width− 30 ).

– Otherwise (0 < exponent\_ref\_display\_width[ i ] < 63), the length is Max( 0, exponent\_ref\_display\_width[ i ] + prec\_ref\_display\_width− 31 ).

**exponent\_ref\_viewing\_distance[**i**]** specifies the exponent part of the reference viewing distance of the i-th reference display. The value of exponent\_ref\_viewing\_distance[ i ] shall be in the range of 0 to 62, inclusive. The value 63 is reserved for future use by ITU‑T | ISO/IEC. Decoders shall treat the value 63 as indicating an unspecified reference display width.

**mantissa\_ref\_viewing\_distance[**i**]** specifies the mantissa part of the reference viewing distance of the i-th reference display. The length of the mantissa\_ref\_viewing\_distance[ i ]syntax element is variable and determined as follows.

– If exponent\_ref\_viewing\_distance[ i ] = = 0, the length is Max( 0, prec\_ref\_viewing\_distance − 30 ).

– Otherwise ( 0 < exponent\_ref\_viewing\_distance[ i ] < 63 ), the length is Max( 0, exponent\_ref\_viewing\_distance[ i ] + prec\_ref\_viewing\_distance − 31 ).

The variables in the x column of Table I‑3 are derived as follows from the respective variables or values in the s, e, n, and v columns of Table I‑3 as follows.

– If 0 < e < 63, x = (−1)s \* 2e−31 \* (1 + n ÷2v).

– Otherwise (e is equal to 0), x = (−1)s \* 2−(30+v) \* n.

NOTE – The above specification is similar to that found in IEC 60559.

Table I‑3 – Association between camera parameter variables and syntax elements

|  |  |  |  |
| --- | --- | --- | --- |
| **x** | **s** | **e** | **n** |
| refBaseline[ i ] | 0 | exponent\_ref\_baseline[ i ] | mantissa\_ref\_baseline[ i ] |
| refDisplayWidth[ i ] | 0 | exponent\_ref\_display\_width[ i ] | mantissa\_ref\_display\_width[ i ] |
| refViewingDistance[ i ] | 0 | exponent\_ref\_viewing\_distance[ i ] | mantissa\_ref\_viewing\_distance[ i ] |

**additional\_shift\_present\_flag[**i**]** equal to 1 indicates that the information about additional horizontal shift of the left and right views for the i-th reference display is present in the bitstream. additional\_shift\_present\_flag[ i ] equal to 0 indicates that the information about additional horizontal shift of the left and right views for the i-th reference display is not present in the bitstream.

**num\_sample\_shift\_plus512[**i**]** indicates the recommended additional horizontal shift for a stereo-pair corresponding to the i-th reference baseline and the i-th reference display. If ( num\_sample\_shift\_plus512[ i ] − 512 ) is less than 0, it is recommended that the left view of the stereo-pair corresponding to the i-th reference baseline and the i-th reference display is shifted in the left direction by ( 512 − num\_sample\_shift\_plus512[ i ] ) samples with respect to the right view of the stereo-pair; if num\_sample\_shift\_plus512[ i ] is equal to 512, it is recommended that shifting is not applied; if ( num\_sample\_shift\_plus512[ i ] − 512 ) is greater than 0, it is recommended that the left view in the stereo-pair corresponding to the i-th reference baseline and the i-th reference display should be shifted in the right direction by ( 512 − num\_sample\_shift\_plus512[ i ] ) samples with respect to the right view of the stereo-pair. The value of num\_sample\_shift\_plus512[ i ] shall be in the range of 0 to 1023, inclusive.

**three\_dimensional\_reference\_displays\_extension\_flag** equal to 0 indicates that no additional data follows within the reference displays SEI message. The value of three\_dimensional\_reference\_displays\_extension\_flag shall be equal to 0. The value of 1 for three\_dimensional\_reference\_displays\_extension\_flag is reserved for future use by ITU-T | ISO/IEC. Decoders shall ignore all data that follows the value of 1 for three\_dimensional\_reference\_displays\_extension\_flag in a reference displays SEI message.

