**COMMITTEE DRAFT AMENDMENT****© ISO/IEC 2021 – All rights reserved****Text of ISO/IEC 14496-10:2020/CDAM 1** **63****Part 10: Advanced video coding, AMENDMENT 1: Additional SEI messages****Information technology — Coding of audio-visual objects****Élément introductif — Élément central — Partie 10: Titre de la partie****Information technology — Coding of audio-visual objects — Part 10: Advanced video coding, AMENDMENT 1: Additional SEI messages****E****2021-04-28****(30) Committee****ISO/IEC****ISO/IEC J****2020****1****Amendment****International Standard****202x****50****ISO/IEC 14496‑****ISO/IEC 14496‑10****ISO/IEC 14496-10:2020/CDAM 1****JISC****Coding of audio, picture, multimedia and hypermedia information****Information technology****5****29****1** **2****見出し 2****見出し 1****0****2****STD Version 2.1c2****30** **4** **ISO/IEC JTC 1/SC 29 /WG 5 N 50**

Date: **2021-04-28**

**Text of ISO/IEC 14496-10:2020/CDAM 1**

ISO/IEC JTC 1/SC 29/WG 5

Secretariat:  JISC

**Information technology — Coding of audio-visual objects — Part 10: Advanced video coding, AMENDMENT 1: Additional SEI messages**

*Élément introductif — Élément central — Partie 10: Titre de la partie*

|  |
| --- |
| **Warning**  This document is not an ISO International Standard. It is distributed for review and comment. It is subject to change without notice and may not be referred to as an International Standard.  Recipients of this draft are invited to submit, with their comments, notification of any relevant patent rights of which they are aware and to provide supporting documentation |

**Copyright notice**

This ISO document is a working draft or committee draft and is copyright-protected by ISO. While the reproduction of working drafts or committee drafts in any form for use by participants in the ISO standards development process is permitted without prior permission from ISO, neither this document nor any extract from it may be reproduced, stored or transmitted in any form for any other purpose without prior written permission from ISO.

Requests for permission to reproduce this document for the purpose of selling it should be addressed as shown below or to ISO's member body in the country of the requester:

[Indicate the full address, telephone number, fax number, telex number, and electronic mail address, as appropriate, of the Copyright Manger of the ISO member body responsible for the secretariat of the TC or SC within the framework of which the working document has been prepared.]

Reproduction for sales purposes may be subject to royalty payments or a licensing agreement.

Violators may be prosecuted.

Foreword

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

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

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see [www.iso.org/patents](https://www.iso.org/iso-standards-and-patents.html)) or the IEC list of patent declarations received (see [http://patents.iec.ch](http://patents.iec.ch/)).

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

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

This document was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 29, *Coding of audio, picture, multimedia and hypermedia information*, in collaboration with ITU-T. Technically aligned twin text is published as Rec. ITU-T H.264.

A list of all parts in the ISO/IEC 14496 series can be found on the ISO website.

Any feedback or questions on this document should be directed to the user’s national standards body. A complete listing of these bodies can be found at [www.iso.org/members.html](https://www.iso.org/members.html).

Information technology — Coding of audio-visual objects — Part 10: Advanced video coding, AMENDMENT 1: Additional SEI messages

*Clause 2*

Add an additional normative reference as follows:

Recommendation ITU-T H.274 (in force) | ISO/IEC 23002-7, *Versatile supplemental enhancement information messages for coded video bitstreams*

*7.4.1*

In subclause 7.4.1, replace the paragraphs of semantics of rbsp\_byte[ i ] with the following:

**rbsp\_byte**[ i ] is the i-th byte of an RBSP. An RBSP is specified as an ordered sequence of bytes as follows:

The RBSP contains an SODB as follows:

– If the SODB is empty (i.e., zero bits in length), the RBSP is also empty.

– Otherwise, the RBSP contains the SODB as follows:

1) The first byte of the RBSP contains the first (most significant, left-most) eight bits of the SODB; the next byte of the RBSP contains the next eight bits of the SODB, etc., until fewer than eight bits of the SODB remain.

2) The rbsp\_trailing\_bits( ) syntax structure is present after the SODB as follows:

i) The first (most significant, left-most) bits of the final RBSP byte contain the remaining bits of the SODB (if any).

ii) The next bit consists of a single bit equal to 1 (i.e., rbsp\_stop\_one\_bit).

iii) When the rbsp\_stop\_one\_bit is not the last bit of a byte-aligned byte, one or more zero-valued bits (i.e., instances of rbsp\_alignment\_zero\_bit) are present to result in byte alignment.

3) One or more cabac\_zero\_word 16-bit syntax elements equal to 0x0000 may be present in some RBSPs after the rbsp\_trailing\_bits( ) at the end of the RBSP.

*Annex D*

Replace the first paragraph of Annex D with the following:

This annex specifies 1) the syntax and semantics for the SEI payload, which is the container of SEI messages, 2) the syntax and semantics for some SEI messages, and 3) the use of the SEI messages for which the payloadType values are specified in this document and the syntax and semantics are specified in Rec. ITU-T H.274 | ISO/IEC 23002‑7.

When any SEI message specified in Rec. ITU-T H.274 | ISO/IEC 23002-7 is included in a non-VCL NAL unit as specified in this document, its syntax elements and semantics shall be as specified in Rec. ITU-T H.274 | ISO/IEC 23002-7.

*D.1.1*

Replace the contents of subclause D.1.1 with the following:

