|  |
| --- |
| **INTERNATIONAL ORGANIZATION FOR STANDARDIZATION ORGANISATION INTERNATIONALE DE NORMALISATION ISO/IEC JTC 1/SC 29/WG 5 MPEG JOINT VIDEO EXPERTS TEAM WITH ITU-T SG 16** |
| **ISO/IEC JTC 1 / SC 29 / WG 5 N 221** |
| **Geneva, CH, 11–19 July 2023** |
| |  |  | | --- | --- | | **Title:** | **Technologies under consideration for future extensions of VSEI (draft 1)** | | **Source:** | **Convenor (Jens-Rainer Ohm)** | | **Type:** | **General** | | **Subtype:** | **Other** | | **Status:** | **Approved** | | **Date:** | **2023-09-15** | | **Expected Action:** | **Info** | | **Action due date:** | **N/A** | | **No. of pages** | **18** (without this cover page) | | **Email of convenor:** | **ohm @ ient . rwth-aachen . de** | | **Committee URL:** | **https://sd.iso.org/documents/ui/#!/browse/iso/iso-iec-jtc-1/iso-iec-jtc-1-sc-29/iso-iec-jtc-1-sc-29-wg-5** | |

|  |  |
| --- | --- |
| **Joint Video Experts Team (JVET)**  **of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29**  31st Meeting, Geneva, CH, 11–19 July 2023 | Document: JVET-AE2032-v1 |

|  |  |  |  |
| --- | --- | --- | --- |
| *Title:* | **Technologies under consideration for future extensions of VSEI (draft 1)** | | |
| *Status:* | Output document approved by JVET | | |
| *Purpose:* | Draft text | | |
| *Author(s) or Contact(s):* | Miska M. Hannuksela Jie Chen Sachin Deshpande Sean McCarthy | Email: | miska.hannuksela@nokia.com jiechen.cj@alibaba-inc.com sdeshpande@sharplabs.com sean.mccarthy@dolby.com |
| *Source:* | Editors | | |

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

**Abstract**

This document contains the draft text for changes under consideration for future extensions to the versatile supplemental enhancement information messages for coded video bitstreams (VSEI) standard (Rec. ITU-T H.274 | ISO/IEC 23002-7) to modify existing SEI messages or specify additional SEI messages.

This version of the document contains draft text to modify the neural-network post-filter SEI messages and draft text to specify the encoder optimization SEI message, source picture timing SEI message, object mask information SEI message, and neural-network post-filter group SEI message..

**Changes yet to be integrated:**

None

**Changes that have been integrated:**

[JVET-AE0061](https://jvet-experts.org/doc_end_user/current_document.php?id=13009) [AHG9] On NNPFC Application Purpose [S. Deshpande (Sharp)]  
[JVET-AE0064](https://jvet-experts.org/doc_end_user/current_document.php?id=13012) AHG8/AHG9: Signalling encoder preprocessing and human / machine viewing indications [C. Kim, D. Gwak, Hendry, J. Lim, S. Kim (LGE), M. M. Hannuksela, F. Cricri, H. Zhang (Nokia)]  
[JVET-AE0079](https://jvet-experts.org/doc_end_user/current_document.php?id=13027) AHG8/AHG9: Source picture timing information SEI message [S. McCarthy, G. J. Sullivan, P. Yin (Dolby)]  
[JVET-AE0095](https://jvet-experts.org/doc_end_user/current_document.php?id=13043) AHG8/AHG9: proposed changes to the candidate new object mask information SEI message [P. de Lagrange, E. François, D. Doyen (InterDigital), J. Chen, S. Wang, Y. Ye (Alibaba)]  
[JVET-AE0298](https://jvet-experts.org/doc_end_user/current_document.php?id=13262) AHG9: Combined text of JVET-AE0052 and JVET-AE0063 [M. M. Hannuksela, F. Cricri (Nokia), L. Chen, O. Chubach, Y.-W. Huang, S. Lei (MediaTek)]

**Changes to the specification text:**

*Changes to subclause 8.28*

Modify subclause 8.28 per [JVET-AE0061](https://jvet-experts.org/doc_end_user/current_document.php?id=13009) and [JVET-AE0298](https://jvet-experts.org/doc_end_user/current_document.php?id=13262) as follows

**8.28 Neural-network post-filter SEI messages**

**8.28.1 General** **post-processing filtering process using NNPFs**

**8.28.1.1 General**

Input to this process is a bitstream BitstreamToFilter. Output of this process is a list of NNPF output pictures ListNnpfOutputPics.