NOTE – Shifting the left view in the left (or right) direction by X samples with respect to the right view can be performed by the following two-step processing:

1. shift the left view by X/2 samples in the left (or right) direction, and shift the right view by X/2 samples in the right (or left) direction
2. fill the left and right image margins of X/2 samples in width in both the left and right views in background colour.

The following pseudo code explains the recommended shifting processing in the case of shifting the left view in the left direction by X samples with respect to the right view.

for ( i = X/2; i < width − X/2; i++ ) {  
 for  ( j=0; j < height; j++ ) {  
 left\_view[ j ][ i ] = left\_view[ j ][ i + X/2 ]   
 right\_view[ j ][ width − 1 − i ] = right\_view[ j ][ width − 1 − i − X/2 ]   
 }  
}  
for ( i = 0; i < X/2; i++) {  
 for ( j = 0; j < height; j++ ) {  
 left\_view[ j ][ width − 1 − i ] = left\_view[ j ][ i ] = Background\_Colour  
 right\_view[ j ][ width − 1 − i ] = right\_view[ j ][ i ] = Background\_Colour  
 }  
}

The following pseudo code explains the recommended shifting processing in the case of shifting the left view in the right direction by X samples with respect to the right view.

for ( i = X/2; i < width − X/2; i++ ) {  
 for ( j = 0; j < height; j++ ) {  
 left\_view[ j ][ width − 1 − i ] = left\_view[ j ][ width − 1 − i − X/2 ]   
 right\_view[ j ][ i ] = right\_view[ j ][ i + X/2 ]  
 }  
}  
for ( i=0; i < X/2; i++ ) {  
 for ( j = 0; j < height; j++ ) {  
 left\_view[ j ][ width − 1− i ] = left\_view[ j ][ i ] = Background\_Colour  
 right\_view[ j ][ width − 1− i ] = right\_view[ j ][ i ] = Background\_Colour  
 }  
}

Background\_Colourmay take different values in different systems, for example black or grey.

* + - 1. Depth timing SEI message semantics

The depth timing SEI message indicates the acquisition time of the depth view components of one or more access units relative to the DPB output time of the same access units. The depth timing SEI message may be present in any access unit and it pertains until the end of the coded video sequence or until the next depth timing SEI message, whichever is earlier in decoding order. The access units that the message pertains to are referred to as the target access unit set.

**per\_view\_depth\_timing\_flag** equal to 0 specifies that all the depth view components within the target access unit set have the same acquisition time offset relative to the DPB output time of the respective access unit in the target access unit set. The single occurrence of the depth\_timing\_offset structure specifies this acquisition time offset.

per\_view\_depth\_timing\_flag equal to 1 specifies that a depth\_timing\_offset syntax structure is present for each depth view in ascending order of view order index values for the depth views and specifies the acquisition time offset for that view.

* + - * 1. Depth timing offset semantics

**offset\_len\_minus1** specifies the length of the depth\_disp\_delay\_offset\_fp syntax element.

**depth\_disp\_delay\_offset\_fp** and **depth\_disp\_delay\_offset\_dp** specify that the acquisition offset of the respective depth view component or components relative to the DPB output time of the access unit containing the depth view component or components is equal to depth\_disp\_delay\_offset\_fp ÷ 2depth\_disp\_delay\_offset\_dp in units of clock ticks as specified in Annex C.

The length of depth\_disp\_delay\_offset\_fp syntax element is equal to offset\_len\_minus1 + 1.

If depth\_disp\_delay\_offset\_fp is not present, it is inferred to be equal to 0. If depth\_disp\_delay\_offset\_dp is not present, it is inferred to be equal to 0.

* + - 1. Depth sampling information SEI message semantics

The depth sampling information SEI message specifies the depth sample size relative to luma texture sample size. In addition, the depth sampling information SEI message specifies the depth sampling grid position of one or more depth view components of the associated access unit relative to the sampling grid of the texture view components of the same access unit with the same view\_id value. When present, the depth sampling information SEI message shall be associated with an IDR access unit. The semantics of the message are valid for the current coded video sequence.