**D.1.1 General SEI message syntax**

|  |  |  |
| --- | --- | --- |
| sei\_payload( payloadType, payloadSize ) { | **C** | **Descriptor** |
| if( payloadType = = 0 ) |  |  |
| buffering\_period( payloadSize ) | 5 |  |
| else if( payloadType = = 1 ) |  |  |
| pic\_timing( payloadSize ) | 5 |  |
| else if( payloadType = = 2 ) |  |  |
| pan\_scan\_rect( payloadSize ) | 5 |  |
| else if( payloadType = = 3 ) |  |  |
| filler\_payload( payloadSize ) | 5 |  |
| else if( payloadType = = 4 ) |  |  |
| user\_data\_registered\_itu\_t\_t35( payloadSize ) | 5 |  |
| else if( payloadType = = 5 ) |  |  |
| user\_data\_unregistered( payloadSize ) | 5 |  |
| else if( payloadType = = 6 ) |  |  |
| recovery\_point( payloadSize ) | 5 |  |
| else if( payloadType = = 7 ) |  |  |
| dec\_ref\_pic\_marking\_repetition( payloadSize ) | 5 |  |
| else if( payloadType = = 8 ) |  |  |
| spare\_pic( payloadSize ) | 5 |  |
| else if( payloadType = = 9 ) |  |  |
| scene\_info( payloadSize ) | 5 |  |
| else if( payloadType = = 10 ) |  |  |
| sub\_seq\_info( payloadSize ) | 5 |  |
| else if( payloadType = = 11 ) |  |  |
| sub\_seq\_layer\_characteristics( payloadSize ) | 5 |  |
| else if( payloadType = = 12 ) |  |  |
| sub\_seq\_characteristics( payloadSize ) | 5 |  |
| else if( payloadType = = 13 ) |  |  |
| full\_frame\_freeze( payloadSize ) | 5 |  |
| else if( payloadType = = 14 ) |  |  |
| full\_frame\_freeze\_release( payloadSize ) | 5 |  |
| else if( payloadType = = 15 ) |  |  |
| full\_frame\_snapshot( payloadSize ) | 5 |  |
| else if( payloadType = = 16 ) |  |  |
| progressive\_refinement\_segment\_start( payloadSize ) | 5 |  |
| else if( payloadType = = 17 ) |  |  |
| progressive\_refinement\_segment\_end( payloadSize ) | 5 |  |
| else if( payloadType = = 18 ) |  |  |
| motion\_constrained\_slice\_group\_set( payloadSize ) | 5 |  |
| else if( payloadType = = 19 ) |  |  |
| film\_grain\_characteristics( payloadSize ) | 5 |  |
| else if( payloadType = = 20 ) |  |  |
| deblocking\_filter\_display\_preference( payloadSize ) | 5 |  |
| else if( payloadType = = 21 ) |  |  |
| stereo\_video\_info( payloadSize ) | 5 |  |
| else if( payloadType = = 22 ) |  |  |
| post\_filter\_hint( payloadSize ) | 5 |  |
| else if( payloadType = = 23 ) |  |  |
| tone\_mapping\_info( payloadSize ) | 5 |  |
| else if( payloadType = = 24 ) |  |  |
| scalability\_info( payloadSize ) /\* specified in Annex ‎G \*/ | 5 |  |
| else if( payloadType = = 25 ) |  |  |
| sub\_pic\_scalable\_layer( payloadSize ) /\* specified in Annex ‎G \*/ | 5 |  |
| else if( payloadType = = 26 ) |  |  |
| non\_required\_layer\_rep( payloadSize ) /\* specified in Annex ‎G \*/ | 5 |  |
| else if( payloadType = = 27 ) |  |  |
| priority\_layer\_info( payloadSize ) /\* specified in Annex ‎G \*/ | 5 |  |
| else if( payloadType = = 28 ) |  |  |
| layers\_not\_present( payloadSize ) /\* specified in Annex ‎G \*/ | 5 |  |
| else if( payloadType = = 29 ) |  |  |
| layer\_dependency\_change( payloadSize ) /\* specified in Annex ‎G \*/ | 5 |  |
| else if( payloadType = = 30 ) |  |  |
| scalable\_nesting( payloadSize ) /\* specified in Annex ‎G \*/ | 5 |  |
| else if( payloadType = = 31 ) |  |  |
| base\_layer\_temporal\_hrd( payloadSize ) /\* specified in Annex ‎G \*/ | 5 |  |
| else if( payloadType = = 32 ) |  |  |
| quality\_layer\_integrity\_check( payloadSize ) /\* specified in Annex ‎G \*/ | 5 |  |
| else if( payloadType = = 33 ) |  |  |
| redundant\_pic\_property( payloadSize ) /\* specified in Annex ‎G \*/ | 5 |  |
| else if( payloadType = = 34 ) |  |  |
| tl0\_dep\_rep\_index( payloadSize ) /\* specified in Annex ‎G \*/ | 5 |  |
| else if( payloadType = = 35 ) |  |  |
| tl\_switching\_point( payloadSize ) /\* specified in Annex ‎G \*/ | 5 |  |
| else if( payloadType = = 36 ) |  |  |
| parallel\_decoding\_info( payloadSize ) /\* specified in Annex ‎H \*/ | 5 |  |
| else if( payloadType = = 37 ) |  |  |
| mvc\_scalable\_nesting( payloadSize ) /\* specified in Annex ‎H \*/ | 5 |  |
| else if( payloadType = = 38 ) |  |  |
| view\_scalability\_info( payloadSize ) /\* specified in Annex ‎H \*/ | 5 |  |
| else if( payloadType = = 39 ) |  |  |
| multiview\_scene\_info( payloadSize ) /\* specified in Annex ‎H \*/ | 5 |  |
| else if( payloadType = = 40 ) |  |  |
| multiview\_acquisition\_info( payloadSize ) /\* specified in Annex ‎H \*/ | 5 |  |
| else if( payloadType = = 41 ) |  |  |
| non\_required\_view\_component( payloadSize ) /\* specified in Annex ‎H \*/ | 5 |  |
| else if( payloadType = = 42 ) |  |  |
| view\_dependency\_change( payloadSize ) /\* specified in Annex ‎H \*/ | 5 |  |
| else if( payloadType = = 43 ) |  |  |
| operation\_points\_not\_present( payloadSize ) /\* specified in Annex ‎H \*/ | 5 |  |
| else if( payloadType = = 44 ) |  |  |
| base\_view\_temporal\_hrd( payloadSize ) /\* specified in Annex ‎H \*/ | 5 |  |
| else if( payloadType = = 45 ) |  |  |
| frame\_packing\_arrangement( payloadSize ) | 5 |  |
| else if( payloadType = = 46 ) |  |  |
| multiview\_view\_position( payloadSize ) /\* specified in Annex ‎H \*/ | 5 |  |
| else if( payloadType = = 47 ) |  |  |
| display\_orientation( payloadSize ) | 5 |  |
| else if( payloadType = = 48 ) |  |  |
| mvcd\_scalable\_nesting( payloadSize ) /\* specified in Annex ‎I \*/ | 5 |  |
| else if( payloadType = = 49 ) |  |  |
| mvcd\_view\_scalability\_info( payloadSize ) /\* specified in Annex ‎I \*/ | 5 |  |
| else if( payloadType = = 50 ) |  |  |
| depth\_representation\_info( payloadSize ) /\* specified in Annex ‎I \*/ | 5 |  |
| else if( payloadType = = 51 ) |  |  |
| three\_dimensional\_reference\_displays\_info( payloadSize )  /\* specified in Annex ‎I \*/ | 5 |  |
| else if( payloadType = = 52 ) |  |  |
| depth\_timing( payloadSize ) /\* specified in Annex ‎I \*/ | 5 |  |
| else if( payloadType = = 53 ) |  |  |
| depth\_sampling\_info( payloadSize ) /\* specified in Annex ‎I \*/ | 5 |  |
| else if( payloadType = = 54 ) |  |  |
| constrained\_depth\_parameter\_set\_identifier( payloadSize )   /\* specified in Annex ‎J \*/ | 5 |  |
| else if( payloadType = = 56 ) |  |  |
| green\_metadata( payloadSize ) /\* specified in ISO/IEC 23001-11 \*/ | 5 |  |
| else if( payloadType = = 137 ) |  |  |
| mastering\_display\_colour\_volume( payloadSize ) | 5 |  |
| else if( payloadType = = 142 ) |  |  |
| colour\_remapping\_info( payloadSize ) | 5 |  |
| else if( payloadType = = 144 ) |  |  |
| content\_light\_level\_info( payloadSize ) | 5 |  |
| else if( payloadType = = 147 ) |  |  |
| alternative\_transfer\_characteristics( payloadSize ) | 5 |  |
| else if( payloadType = = 148 ) |  |  |
| ambient\_viewing\_environment( payloadSize ) | 5 |  |
| else if( payloadType = = 149 ) |  |  |
| content\_colour\_volume( payloadSize ) | 5 |  |
| else if( payloadType = = 150 ) |  |  |
| equirectangular\_projection( payloadSize ) | 5 |  |
| else if( payloadType = = 151 ) |  |  |
| cubemap\_projection( payloadSize ) | 5 |  |
| else if( payloadType = = 154 ) |  |  |
| sphere\_rotation( payloadSize ) | 5 |  |
| else if( payloadType = = 155 ) |  |  |
| regionwise\_packing( payloadSize ) | 5 |  |
| else if( payloadType = = 156 ) |  |  |
| omni\_viewport( payloadSize ) | 5 |  |
| else if( payloadType = = 181 ) |  |  |
| alternative\_depth\_info( payloadSize ) /\* specified in Annex ‎I \*/ | 5 |  |
| else if( payloadType = = 200 ) |  |  |
| sei\_manifest( payloadSize ) | 5 |  |
| else if( payloadType = = 201 ) |  |  |
| sei\_prefix\_indication( payloadSize ) | 5 |  |
| else if( payloadType = = 202)  /\* specified in Rec. ITU-T H.274 | ISO/IEC 23002-7 \*/ |  |  |
| annotated\_regions( payloadSize ) | 5 |  |
| else if( payloadType = = 205 ) |  |  |
| shutter\_interval\_info( payloadSize ) | 5 |  |
| else |  |  |
| reserved\_sei\_message( payloadSize ) | 5 |  |
| if( !byte\_aligned( ) ) { |  |  |
| **bit\_equal\_to\_one** /\* equal to 1 \*/ | 5 | f(1) |
| while( !byte\_aligned( ) ) |  |  |
| **bit\_equal\_to\_zero** /\* equal to 0 \*/ | 5 | f(1) |
| } |  |  |
| } |  |  |