First, BitstreamToFilter is decoded, and the list CroppedDecodedPictures is set to be the list of the cropped decoded pictures in output order resulted from decoding BitstreamToFilter.

Second, NnpfCand is set to contain any single NNPF or any single NNPF group. When NnpfCand contains an NNPF group with nnpfgc\_grouping\_type equal to 3, the subseequnt specifications of this subclause apply when NnpfCand is set to contain individually each member NNPF, if any, and each member NNPF group, if any, of the NNPF group with nnpfgc\_grouping\_type equal to 3.

Third, the filtering process for one picture, as specified in subclause 8.28.1.2, is repeatedly invoked, in output order, for each cropped decoded picture that is in CroppedDecodedPictures and for which the single NNPF contained in NnpfCand or the single NNPF group contained in NnpfCand, or one or more NNPFs or NNPF groups defined as alternatives or alternating in the NNPF group contained in NnpfCand are activated.

The order of the pictures in ListNnpfOutputPics is in output order.

Within ListNnpfOutputPics there shall be no more than one picture pertaining to any particular output time instance. When for any particular picture in CroppedDecodedPictures there are multiple NNPFs activated and only one of the NNPFs is allowed to be chosen to be applied although any of the NNPFs may be chosen, the above constraint shall apply regardless of which NNPF is chosen to be applied to the particular picture.

BitstreamToFilter may be processed multiple times to generate multiple different ListNnpfOutputPics through the second and third steps above.

**8.28.1.2 Filtering process for one picture ~~using an NNPF~~**

The filtering process specified in this subclause applies to each cropped decoded picture, referred to as the current picture, that is in CroppedDecodedPictures and for which one or more NNPFs or NNPF groups in NnpfCand are activated.

An NNPF or an NNPF group to be applied to the current picture is selected as follows:

– If NnpfCand contains a single NNPF and that NNPF is activated for the current picture according to an NNPFA SEI message, that NNPF is selected to be applied to the current picture.

– Otherwise, if NnpfCand contains an NNPF group with nnpfgc\_grouping\_type equal to 2 and any NNPF of the NNPF group is activated for the current picture according to NNPFA SEI message, that NNPF is selected to be applied to the current picture.

– Otherwise, if NnpfCand contains an NNPF group with nnpfgc\_grouping\_type equal to 0 and that NNPF group is activated for the current picture according to an NNPFGA SEI message, that NNPF group is selected to be applied to the current picture.

– Otherwise (NnpfCand contains an NNPF group with nnpfgc\_grouping\_type equal to 1), the following applies:

– A set of candidate NNPFs or NNPF groups candSet is initially empty and then set to contain the following:

– The NNPFs that are activated for the current picture according to NNPFA SEI messages and are included in the NNPF group contained in NnpfCand.

– The NNPF groups that are activated for the current picture according to NNPFGA SEI messages and are included in the NNPF group contained in NnpfCand.

– For each candidate NNPF or NNPF group candFilter in candSet, the following applies:

– When one or more of the input pictures of candFilter are input pictures to the NNPF or NNPF group prevFilter that was used in any previous invocation of the filtering process specified in this subclause for the same NnpfCand, candFilter is excluded from candSet.

– Any NNPF or NNPF group remaining in candSet is selected to be applied to the current picture.

When applying an NNPF to the current picture, the following applies:

– The filtered and/or interpolated pictures are generated by the NNPF by applying the NNPF process specified in the semantics of the NNPFC SEI message, in a patch-wise manner, to the current picture.

– ~~When applying an NNPF to the current picture, t~~The order of the pictures generated by the NNPF by applying the NNPF process being stored into the output tensor of the NNPF is in output order.

– ~~When the applied NNPF is the last NNPF that is applied to the current picture, t~~The pictures generated by the NNPF and output by the NNPF process are included into ListNnpfOutputPics, in the same order as when the pictures are stored into the output tensor of the NNPF.

When applying an NNPF group to the current picture, the following applies:

– The filtered and/or interpolated pictures are generated by applying the NNPF process specified in the semantics of the NNPFC SEI message, in a patch-wise manner, as specified in the semantics of the NNPFGA SEI message activating the NNPF group.