NOTE – The depth sample size and the depth sampling grid position are indicated for frame or field view components that are present in the associated IDR access unit. In subsequent access units in the coded video sequence the depth view components may have different values of field\_pic\_flag and bottom\_field\_flag compared to those of the depth view components of the IDR access unit. Likewise, in subsequent access units in the coded video sequence the texture view components may have different values of field\_pic\_flag and bottom\_field\_flag compared to those of the texture view components of the IDR access unit. The depth sample size and depth sampling grid position should be modified according to the values of field\_pic\_flag and bottom\_field\_flag of the texture and depth view components of an access unit compared to those of the IDR access unit.

**dttsr\_x\_mul** and **dttsr\_x\_dp** indicate that the width of a depth sample relative to the width of a luma texture sample is approximately dttsr\_x\_mul ÷ 2dttsr\_x\_dp. When dttsr\_x\_mul is not present, it is inferred to be equal to 1. When dttsr\_x\_dp is not present, it is inferred to be equal to 0. The value of 0 for dttsr\_x\_mul is reserved.

**dttsr\_y\_mul** and **dttsr\_y\_dp** indicate that the height of a depth sample relative to the height of a luma texture sample is approximately dttsr\_y\_mul ÷ 2dttsr\_y\_dp. When dttsr\_y\_mul is not present, it is inferred to be equal to 1. When dttsr\_y\_dp is not present, it is inferred to be equal to 0. The value of 0 for dttsr\_y\_mul is reserved.

**per\_view\_depth\_grid\_pos\_flag** equal to 0 specifies that the depth sampling grid position information is the same for all depth views for which there is a texture view with the same view\_id present. The single occurrence of the depth\_grid\_position( ) syntax structure indicates the depth sampling grid position. per\_view\_depth\_grid\_pos\_flag equal to 1 specifies that a depth\_grid\_position( ) syntax structure is present for indicated depth views.

**num\_video\_plus\_depth\_views\_minus1** (when present) + 1 specifies the number of views for which the depth sampling grid position information is present in this SEI message.

**depth\_grid\_view\_id[** i **]** specifies the i-th view\_id value for which the depth sampling grid position information is specified with the depth\_grid\_position( ) structure following in the syntax structure.

* + - * 1. Depth grid position semantics

**depth\_grid\_pos\_x\_fp**, **depth\_grid\_pos\_x\_dp** and **depth\_grid\_pos\_x\_sign\_flag** indicate that the location of the horizontal position of the top-left sample in the sampling grid for a depth view component, relative to the location of the top-left sample in the sampling grid for the luma component of the texture view component with the same value of view\_id, is equal to ( 1 − 2 \* depth\_grid\_pos\_x\_sign\_flag ) \* ( depth\_grid\_pos\_x\_fp ÷ 2depth\_grid\_pos\_x\_dp ).

When depth\_grid\_pos\_x\_fp, depth\_grid\_pos\_x\_dp, and depth\_grid\_pos\_x\_sign\_flag are not present, they should be inferred to be equal to 0.

**depth\_grid\_pos\_y\_fp**, **depth\_grid\_pos\_y\_dp** and **depth\_grid\_pos\_y\_sign\_flag** indicate that the location of the vertical position of the top-left sample in the sampling grid for a depth view component, relative to the location of the top-left sample in the sampling grid for the luma component of the texture view component with the same value of view\_id, is equal to ( 1 − 2 \* depth\_grid\_pos\_y\_sign\_flag ) \* ( depth\_grid\_pos\_y\_fp ÷ 2depth\_grid\_pos\_y\_dp ).

When depth\_grid\_pos\_y\_fp, depth\_grid\_pos\_y\_dp, and depth\_grid\_pos\_y\_sign\_flag are not present, they should be inferred to be equal to 0.