*D.1.38*

Renumber subclause D.1.38 as D.1.39.

Add a new subclause D.1.38 as follows:

**D.1.38 Shutter interval information SEI message syntax**

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

*D.2.1*

At the beginning of subclause D.2.1, insert the following paragraph:

The variable PayloadBits, representing the number of bits in the SEI message syntax structure (e.g., the buffering\_period( ) syntax structure), is derived as follows: If byte\_aligned( ) is FALSE after the parsing of the SEI message syntax structure, PayloadBits is set equal to 8 \* payloadSize − nPayloadZeroBits − 1, where nPayloadZeroBits is the number of bit\_equal\_to\_zero syntax elements at the end of the sei\_payload( ) syntax structure; otherwise, PayloadBits is set equal to 8 \* payloadSize.

Add a new subclause D.2.1.1 as follows:

**D.2.1.1 Use of Rec. ITU-T H.274 | ISO/IEC 23002-7 SEI messages**

The SEI messages having syntax structures identified in subclause D.1.1 that are specified in Rec. ITU-T H.274 | ISO/IEC 23002-7 may be used together with bitstreams specified by this document.

When any particular Rec. ITU-T H.274 | ISO/IEC 23002-7 SEI message is included in a bitstream specified by this document, the SEI payload syntax shall be included into the sei\_payload( ) syntax structure as specified in subclause D.1.1, shall use the payloadType value specified in subclause D.1.1, and, additionally, any SEI-message-specific constraints specified in this annex for that particular SEI message shall apply.

The value of PayloadBits, as specified in aubclause D.2.1, is passed to the parser of the SEI message syntax structures specified in Rec. ITU-T H.274 | ISO/IEC 23002-7.

When an annotated regions SEI message is included in a bitstream specified by this document, for purposes of interpretation of the annotated regions SEI message, the following variables are specified:

– CroppedWidth is set equal to pic\_width\_in\_luma\_samples − SubWidthC \* ( pps\_conf\_win\_right\_offset + pps\_conf\_win\_left\_offset ).

– CroppedHeight is set equal to pic\_height\_in\_luma\_samples − SubHeightC \* ( pps\_conf\_win\_bottom\_offset + pps\_conf\_win\_top\_offset ).

– ConfWinLeftOffset is set equal to pps\_conf\_win\_left\_offset.

– ConfWinTopOffset is set equal to pps\_conf\_win\_top\_offset.

*D.2.21*

Replace the contents of D.2.21 with the following:

**D.2.21 Film grain characteristics SEI message semantics**

This SEI message provides the decoder with a parameterized model for film grain synthesis.

NOTE 1 For example, an encoder could use the film grain characteristics SEI message to characterize film grain that was present in the original source video material and was removed by pre-processing filtering techniques. Synthesis of simulated film grain on the decoded images for the display process is optional and does not need to exactly follow the specified semantics of the film grain characteristics SEI message. When synthesis of simulated film grain on the decoded images for the display process is performed, there is no requirement that the method by which the synthesis is performed be the same as the parameterized model for the film grain as provided in the film grain characteristics SEI message.

NOTE 2 The display process is not specified in this document.

NOTE 3 SMPTE RDD 5 specifies a film grain simulator based on the information provided in the film grain characteristics SEI message.

The film grain models specified in the film grain characteristics SEI message are expressed for application to decoded pictures that have 4:4:4 colour format with luma and chroma bit depths corresponding to the luma and chroma bit depths of the film grain model and use the same colour representation domain as the identified film grain model. When the colour format of the decoded video is not 4:4:4 or the decoded video uses a different luma or chroma bit depth from that of the film grain model or uses a different colour representation domain from that of the identified film grain model, an unspecified conversion process is expected to be applied to convert the decoded pictures to the form that is expressed for application of the film grain model.

NOTE 4 Because the use of a specific method is not required for performing the film grain generation function used by the display process, a decoder could, if desired, down-convert the model information for chroma in order to simulate film grain for other chroma formats (4:2:0 or 4:2:2) rather than up-converting the decoded video (using a method not specified in this document) before performing film grain generation.

**film\_grain\_characteristics\_cancel\_flag** equal to 1 indicates that the SEI message cancels the persistence of any previous film grain characteristics SEI message in output order. film\_grain\_characteristics\_cancel\_flag equal to 0 indicates that film grain modelling information follows.

**film\_grain\_model\_id** identifies the film grain simulation model as specified in Table D‑5. The value of film\_grain\_model\_id shall be in the range of 0 to 1, inclusive. The values of 2 and 3 for film\_grain\_model\_id are reserved for future use by ITU‑T | ISO/IEC and shall not be present in bitstreams conforming to this version of this document. Decoders shall ignore film grain characteristic SEI messages with film\_grain\_model\_id equal to 2 or 3.

**Table D‑5 – film\_grain\_model\_id values**

|  |  |
| --- | --- |
| **Value** | **Description** |
| 0 | frequency filtering |
| 1 | auto-regression |
| 2 | reserved |
| 3 | reserved |

**separate\_colour\_description\_present\_flag** equal to 1 indicates that a distinct combination of luma bit depth, chroma bit depth, video full range flag, colour primaries, transfer characteristics, and matrix coefficients for the film grain characteristics specified in the SEI message is present in the film grain characteristics SEI message syntax. separate\_colour\_description\_present\_flag equal to 0 indicates that the combination of luma bit depth, chroma bit depth, video full range flag, colour primaries, transfer characteristics, and matrix coefficients for the film grain characteristics specified in the SEI message are the same as indicated in VUI parameters for the coded video sequence.

NOTE 5 When separate\_colour\_description\_present\_flag is equal to 1, any of the luma bit depth, chroma bit depth, video full range flag, colour primaries, transfer characteristics, and matrix coefficients specified for the film grain characteristics specified in the SEI message could differ from that for the pictures in the coded video sequence.