– The pictures in NnpfgaOutputPicList are included into ListNnpfOutputPics, in the same order as the pictures are stored in NnpfgaOutputPicList.

**8.28.2 Neural-network post-filter characteristics SEI message**

**8.28.2.1 Neural-network post-filter characteristics SEI message syntax**

|  |  |
| --- | --- |
| nn\_post\_filter\_characteristics( payloadSize ) { | **Descriptor** |
| … | u(1) |
| if( nnpfc\_complexity\_info\_present\_flag ) { |  |
| **nnpfc\_parameter\_type\_idc** | u(2) |
| if( nnpfc\_parameter\_type\_idc != 2 ) |  |
| **nnpfc\_log2\_parameter\_bit\_length\_minus3** | u(2) |
| **nnpfc\_num\_parameters\_idc** | u(6) |
| **nnpfc\_num\_kmac\_operations\_idc** | ue(v) |
| **nnpfc\_total\_kilobyte\_size** | ue(v) |
| } |  |
| **nnpfc\_metadata\_extension\_num\_bits** | ue(v) |
| if( nnpfc\_metadata\_extension\_num\_bits > 0 ) { |  |
| if( nnpfc\_purpose = = 0 ) { |  |
| **nnpfc\_application\_purpose\_tag\_uri\_present\_flag** | u(1) |
| if( nnpfc\_application\_purpose\_tag\_uri\_present\_flag) |  |
| **nnpfc\_application\_purpose\_tag\_uri** | st(v) |
| } |  |
| **nnpfc\_reserved\_metadata\_extension** /\*Remaining bits of the metadata extension\*/ | u(v) |
| } |  |
| } |  |
| … |  |
| } |  |

**8.28.2.2 Neural-network post-filter characteristics SEI message semantics**

…

**nnpfc\_purpose** indicates the purpose of the NNPF as specified in Table 20, where ( nnpfc\_purpose & bitMask ) not equal to 0 indicates that the NNPF has the purpose associated with the bitMask value in Table 20. When nnpfc\_purpose is greater than 0 and ( nnpfc\_purpose & bitMask ) is equal to 0, the purpose associated with the bitMask value is not applicable to the NNPF. When nnpfc\_pupose is equal to 0, the NNPF may be used as determined by the application and as specified by the nnpfc\_application\_purpose\_tag\_uri.

The value of nnpfc\_purpose shall be in the range of 0 to 63, inclusive, in bitstreams conforming to this edition of this document. Values of 64 to 65 535, inclusive, for nnpfc\_purpose are reserved for future use by ITU-T | ISO/IEC and shall not be present in bitstreams conforming to this edition of this document. Decoders conforming to this edition of this document shall ignore NNPFC SEI messages with nnpfc\_purpose in the range of 64 to 65 535, inclusive.

**Table 20 – Definition of nnpfc\_purpose**

|  |  |
| --- | --- |
| **bitMask** | **Interpretation** |
| 0x01 | General visual quality improvement |
| 0x02 | Chroma upsampling (from the 4:2:0 chroma format to the 4:2:2 or 4:4:4 chroma format, or from the 4:2:2 chroma format to the 4:4:4 chroma format) |
| 0x04 | Resolution resampling (increasing or decreasing the width or height) |
| 0x08 | Picture rate upsampling |
| 0x10 | Bit depth upsampling (increasing the luma bit depth or the chroma bit depth) |
| 0x20 | Colourization |

The variables chromaUpsamplingFlag, resolutionResamplingFlag, pictureRateUpsamplingFlag, bitDepthUpsamplingFlag, and colourizationFlag, specifying whether nnpfc\_purpose indicates the purpose of the NNPF to include chroma upsampling, resolution resampling, picture rate upsampling, bit depth upsampling, and colourization, respectively, are derived as follows:

chromaUpsamplingFlag = ( ( nnpfc\_purpose & 0x02 ) > 0 ) ? 1 : 0  
resolutionResamplingFlag = ( ( nnpfc\_purpose & 0x04 ) > 0 ) ? 1 : 0  
pictureRateUpsamplingFlag = ( ( nnpfc\_purpose & 0x08 ) > 0 ) ? 1 : 0 (76)  
bitDepthUpsamplingFlag = ( ( nnpfc\_purpose & 0x10 ) > 0 ) ? 1 : 0  
colourizationFlag = ( ( nnpfc\_purpose & 0x20 ) > 0 ) ? 1 : 0

NOTE 2– When a reserved value of nnpfc\_purpose is taken into use in the future by ITU-T | ISO/IEC, the syntax of this SEI message could be extended with syntax elements whose presence is conditioned by nnpfc\_purpose being equal to that value.