* 1. Video usability information
     1. MVCD VUI parameters extension syntax

|  |  |  |
| --- | --- | --- |
| mvcd\_vui\_parameters\_extension( ) { | **C** | **Descriptor** |
| **vui\_mvcd\_num\_ops\_minus1** | 0 | ue(v) |
| for( i = 0; i <= vui\_mvcd\_num\_ops\_minus1; i++ ) { |  |  |
| **vui\_mvcd\_temporal\_id[** i **]** | 0 | u(3) |
| **vui\_mvcd\_num\_target\_output\_views\_minus1[** i **]** | 0 | ue(v) |
| for( j = 0; j <= vui\_mvcd\_num\_target\_output\_views\_minus1[ i ]; j++ ) { |  |  |
| **vui\_mvcd\_view\_id[** i **][** j **]** | 0 | ue(v) |
| **vui\_mvcd\_depth\_flag**[ i ][ j ] | 0 | u(1) |
| **vui\_mvcd\_texture\_flag**[ i ][ j ] | 0 | u(1) |
| } |  |  |
| **vui\_mvcd\_timing\_info\_present\_flag[** i **]** | 0 | u(1) |
| if( vui\_mvcd\_timing\_info\_present\_flag[ i ] ) { |  |  |
| **vui\_mvcd\_num\_units\_in\_tick[** i **]** | 0 | u(32) |
| **vui\_mvcd\_time\_scale[** i **]** | 0 | u(32) |
| **vui\_mvcd\_fixed\_frame\_rate\_flag[** i **]** | 0 | u(1) |
| } |  |  |
| **vui\_mvcd\_nal\_hrd\_parameters\_present\_flag[** i **]** | 0 | u(1) |
| if( vui\_mvcd\_nal\_hrd\_parameters\_present\_flag[ i ] ) |  |  |
| hrd\_parameters( ) | 0 |  |
| **vui\_mvcd\_vcl\_hrd\_parameters\_present\_flag[** i **]** | 0 | u(1) |
| if( vui\_mvcd\_vcl\_hrd\_parameters\_present\_flag[ i ] ) |  |  |
| hrd\_parameters( ) | 0 |  |
| if( vui\_mvcd\_nal\_hrd\_parameters\_present\_flag[ i ] | |   vui\_mvcd\_vcl\_hrd\_parameters\_present\_flag[ i ] ) |  |  |
| **vui\_mvcd\_low\_delay\_hrd\_flag[** i **]** | 0 | u(1) |
| **vui\_mvcd\_pic\_struct\_present\_flag[** i **]** | 0 | u(1) |
| } |  |  |
| } |  |  |

* + 1. MVCD VUI parameters extension semantics

The MVCD VUI parameters extension specifies VUI parameters that apply to one or more operation points for the coded video sequence. In Annex C it is specified which of the HRD parameter sets specified in the MVCD VUI parameters extension are used for conformance checking. All MVCD VUI parameters extensions that are referred to by a coded video sequence shall be identical.

Some texture and depth views identified by vui\_mvcd\_view\_id[ i ][ j ] may not be present in the coded video sequence. Some temporal subsets identified by vui\_mvcd\_temporal\_id[ i ] may not be present in the coded video sequence.

**vui\_mvcd\_num\_ops\_minus1** plus 1 specifies the number of operation points for which timing information, NAL HRD parameters, VCL HRD parameters, and the pic\_struct\_present\_flag may be present. The value of vui\_mvcd\_num\_ops\_minus1 shall be in the range of 0 to 1023, inclusive.

**vui\_mvcd\_temporal\_id[** i **]** indicates the maximum value of temporal\_id for all VCL NAL units in the representation of the i-th operation point.

**vui\_mvcd\_num\_target\_output\_views\_minus1[** i **]** plus one specifies the number of target output views for the i-th operation point. The value of vui\_mvcd\_num\_target\_output\_views\_minus1[ i ] shall be in the range of 0 to 1023, inclusive.

**vui\_mvcd\_view\_id[** i **][** j **]** indicates the j-th target output view in the i-th operation point. The value of vui\_mvcd\_view\_id[ i ] shall be in the range of 0 to 1023, inclusive.