When VUI parameters are not present for the coded video sequence or the value of colour\_description\_present\_flag is equal to 0, and equivalent information to that conveyed when colour\_description\_present\_flag is equal to 1 is not conveyed by external means, separate\_colour\_description\_present\_flag shall be equal to 1.

The decoded image Idecoded used in the equations in this subclause is in the same colour representation domain as the simulated film grain signal. Therefore, when any of these parameters does differ from that for the pictures in the coded video sequence, the decoded image Idecoded used in the equations in this subclause would be in a different colour representation domain than that for the pictures in the coded video sequence. For example, when the value of film\_grain\_bit\_depth\_luma\_minus8 + 8 is greater than the bit depth of the luma component of the pictures in the coded video sequence, the bit depth of Idecoded used in the equations in this subclause is also greater than the bit depth of the luma component of the pictures in the coded video sequence. In such a case, the decoded image Idecoded corresponding to an actual decoded picture would be generated by converting the actual decoded picture to be in the same colour representation domain as the the simulated film grain signal. The process for converting an actual decoded pictures to the 4:4:4 colour format with same colour representation domain as the the simulated film grain signal is not specified in this document.

**film\_grain\_bit\_depth\_luma\_minus8** plus 8 specifies the bit depth used for the luma component of the film grain characteristics specified in the SEI message. When film\_grain\_bit\_depth\_luma\_minus8 is not present in the film grain characteristics SEI message, the value of film\_grain\_bit\_depth\_luma\_minus8 is inferred to be equal to bit\_depth\_luma\_minus8.

The value of filmGrainBitDepth[ 0 ] is derived as follows:

filmGrainBitDepth[ 0 ] = film\_grain\_bit\_depth\_luma\_minus8 + 8 (D-14)

**film\_grain\_bit\_depth\_chroma\_minus8** plus 8 specifies the bit depth used for the Cb and Cr components of the film grain characteristics specified in the SEI message. When film\_grain\_bit\_depth\_chroma\_minus8 is not present in the film grain characteristics SEI message, the value of film\_grain\_bit\_depth\_chroma\_minus8 is inferred to be equal to bit\_depth\_chroma\_minus8.

The value of filmGrainBitDepth[ c ] for c = 1 and 2 is derived as follows:

filmGrainBitDepth[ c ] = film\_grain\_bit\_depth\_chroma\_minus8 + 8    with c = 1, 2 (D-15)

**film\_grain\_full\_range\_flag** has the same semantics as specified in clause ‎E.2.1 for the video\_full\_range\_flag syntax element, except as follows:

– film\_grain\_full\_range\_flag specifies the video full range flag of the film grain characteristics specified in the SEI message, rather than the video full range flag used for the coded video sequence.

– When film\_grain\_full\_range\_flag is not present in the film grain characteristics SEI message, the value of film\_grain\_full\_range\_flag is inferred to be equal to video\_full\_range\_flag.

**film\_grain\_colour\_primaries** has the same semantics as specified in clause ‎E.2.1 for the colour\_primaries syntax element, except as follows:

– film\_grain\_colour\_primaries specifies the colour primaries of the film grain characteristics specified in the SEI message, rather than the colour primaries used for the coded video sequence.

– When film\_grain\_colour\_primaries is not present in the film grain characteristics SEI message, the value of film\_grain\_colour\_primaries is inferred to be equal to colour\_primaries.

**film\_grain\_transfer\_characteristics** has the same semantics as specified in clause ‎E.2.1 for the transfer\_characteristics syntax element, except as follows:

– film\_grain\_transfer\_characteristics specifies the transfer characteristics of the film grain characteristics specified in the SEI message, rather than the transfer characteristics used for the coded video sequence.

– When film\_grain\_transfer\_characteristics is not present in the film grain characteristics SEI message, the value of film\_grain\_transfer\_characteristics is inferred to be equal to transfer\_characteristics.

**film\_grain\_matrix\_coefficients** has the same semantics as specified in clause ‎E.2.1 for the matrix\_coefficients syntax element, except as follows:

– film\_grain\_matrix\_coefficients specifies the matrix coefficients of the film grain characteristics specified in the SEI message, rather than the matrix coefficients used for the coded video sequence.

– When film\_grain\_matrix\_coefficients is not present in the film grain characteristics SEI message, the value of film\_grain\_matrix\_coefficients is inferred to be equal to matrix\_coefficients.

– The values allowed for film\_grain\_matrix\_coefficients are not constrained by the chroma format of the decoded video pictures that is indicated by the value of chroma\_format\_idc for the semantics of the VUI parameters.

**blending\_mode\_id** identifies the blending mode used to blend the simulated film grain with the decoded images as specified in Table D‑6. blending\_mode\_id shall be in the range of 0 to 1, inclusive. The values of 2 and 3 for blending\_mode\_id are reserved for future use by ITU‑T | ISO/IEC and shall not be present in bitstreams conforming to this version of this document. Decoders shall ignore film grain characteristic SEI messages with blending\_mode\_id equal to 2 or 3.

**Table D‑6 – blending\_mode\_id values**

|  |  |
| --- | --- |
| **Value** | **Description** |
| 0 | additive |
| 1 | multiplicative |
| 2 | reserved |
| 3 | reserved |

Depending on the value of blending\_mode\_id, the blending mode is specified as follows:

– If blending\_mode\_id is equal to 0 the blending mode is additive as specified by

Igrain[ c ][ x ][ y ] = Clip3( 0, ( 1 << filmGrainBitDepth[ c ] ) − 1, Idecoded[ c ][ x ][ y ] + G[ c ][ x ][ y ] ) (D-16)

– Otherwise (blending\_mode\_id is equal to 1), the blending mode is multiplicative as specified by

Igrain[ c ][ x ][ y ] = Clip3( 0, ( 1 << filmGrainBitDepth[ c ] ) − 1, Idecoded[ c ][ x ][ y ] (D-17)  
 + Round( ( Idecoded[ c ][ x ][ y ] \* G[ c ][ x ][ y ] ) ÷ ( ( 1 << bitDepth[ c ] ) − 1 ) ) )

where Idecoded[ c ][ x ][ y ] represents the sample value at coordinates x,  y of the colour component c of the decoded image Idecoded, G[ c ][ x ][ y ] is the simulated film grain value at the same position and colour component, and filmGrainBitDepth[ c ] is the number of bits used for each sample in a fixed-length unsigned binary representation of the arrays Igrain[ c ][ x ][ y ], Idecoded[ c ][ x ][ y ], and G[ c ][ x ][ y ], where c = 0..2, x = 0..PicWidthInSamplesL − 1, and y = 0..PicHeightInSamplesL – 1.

**log2\_scale\_factor** specifies a scale factor used in the film grain characterization equations.

**comp\_model\_present\_flag[** c ]equal to 0 indicates that film grain is not modelled on the c-th colour component, where c equal to 0 refers to the luma component, c equal to 1 refers to the Cb component, and c equal to 2 refers to the Cr component. comp\_model\_present\_flag[ c ] equal to 1 indicates that syntax elements specifying modelling of film grain on colour component c are present in the SEI message.

When separate\_colour\_description\_present\_flag is equal to 0 and chroma\_format\_idc is equal to 0, the value of comp\_model\_present\_flag[ 1 ] and comp\_model\_present\_flag[ 2 ] shall be equal to 0.