When ChromaFormatIdc is equal to 3, chromaUpsamplingFlag shall be equal to 0.

When ChromaFormatIdc or chromaUpsamplingFlag is not equal to 0, colourizationFlag shall be equal to 0.

When pictureRateUpsamplingFlag is equal to 1 and the input picture with index 0 is associated with a frame packing arrangement SEI message with fp\_arrangement\_type equal to 5, all input pictures are associated with a frame packing arrangement SEI message with fp\_arrangement\_type equal to 5 and the same value of fp\_current\_frame\_is\_frame0\_flag.

…

**nnpfc\_application\_purpose\_tag\_uri\_present\_flag** equal to 1indicates that the nnpfc\_application\_purpose\_tag\_uri syntax element is present in this NNPFC SEI message. nnpfc\_application\_purpose\_tag\_uri\_present\_flag equal to 0indicates that the nnpfc\_application\_purpose\_tag\_uri syntax element is not present in this NNPFC SEI message. When not present nnpfc\_application\_purpose\_tag\_uri\_present\_flag is inferred to be equal to 0.

**nnpfc\_application\_purpose\_tag\_uri** specifies a tag URI with syntax and semantics as specified in IETF RFC 4151 identifying the application determined purpose of the NNPF, when nnpfc\_purpose is equal to 0.

NOTE 4 – nnpfc\_application\_purpose\_tag\_uri enables uniquely identifying the application determined purpose of NNPF without needing a central registration authority.

…

**nnpfc\_metadata\_extension\_num\_bits** equal to 0 specifies that nnpfc\_reserved\_metadata\_extension is not present. nnpfc\_metadata\_extension\_num\_bits greater than 0 specifies the length, in bits, of nnpfc\_reserved\_metadata\_extension. nnpfc\_metadata\_extension\_num\_bits shall be ~~equal to 0~~ in the range of 0 to 2048 in this edition of this document when nnpfc\_purpose is not equal to 0 and in the range of 1 to 2048 when nnpfc\_purpose is equal to 0. Values in the range of ~~1 to 2 048~~ 2049 to 4096, inclusive, for nnpfc\_metadata\_extension\_num\_bits are reserved for future use by ITU-T | ISO/IEC and shall not be present in bitstreams conforming to this edition of this document. Decoders conforming to this edition of this document shall allow any value of nnpfc\_metadata\_extension\_num\_bits in the range of 0 to ~~2 048~~ 4096, inclusive. Values of nnpfc\_metadata\_extension\_num\_bits greater than ~~2 048~~ 4096 shall not be present in bitstreams conforming to this edition of this document and are not reserved for future use.

**nnpfc\_reserved\_metadata\_extension** shall not be present in bitstreams conforming to this edition of this document. However, decoders conforming to this edition of this document shall ignore the presence and value of nnpfc\_reserved\_metadata\_extension. When present, and when nnpfc\_purpose is equal to 0 and nnpfc\_application\_purpose\_tag\_uri\_present\_flag is equal to 1 the length, in bits, of nnpfc\_reserved\_metadata\_extension is equal to nnpfc\_metadata\_extension\_num\_bits – Length of (nnpfc\_application\_purpose\_tag\_uri) – 1. When present and when nnpfc\_purpose is equal to 0 and nnpfc\_application\_purpose\_tag\_uri\_present\_flag is equal to 0 the length, in bits, of nnpfc\_reserved\_metadata\_extension is equal to nnpfc\_metadata\_extension\_num – 1 bits. When present and when nnpfc\_purpose is not equal to 0 the length, in bits, of nnpfc\_reserved\_metadata\_extension is equal to nnpfc\_metadata\_extension\_num\_bits.

**8.28.3 Neural-network post-filter activation SEI message**

**8.28.3.1 Neural-network post-filter activation SEI message syntax**

*No change*

**8.28.3.1 Neural-network post-filter activation SEI message syntax**

[Ed. (MH): The NNPF activation SEI message may be extended to include activation of an NNPFG. Below are the related text changes as highlighted.]