**vui\_mvcd\_depth\_flag**[ i ][ j ] equal to 0 specifies that no depth view with view\_id equal to vui\_mvcd\_view\_id[ i ][ j ] is included in the j-th operation point. vui\_mvcd\_depth\_flag[ i ][ j ] equal to 1 specifies that the depth view with view\_id equal to vui\_mvcd\_view\_id[ i ][ j ] is included in the j-th operation point.

The value of vuimvcdOpDepthPresent[ i ] is derived as follows:

vuimvcdOpDepthPresent[ i ] = 0  
for( k = 0; k < vui\_mvcd\_num\_target\_output\_views\_minus1[ i ]; k++ )  
 vuimvcdOpDepthPresent[ i ] = vuimvcdOpDepthPresent[ i ] | vui\_mvcd\_depth\_flag[ i ][ k ]

**vui\_mvcd\_texture\_flag**[ i ][ j ] equal to 0 specifies that no texture view with view\_id equal to vui\_mvcd\_view\_id[ i ][ j ] is included in the j-th operation point. vui\_mvcd\_depth\_flag[ i ][ j ] equal to 1 specifies that the texture view with view\_id equal to vui\_mvcd\_view\_id[ i ][ j ] is included in the j-th operation point. When vui\_mvcd\_depth\_flag[ i ][ j ] is equal to 0, vui\_mvcd\_texture\_flag[ i ][ j ] shall be equal to 1.

The following syntax elements apply to the coded video sequence that is obtained by the sub-bitstream extraction process as specified in clause I.8.5.3 with tIdTarget equal to vui\_mvcd\_temporal\_id[ i ], viewIdTargetList containing vui\_mvcd\_view\_id[ i ][ j ] for all j in the range of 0 to vui\_mvcd\_num\_target\_output\_views\_minus1[ i ], inclusive, for which vui\_mvcd\_texture\_flag[ i ][ j ] is equal to 1, depthPresentFlagTarget equal to vuimvcdOpDepthPresent[ i ], and, if vuimvcdOpDepthPresent[ i ] is equal to 1, viewIdDepthTargetList containing vui\_mvcd\_view\_id[ i ][ j ] for all j in the range of 0 to vui\_mvcd\_num\_target\_output\_views\_minus1[ i ], inclusive, for which vui\_mvcd\_depth\_flag[ i ][ j ] is equal to 1 as the inputs and the i-th sub-bitstream as the output.

**vui\_mvcd\_timing\_info\_present\_flag[** i **]** equal to 1 specifies that vui\_mvcd\_num\_units\_in\_tick[ i ], vui\_mvcd\_time\_scale[ i ], and vui\_mvcd\_fixed\_frame\_rate\_flag[ i ] for the i-th sub-bitstream are present in the MVCD VUI parameters extension. vui\_mvcd\_timing\_info\_present\_flag[ i ] equal to 0 specifies that vui\_mvcd\_num\_units\_in\_tick[ i ], vui\_mvcd\_time\_scale[ i ], and vui\_mvcd\_fixed\_frame\_rate\_flag[ i ] for the i-th sub-bitstream are not present in the MVCD VUI parameters extension.

The following syntax elements for the i-th sub-bitstream are specified using references to Annex E. For these syntax elements the same semantics and constraints as the ones specified in Annex E apply, as if these syntax elements vui\_mvcd\_num\_units\_in\_tick[ i ], vui\_mvcd\_time\_scale[ i ], vui\_mvcd\_fixed\_frame\_rate\_flag[ i ], vui\_mvcd\_nal\_hrd\_parameters\_present\_flag[ i ], vui\_mvcd\_vcl\_hrd\_parameters\_present\_flag[ i ], vui\_mvcd\_low\_delay\_hrd\_flag[ i ], and vui\_mvcd\_pic\_struct\_present\_flag[ i ] were present as the syntax elements num\_units\_in\_tick, time\_scale, fixed\_frame\_rate\_flag, nal\_hrd\_parameters\_present\_flag, vcl\_hrd\_parameters\_present\_flag, low\_delay\_hrd\_flag, and pic\_struct\_present\_flag, respectively, in the VUI parameters of the active MVCD sequence parameter sets for the i-th sub-bitstream.