**num\_intensity\_intervals\_minus1**[ c ] plus 1 specifies the number of intensity intervals for which a specific set of model values has been estimated.

NOTE 6 – The intensity intervals could overlap in order to simulate multi-generational film grain.

**num\_model\_values\_minus1**[ c ] plus 1 specifies the number of model values present for each intensity interval in which the film grain has been modelled. The value of num\_model\_values\_minus1[ c ] shall be in the range of 0 to 5, inclusive.

**intensity\_interval\_lower\_bound**[ c ][ i ] specifies the lower bound of the i-th of intensity interval for which the set of model values applies.

**intensity\_interval\_upper\_bound**[ c ][ i ] specifies the upper bound of the i-th of intensity interval for which the set of model values applies.

The variable intensityIntervalIdx[ c ][ x ][ y ][ j ] represents the j-th index to the list of intensity intervals selected for the sample value Idecoded[ c ][ x ][ y ] for c = 0..2, x = 0..PicWidthInSamplesL − 1, y = 0..PicHeightInSamplesL − 1, and j = 0..numApplicableIntensityIntervals[ c ][ x ][ y ] − 1, where numApplicableIntensityIntervals[ c ][ x ][ y ] is derived below.

Depending on film\_grain\_model\_id, the selection of one or more intensity intervals for the sample value Idecoded[ c ][ x ][ y ] is specified as follows:

– The variable numApplicableIntensityIntervals[ c ][ x ][ y ] is initially set equal to 0.

If film\_grain\_model\_id is equal to 0, the following applies:

* The top-left sample location ( xB, yB ) of the current 8x8 block b that contains the sample value Idecoded[ c ][ x ][ y ] is derived as ( xB, yB ) = ( x / 8, y / 8 ).
* The average value bavg of the current 8x8 block b is derived as follows:

sum8x8 = 0  
for( i = 0; i  < 8; i++ )  
 for( j = 0; j < 8; j++ )  
      sum8x8 += Idecoded[ xB \* 8 + I, yB \* 8 + j, c ] (D-18)  
bavg = Clip3( 0, 255, ( sum8x8 + ( 1  <<  ( filmGrainBitDepth[ c ] − 3 ) ) )  >>  ( filmGrainBitDepth[ c ] − 2 ) )

* The value of intensityIntervalIdx[ c ][ x ][ y ][ j ] is derived as follows:

for( i = 0, j = 0; i  <=  num\_intensity\_intervals\_minus1[ c ]; i++ )  
 if( bavg  >=  intensity\_interval\_lower\_bound[ c ][ i ] &&  
      bavg  <=  intensity\_interval\_upper\_bound[ c ][ i ] ) {  
   intensityIntervalIdx[ c ][ x ][ y ][ j ] = i (D-19)  
   j++  
 }  
numApplicableIntensityIntervals[ c ][ x ][ y ] = j

– Otherwise (film\_grain\_model\_id is equal to 1), the value of intensityIntervalIdx[ c ][ x ][ y ][ j ] is derived as follows:

I8[ c ][ x ][ y ] = ( filmGrainBitDepth[ c ]  = =  8 ) ? ( Idecoded[ c ][ x ][ y ] :  
 Clip3( 0, 255, ( Idecoded[ c ][ x ][ y ] + ( 1  <<  ( filmGrainBitDepth[ c ] − 9 ) ) )  >>  ( filmGrainBitDepth[ c ] − 8 ) )  
for( i = 0, j = 0; i  <=  num\_intensity\_intervals\_minus1[ c ]; i++ )  
 if( I8[ c ][ x ][ y ]  >=  intensity\_interval\_lower\_bound[ c ][ i ] &&  
      I8[ c ][ x ][ y ]  <=  intensity\_interval\_upper\_bound[ c ][ i ] ) { (D-20)  
   intensityIntervalIdx[ c ][ x ][ y ][ j ] = i  
   j++  
 }  
numApplicableIntensityIntervals[ c ][ x ][ y ] = j

Samples that do not fall into any of the defined intervals (i.e., those samples for which the value of numApplicableIntensityIntervals[ c ][ x ][ y ] is equal to 0) are not modified by the grain generation function. Samples that fall into more than one interval (i.e., those samples for which the value of numApplicableIntensityIntervals[ c ][ x ][ y ] is greater than 1) will originate multi-generation grain. Multi-generation grain results from adding the grain computed independently for each of the applicable intensity intervals.

In the equations in the remainder of this clause, the variable sj in each instance of the list comp\_model\_value[ c ][ sj ] is the value of intensityIntervalIdx[ c ][ x ][ y ][ j ] derived for the sample value Idecoded[ c ][ x ][ y ].

**comp\_model\_value**[ c ][ i ][ j ] specifies the j-th model value for the colour component c and the i-th intensity interval. The set of model values has different meaning depending on the value of film\_grain\_model\_id.

The value of comp\_model\_value[ c ][ i ][ j ] is constrained as follows, and could be additionally constrained as specified elsewhere in this clause.

– If film\_grain\_model\_id is equal to 0, comp\_model\_value[ c ][ i ][ j ] shall be in the range of 0 to 2filmGrainBitDepth[ c ] − 1, inclusive.

– Otherwise (film\_grain\_model\_id is equal to 1), comp\_model\_value[ c ][ i ][ j ] shall be in the range of −2( filmGrainBitDepth[ c ] − 1 ) to 2( filmGrainBitDepth[ c ] − 1 ) − 1, inclusive.

Depending on the value of film\_grain\_model\_id, the synthesis of the film grain is modelled as follows:

– If film\_grain\_model\_id is equal to 0, a frequency filtering model enables simulating the original film grain for c = 0..2, x = 0..PicWidthInSamplesL − 1, and y = 0..PicHeightInSamplesL − 1, as specified by:

G[ c ][ x ][ y ] = ( comp\_model\_value[ c ][ sj ][ 0 ] \* Q[ c ][ x ][ y ] + comp\_model\_value[ c ][ sj ][ 5 ] \*  
 G[ c − 1 ][ x ][ y ] ) >> log2\_scale\_factor (D-21)

where Q[ c ] is a two-dimensional random process generated by filtering 16x16 blocks gaussRv with random-value elements gaussRvij generated with a normalized Gaussian distribution (independent and identically distributed Gaussian random variable samples with zero mean and unity variance) and where the value of an element G[ x, y, c −1 ] used in the right-hand side of the equation is inferred to be equal to 0 when c − 1 is less than 0.

NOTE 7 A normalized Gaussian random variable can be generated from two independent, uniformly distributed random values over the interval from 0 to 1 (and not equal to 0), denoted as uRv0 and uRv1, using the Box-Muller transformation specified by

gaussRvi,j = Sqrt( −2 \* Ln( uRv0 ) ) \* Cos( 2 \* π \* uRv1 ) (D-22)

where π is Archimedes' constant 3.141 592 653 589 793....

The band-pass filtering of blocks gaussRv can be performed in the discrete cosine transform (DCT) domain as follows:

for( y = 0; y < 16; y++ )  
 for( x = 0; x < 16; x++ )  
 if( ( x < comp\_model\_value[ c ][ sj ][ 3 ]  &&  y < comp\_model\_value[ c ][ sj ][ 4 ] )  | | (D-23)  
 x > comp\_model\_value[ c ][ sj ][ 1 ]  | |  y > comp\_model\_value[ c ][ sj ][ 2 ] )  
 gaussRv[ x ][ y ] = 0  
filteredRv = IDCT16x16( gaussRv )

where IDCT16x16( z ) refers to a unitary inverse discrete cosine transformation (IDCT) operating on a 16x16 matrix argument z as specified by

IDCT16x16( z ) = r \* z \* rT (D-24)

where the superscript T indicates a matrix transposition and r is the 16x16 matrix with elements rij specified by

(D-25)

where π is Archimedes' constant 3.141 592 653 589 793....

Q[ c ] is formed by the frequency-filtered blocks filteredRv.

NOTE 8 Coded model values are based on blocks of size 16x16, but a decoder implementation could use other block sizes. For example, decoders implementing the IDCT on 8x8 blocks could down-convert by a factor of two the set of coded model values comp\_model\_value[ c ][ sj ][ i ] for i equal to 1..4.

NOTE 9 To reduce the degree of visible blocks that result from mosaicking the frequency-filtered blocks filteredRv, decoders could apply a low-pass filter to the boundaries between frequency-filtered blocks.

– Otherwise (film\_grain\_model\_id is equal to 1), an auto-regression model enables simulating the original film grain for c = 0..2, x = 0..PicWidthInSamplesL − 1, and y = 0..PicHeightInSamplesL − 1 as specified by

G[ c ][ x ][ y ] = ( comp\_model\_value[ c ][ s ][ 0 ] \* n[ c ][ x ][ y ] +  
comp\_model\_value[ c ][ sj ][ 1 ] \* ( G[ c ][ x − 1 ][ y ] + ( ( comp\_model\_value[ c ][ sj ][ 4 ] \* G[ c ][ x ][ y − 1 ] ) >>  
 log2\_scale\_factor ) ) +  
comp\_model\_value[ c ][ sj ][ 3 ] \* ( ( ( comp\_model\_value[ c ][ sj ][ 4 ] \* G[ c ][ x − 1 ][ y − 1 ] ) >>  
 log2\_scale\_factor ) + G[ c ][ x + 1 ][ y − 1 ] ) +  
comp\_model\_value[ c ][ sj ][ 5 ] \* ( G[ c ][ x − 2 ][ y ] +  
 ( ( comp\_model\_value[ c ][ sj ][ 4 ] \* comp\_model\_value[ c ][ sj ][ 4 ] \* G[ c ][ x ][ y − 2 ] ) >>  
 ( 2 \* log2\_scale\_factor ) ) ) +  
 comp\_model\_value[ c ][ sj ][ 2 ] \* G[ c − 1 ][ x ][ y ] ) >> log2\_scale\_factor (D-26)

where n[ c ][ x ][ y ] is a random value with normalized Gaussian distribution (independent and identically distributed Gaussian random variable samples with zero mean and unity variance for each value of x, y, and c) and the value of an element G[ c ][ x ][ y ] used in the right-hand side of the equation is inferred to be equal to 0 when any of the following conditions are true:

– x is less than 0,

– y is less than 0,

– c is less than 0.

comp\_model\_value[ c ][ i ][ 0 ] provides the first model value for the model as specified by film\_grain\_model\_id. comp\_model\_value[ c ][ i ][ 0 ] corresponds to the standard deviation of the Gaussian noise term in the generation functions specified in Equations D-21 through D-26.

comp\_model\_value[ c ][ i ][ 1 ] provides the second model value for the model as specified by film\_grain\_model\_id. When film\_grain\_model\_id is equal to 0, comp\_model\_value[ c ][ i ][ 1 ] shall be greater than or equal to 0 and less than 16.

When not present in the film grain characteristics SEI message, comp\_model\_value[ c ][ i ][ 1 ] is inferred as follows:

– If film\_grain\_model\_id is equal to 0, comp\_model\_value[ c ][ i ][ 1 ] shall be inferred to be equal to  8.

– Otherwise (film\_grain\_model\_id is equal to 1), comp\_model\_value[ c ][ i ][ 1 ] is inferred to be equal to  0.

comp\_model\_value[ c ][ i ][ 1 ] is interpreted as follows:

– If film\_grain\_model\_id is equal to 0, comp\_model\_value[ c ][ i ][ 1 ] indicates the horizontal high cut frequency to be used to filter the DCT of a block of 16x16 random values.

– Otherwise (film\_grain\_model\_id is equal to 1), comp\_model\_value[ c ][ i ][ 1 ] indicates the first order spatial correlation for neighbouring samples (x − 1, y) and (x, y − 1).

comp\_model\_value[ c ][ i ][ 2 ] provides the third model value for the model as specified by film\_grain\_model\_id. When film\_grain\_model\_id is equal to 0, comp\_model\_value[ c ][ i ][ 2 ] shall be greater than or equal to 0 and less than 16.

When not present in the film grain characteristics SEI message, comp\_model\_value[ c ][ i ][ 2 ] shall be inferred as follows:

– If film\_grain\_model\_id is equal to 0, comp\_model\_value[ c ][ i ][ 2 ] is inferred to be equal to comp\_model\_value[ c ][ i ][ 1 ]

– Otherwise (film\_grain\_model\_id is equal to 1), comp\_model\_value[ c ][ i ][ 2 ] is inferred to be equal to 0.

comp\_model\_value[ c ][ i ][ 2 ] is interpreted as follows:

– If film\_grain\_model\_id is equal to 0, comp\_model\_value[ c ][ i ][ 2 ] indicates the vertical high cut frequency to be used to filter the DCT of a block of 16x16 random values.

– Otherwise (film\_grain\_model\_id is equal to 1), comp\_model\_value[ c ][ i ][ 2 ] indicates the colour correlation between consecutive colour components.

comp\_model\_value**[**c ][ i ][ 3 ] provides the fourth model value for the model as specified by film\_grain\_model\_id. When film\_grain\_model\_id is equal to 0, comp\_model\_value[ c ][ i ][ 3 ] shall be greater than or equal to 0 and less than or equal to comp\_model\_value[ c ][ i ][ 1 ].

When not present in the film grain characteristics SEI message, comp\_model\_value[ c ][ i ][ 3 ] is inferred to be equal to 0.

comp\_model\_value[ c ][ i ][ 3 ] is interpreted as follows:

– If film\_grain\_model\_id is equal to 0, comp\_model\_value[ c ][ i ][ 3 ] indicates the horizontal low cut frequency to be used to filter the DCT of a block of 16x16 random values.

– Otherwise (film\_grain\_model\_id is equal to 1), comp\_model\_value[ c ][ i ][ 3 ] indicates the first order spatial correlation for neighbouring samples (x − 1, y − 1) and (x + 1, y − 1).

comp\_model\_value[ c ][ i ][ 4 ] provides the fifth model value for the model as specified by film\_grain\_model\_id. When film\_grain\_model\_id is equal to 0, comp\_model\_value[ c ][ i ][ 4] shall be greater than or equal to 0 and less than or equal to comp\_model\_value[ c ][ i ][ 2 ].