**vui\_mvcd\_num\_units\_in\_tick[** i **]** specifies the value of num\_units\_in\_tick, as specified in clause E.2.1, for the i-th sub-bitstream.

**vui\_mvcd\_time\_scale[** i **]** specifies the value of time\_scale, as specified in clause E.2.1, for the i-th sub-bitstream.

**vui\_mvcd\_fixed\_frame\_rate\_flag[** i **]** specifies the value of fixed\_frame\_rate\_flag, as specified in clause E.2.1, for the i-th sub-bitstream.

**vui\_mvcd\_nal\_hrd\_parameters\_present\_flag[** i **]** specifies the value of nal\_hrd\_parameters\_present\_flag, as specified in clause E.2.1, for the i-th sub-bitstream.

When vui\_mvcd\_nal\_hrd\_parameters\_present\_flag[ i ] is equal to 1, NAL HRD parameters (clauses E.1.2 and E.2.2) for the i-th sub-bitstream immediately follow the flag.

The variable VuiMvcNalHrdBpPresentFlag[ i ] is derived as follows:

– If any of the following is true, the value of VuiMvcNalHrdBpPresentFlag[ i ] shall be set equal to 1:

– vui\_mvcd\_nal\_hrd\_parameters\_present\_flag[ i ] is present in the bitstream and is equal to 1,

– for the i-th sub-bitstream, the need for presence of buffering periods for NAL HRD operation to be present in the bitstream in buffering period SEI messages is determined by the application, by some means not specified in this Recommendation | International Standard.

– Otherwise, the value of VuiMvcNalHrdBpPresentFlag[ i ] shall be set equal to 0.

**vui\_mvcd\_vcl\_hrd\_parameters\_present\_flag[** i **]** specifies the value of vcl\_hrd\_parameters\_present\_flag, as specified in clause E.2.1, for the i-th sub-bitstream.

When vui\_mvcd\_vcl\_hrd\_parameters\_present\_flag[ i ] is equal to 1, VCL HRD parameters (clauses E.1.2 and E.2.2) for the i-th sub-bitstream immediately follow the flag.

The variable VuiMvcVclHrdBpPresentFlag[ i ] is derived as follows:

– If any of the following is true, the value of VuiMvcVclHrdBpPresentFlag[ i ] shall be set equal to 1:

– vui\_mvcd\_vcl\_hrd\_parameters\_present\_flag[ i ] is present in the bitstream and is equal to 1,

– for the i-th sub-bitstream, the need for presence of buffering periods for VCL HRD operation to be present in the bitstream in buffering period SEI messages is determined by the application, by some means not specified in this Recommendation | International Standard.

– Otherwise, the value of VuiMvcVclHrdBpPresentFlag[ i ] shall be set equal to 0.

The variable VuiMvcCpbDpbDelaysPresentFlag[ i ] is derived as follows:

– If any of the following is true, the value of VuiMvcCpbDpbDelaysPresentFlag[ i ] shall be set equal to 1:

– vui\_mvcd\_nal\_hrd\_parameters\_present\_flag[ i ] is present in the bitstream and is equal to 1,

– vui\_mvcd\_vcl\_hrd\_parameters\_present\_flag[ i ] is present in the bitstream and is equal to 1,

– for the i-th sub-bitstream, the need for presence of CPB and DPB output delays to be present in the bitstream in picture timing SEI messages is determined by the application, by some means not specified in this Recommendation | International Standard.

– Otherwise, the value of VuiMvcCpbDpbDelaysPresentFlag[ i ] shall be set equal to 0.

**vui\_mvcd\_low\_delay\_hrd\_flag[** i **]** specifies the value of low\_delay\_hrd\_flag, as specified in clause E.2.1, for the i-th sub-bitstream.

**vui\_mvcd\_pic\_struct\_present\_flag[** i **]** specifies the value of pic\_struct\_present\_flag, as specified in clause E.2.1, for the i-th sub-bitstream.