When not present in the film grain characteristics SEI message, comp\_model\_value[ c ][ i ][ 4 ] is inferred to be equal to film\_grain\_model\_id.

comp\_model\_value[ c ][ i ][ 4 ] is interpreted as follows:

– If film\_grain\_model\_id is equal to 0, comp\_model\_value[ c ][ i ][ 4 ] indicates the vertical low cut frequency to be used to filter the DCT of a block of 16x16 random values.

– Otherwise (film\_grain\_model\_id is equal to 1), comp\_model\_value[ c ][ i ][ 4 ] indicates the aspect ratio of the modelled grain.

comp\_model\_value[ c ][ i ][ 5 ] provides the sixth model value for the model as specified by film\_grain\_model\_id.

When not present in the film grain characteristics SEI message, comp\_model\_value[ c ][ i ][ 5 ] is inferred to be equal to 0.

comp\_model\_value[ c ][ i ][ 5 ] is interpreted as follows:

– If film\_grain\_model\_id is equal to 0, comp\_model\_value[ c ][ i ][ 5 ] indicates the colour correlation between consecutive colour components.

– Otherwise (film\_grain\_model\_id is equal to 1), comp\_model\_value[ c ][ i ][ 5 ] indicates the second order spatial correlation for neighbouring samples (x, y − 2) and (x − 2, y).

**film\_grain\_characteristics\_repetition\_period** specifies the persistence of the film grain characteristics SEI message and may specify a picture order count interval within which another film grain characteristics SEI message or the end of the coded video sequence shall be present in the bitstream. The value of film\_grain\_characteristics\_repetition\_period shall be in the range 0 to 16 384, inclusive.

film\_grain\_characteristics\_repetition\_period equal to 0 specifies that the film grain characteristics SEI message applies to the current decoded picture only.

film\_grain\_characteristics\_repetition\_period equal to 1 specifies that the film grain characteristics SEI message persists in output order until any of the following conditions are true:

– A new coded video sequence begins.

– A picture in an access unit containing a film grain characteristics SEI message is output having PicOrderCnt( ) greater than PicOrderCnt( CurrPic ).

film\_grain\_characteristics\_repetition\_period greater than 1 specifies that the film grain characteristics SEI message persists until any of the following conditions are true:

– A new coded video sequence begins.

– A picture in an access unit containing a film grain characteristics SEI message is output having PicOrderCnt( ) greater than PicOrderCnt( CurrPic ) and less than or equal to PicOrderCnt( CurrPic ) + film\_grain\_characteristics\_repetition\_period.

film\_grain\_characteristics\_repetition\_period greater than 1 indicates that another film grain characteristics SEI message shall be present for a picture in an access unit that is output having PicOrderCnt( ) greater than PicOrderCnt( CurrPic ) and less than or equal to PicOrderCnt( CurrPic ) + film\_grain\_characteristics\_repetition\_period; unless the bitstream ends or a new coded video sequence begins without output of such a picture.

*D.2.38*

Renumber subclause D.2.38 as D.2.39.

Add a new subclause D.2.38 as follows:

**D.2.38 Shutter interval information SEI message semantics**

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

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

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

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

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

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

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

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

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

The variable subLayerShutterInterval[ i ], corresponding to the indicated shutter interval of each picture in the CVS for which the value of sii\_sub\_layer\_idx is equal to i, is thus derived as follows:

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

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

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

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

*E.2.1*

In subclause E.2.1, replace the paragraph of semantics of nal\_hrd\_parameters\_present\_flag and NOTE 12 with the following:

**nal\_hrd\_parameters\_present\_flag** equal to 1 specifies that NAL HRD parameters (pertaining to the Type II bitstream conformance point) are present. nal\_hrd\_parameters\_present\_flag equal to 0 specifies that NAL HRD parameters are not present.

NOTE 12 When nal\_hrd\_parameters\_present\_flag is equal to 0, the conformance of the bitstream cannot be verified without provision of the NAL HRD parameters and all buffering period SEI messages, and, when vcl\_hrd\_parameters\_present\_flag is also equal to 0, all picture timing SEI messages, by some means not specified in this Recommendation | International Standard.

Replace the paragraph of semantics of vcl\_hrd\_parameters\_present\_flag and NOTE 13 with the following:

**vcl\_hrd\_parameters\_present\_flag** equal to 1 specifies that VCL HRD parameters (pertaining to the Type I bitstream conformance point) are present. vcl\_hrd\_parameters\_present\_flag equal to 0 specifies that VCL HRD parameters are not present.

NOTE 13 When vcl\_hrd\_parameters\_present\_flag is equal to 0, the conformance of the bitstream cannot be verified without provision of the VCL HRD parameters and all buffering period SEI messages, and, when nal\_hrd\_parameters\_present\_flag is also equal to 0, all picture timing SEI messages, by some means not specified in this Recommendation | International Standard.

In Table E-4, replace the row for the value 13 with the following:

|  |  |  |
| --- | --- | --- |
| 13 | – If matrix\_coeffs is equal to 0           V = *α* \* Lc( 1 ÷ 2.4 ) − ( *α* − 1 ) for 1 >= Lc >= *β*           V = 12.92 \* Lc for *β* > Lc >= 0  – Otherwise           V = *α* \* Lc( 1 ÷ 2.4 ) − ( *α* − 1 ) for Lc >= *β*           V = 12.92 \* Lc for *β* > Lc > −*β*           V = − *α* \* ( −Lc )( 1 ÷ 2.4 ) + ( *α* − 1 ) for −*β* >= Lc | IEC 61966-2-1 sRGB (with matrix\_coeffs equal to 0)  IEC 61966-2-1 sYCC (with matrix\_coeffs equal to 5) |

After formula E-3, replace the next paragraph and associated bullet points with:

In this case, the range of E′R, E′G, and E′B is specified as follows:

– If transfer\_characteristics is equal to 11 or 12, or transfer\_characteristics is equal to 13 and matrix\_coeffs is not equal to 0, E′R, E′G, and E′B are real numbers with values that have a larger range than the range of 0 to 1, inclusive, and their range is not specified in this document.

– Otherwise, E′R, E′G and E′B are real numbers in the range of 0 to 1.

In Table E-5, move “IEC 61966-2-1 sYCC” from the row for the value 1 to the row for the value 5.

*F.9.2*

In subclause F.9.2, replace the four occurences of the phrase “156, 200, or 201” with “156, 200, 201, 202, or 205”.

*G.9.2*

In subclause G.9.2, replace the three occurences of the phrase “156, 200, or 201” with “156, 200, 201, 202, or 205”.

*H.9.2*

In subclause H.9.2, replace the three occurences of the phrase “156, 200, or 201” with “156, 200, 201, 202, or 205”.

*H.9.2.3.1*

In subclause H.9.2.3.1, replace “da\_mantissa\_len\_minus + 1” with “da\_mantissa\_len\_minus1 + 1”.

*H.9.2.4*

In subclause H.9.2.4, replace “should be shifted in the right direction by ( 512 − num\_sample\_shift\_plus512[ i ] ) samples” with “should be shifted in the right direction by ( num\_sample\_shift\_plus512[ i ] − 512 ) samples”.

*I.3.3.2.13.1*

Replace the contents of I.3.3.2.13.1 with the following:

**I.3.3.2.13.1 Depth ranges syntax**

|  |  |  |
| --- | --- | --- |
| depth\_ranges( numViews, predDirection, index ) { | C | **Descriptor** |
| **z\_near\_flag** | 11 | u(1) |
| **z\_far\_flag** | 11 | u(1) |
| if( z\_near\_flag ) |  |  |
| 3dv\_acquisition\_element( numViews, predDirection, 7, index, ZNearSign,  ZNearExp, ZNearMantissa, ZNearManLen ) |  |  |
| if( z\_far\_flag ) |  |  |
| 3dv\_acquisition\_element( numViews, predDirection, 7, index, ZFarSign,  ZFarExp, ZFarMantissa, ZFarManLen ) |  |  |
| } |  |  |

*I.3.3.2.13.2*

Replace the contents of I.3.3.2.13.2 with the following:

**I.3.3.2.13.2 3DV acquisition element syntax**

|  |  |  |
| --- | --- | --- |
| 3dv\_acquisition\_element( numViews, predDirection, expLen, index, outSign, outExp, outMantissa, outManLen ) { | **C** | **Descriptor** |
| if( numViews > 1 ) |  |  |
| **element\_equal\_flag** | 11 | u(1) |
| if( element\_equal\_flag  = =  0 ) |  |  |
| numValues = numViews |  |  |
| else |  |  |
| numValues = 1 |  |  |
| for( i = 0; i < numValues; i++ ) { |  |  |
| if( predDirection  = =  2  &&  i  = =  0 ) { |  |  |
| **mantissa\_len\_minus1** | 11 | u(5) |
| outManLen[ index, i ] = manLen = mantissa\_len\_minus1 + 1 |  |  |
| } |  |  |
| if( predDirection  = =  2 ) { |  |  |
| **sign0** | 11 | u(1) |
| outSign[ index, i ] = sign0 |  |  |
| **exponent0** | 11 | u(v) |
| outExp[ index, i ] = exponent0 |  |  |
| **mantissa0** | 11 | u(v) |
| outMantissa[ index, i ] = mantissa0 |  |  |
| } else { |  |  |
| **skip\_flag** | 11 | u(1) |
| if( skip\_flag = = 0 ) { |  |  |
| **sign1** | 11 | u(1) |
| outSign[ index, i ] = sign1 |  |  |
| **exponent\_skip\_flag** | 11 | u(1) |
| if( exponent\_skip\_flag = = 0 ) { |  |  |
| **exponent1** | 11 | u(v) |
| outExp[ index, i ] = exponent1 |  |  |
| } else |  |  |
| outExp[ index, i ] = outExp[ ref\_dps\_id0, i ] |  |  |
| **mantissa\_diff** | 11 | se(v) |
| if( predDirection = = 0 ) |  |  |
| mantissaPred = (( outMantissa[ ref\_dps\_id0, i ] \* predWeight0 +  outMantissa[ ref\_dps\_id1, i ] \* ( 64-predWeight0 ) + 32 ) >> 6 ) |  |  |
| else |  |  |
| mantissaPred = outMantissa[ ref\_dps\_id0, i ] |  |  |
| outMantissa[ index, i ] = mantissaPred + mantissa\_diff |  |  |
| outManLen[ index, i ] = outManLen[ ref\_dps\_id0, i ] |  |  |
| } else { |  |  |
| outSign[ index, i ] = outSign[ ref\_dps\_id0, i ] |  |  |
| outExp[ index, i ] = outExp[ ref\_dps\_id0, i ] |  |  |
| outMantissa[ index, i ] = outMantissa[ ref\_dps\_id0, i ] |  |  |
| outManLen[ index, i ] = outManLen[ ref\_dps\_id0, i ] |  |  |
| } |  |  |
| } |  |  |
| } |  |  |
| if( element\_equal\_flag = = 1 ) { |  |  |
| for( i = 1; i < numViews; i++ ) { |  |  |
| outSign[ index, i ] = outSign[ index, 0 ] |  |  |
| outExp[ index, i ] = outExp[ index, 0 ] |  |  |
| outMantissa[ index, i ] = outMantissa[ index, 0 ] |  |  |
| outManLen[ index, i ] = outManLen[ index, 0 ] |  |  |
| } |  |  |
| } |  |  |
| } |  |  |

*I.3.4.2.13.2*

Replace the contents of I.3.4.2.13.2 with the following:

**I.3.4.2.13.2 3DV acquisition element semantics**

The syntax structure specifies the value of an element in the depth ranges syntax structure. The element may contain one or more loop entries i of the order specified by view\_id\_3dv syntax elements.

The contents of the syntax structure are controlled through input variables predDirection, expLen, and index, the semantics of which are as follows.

– predDirection equal to 2 specifies that the first loop entry of the element is not predicted and coded in the sign, exponent, and mantissa syntax elements. predDirection equal to 0 or 1 specifies that the first loop entry of the element is predicted and a difference relative to a prediction value is coded in the difference syntax element.

– expLen specifies the number of bits in the exponent syntax element.

– index greater than 0 specifices the depth\_parameter\_set\_id of the depth parameter set wherein the parameters are present, and index equal to 0 specifies that the parameters are present in a sequence parameter set.

The syntax structure uses outSign, outExp, outMantissa, and outManLen variables for both input and output, where each variable is indexed by [ index, viewIdc ], index being an identifier (equal to either 0 when decoding depth ranges in sequence parameter set or depth\_parameter\_set\_id value when decoding depth range parameter set) to a depth parameter set and viewIdc being a view indicator (in the order of views for 3DV acquisition parameters).

**element\_equal\_flag** equal to 0 specifies that the sign, exponent, and mantissa may or may not be identical to respective values for any two loop entries i and j. element\_equal\_flag equal to 1 specifies that the sign, exponent, and mantissa are identical to respective values for any two loop entries i and j. When not present, element\_equal\_flag is inferred to be equal to 0.

**mantissa\_len\_minus1** plus 1 specifies the number of bits in the mantissa syntax element. The value of mantissa\_len\_minus1 shall be in the range of 0 to 31, inclusive.

**sign0** equal to 0 indicates that the sign of the value provided in the loop entry is positive. sign0 equal to 1 indicates that the sign is negative.

**exponent0** specifies the exponent of the value provided by the loop entry. The syntax element exponent0 is represented by expLen bits. The value of exponent0 shall be in the range of 0 to 2expLen – 2, inclusive. The value 2expLen – 1 is reserved for future use by ITU‑T | ISO/IEC. Decoders shall treat the value 2expLen – 1 as indicating an unspecified value.

**mantissa0** specifies the mantissa of the value provided by the loop entry. The syntax element mantissa0 is represented by manLen bits.

**skip\_flag** equal to 0 specifies that syntax elements sign1, exponent\_skip\_flag and mantissa\_diff are present for the loop entry. skip\_flag equal to 1 specifies that elements sign1, exponent\_skip\_flag and mantissa\_diff are not present for the loop entry.

**sign1** equal to 0 indicates that the sign of the value provided in the loop entry is positive. sign1 equal to 1 indicates that the sign is negative.

**exponent1**, if present, specifies the exponent of the value provided by the loop entry. The syntax element exponent1 is represented by expLen bits. The value of exponent1 shall be in the range of 0 to 2expLen – 2, inclusive. The value 2expLen – 1 is reserved for future use by ITU‑T | ISO/IEC. Decoders shall treat the value 2expLen – 1 as indicating an unspecified value.

**mantissa\_diff** specifies the difference of the mantissa of the value provided by the loop entry relative to its prediction value.

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