The neural-network post-filter activation (NNPFA) SEI message activates or de-activates the possible use of the target neural-network post-processing filter (NNPF) or the target neural-network post-processing filter group (NNPFG), identified by nnpfa\_target\_id and nnpfa\_target\_base\_flag, for post-processing filtering of a set of pictures.

For a particular picture for which the NNPF is activated, the target NNPF is derived as follows:

– If nnpfa\_target\_base\_flag is equal to 1, the target NNPF is the base NNPF with nnpfc\_id equal to nnpfa\_target\_id.

– Otherwise (nnpfa\_target\_base\_flag is equal to 0), the target NNPF is the NNPF specified by the last NNPFC SEI message with nnpfc\_id equal to nnpfa\_target\_id that precedes the first VCL NAL unit of the current picture in decoding order and is not a repetition of the NNPFC SEI message that contains the base NNPF.

NOTE 1 – There can be several NNPFA SEI messages present for the same picture, for example, when the NNPFs are meant for different purposes or for filtering of different colour components.

For a particular picture for which the NNPFG is activated, the target NNPFG is the NNPFG specified by the NNPFG SEI message with nnpfg\_id equal to nnpfa\_target\_id that precedes the first VCL NAL unit of the current picture in decoding order.

**nnpfa\_target\_id** indicates the target NNPF or NNPFG, which is specified by one or more NNPFC SEI messages or an NNPFG SEI message that pertain to the current picture and have nnpfc\_id or nnpfg\_id equal to nnpfa\_target\_id. The value of nnpfa\_target\_id shall be in the range of 0 to 232 − 2, inclusive.

*New subclauses*

Add subclauses 8.28.4 and 8.28.5 as follows:

* + 1. **Neural-network post-filter group characteristics SEI message**
       1. **Neural-network post-filter group characteristics SEI message syntax**

|  |  |
| --- | --- |
| nn\_post\_filter\_group\_characteristics( payloadSize ) { | **Descriptor** |
| **nnpfgc\_id** | ue(v) |
| **nnpfgc\_grouping\_type** | ue(v) |
| if( nnpfgc\_grouping\_type = = 0 | | nnpfgc\_grouping\_type = = 2 ) |  |
| **nnpfgc\_purpose** | u(16) |
| **nnpfgc\_num\_members\_minus2** | ue(v) |
| for( i = 0; i <= nnpfgc\_num\_members\_minus2 + 1; i++ ) |  |
| **nnpfgc\_member\_id**[ i ] | ue(v) |
| **nnpfgc\_complexity\_info\_present\_flag** | u(1) |
| if( nnpfgc\_complexity\_info\_present\_flag ) { |  |
| **nnpfgc\_parameter\_type\_idc** | u(2) |
| if( nnpfgc\_parameter\_type\_idc != 2 ) |  |
| **nnpfgc\_log2\_parameter\_bit\_length\_minus3** | u(2) |
| **nnpfgc\_num\_parameters\_idc** | u(6) |
| **nnpfgc\_num\_kmac\_operations\_idc** | ue(v) |
| **nnpfgc\_total\_kilobyte\_size** | ue(v) |
| } |  |
| } |  |

* + - 1. **Neural-network post-filter group characteristics SEI message semantics**

The neural-network post-filter group characteristics (NNPFGC) SEI message specifies a neural network post-filter (NNPF) group. It is indicated by the SEI message if the NNPF group defines an NNPF cascade or defines NNPFs or NNPF groups of NNPF cascades that are alternatives to each other. The use of NNPF groups of NNPF cascades for specific pictures is indicated with neural-network post-filter group activation (NNPFGA) SEI messages.

**nnpfgc\_id** contains an identifying number that may be used to identify an NNPF group. The value of nnpfgc\_id shall be in the range of 0 to 232 − 2, inclusive. Values of nnpfgc\_id from 256 to 511, inclusive, and from 231 to 232 − 2, inclusive, are reserved for future use by ITU-T | ISO/IEC. Decoders conforming to this edition of this document encountering an NNPFGC SEI message with nnpfgc\_id in the range of 256 to 511, inclusive, or in the range of 231 to 232 − 2, inclusive, shall ignore the SEI message. The value of nnpfgc\_id shall not be equal to any nnpfc\_id value of any NNPFC SEI message present in the same CLVS. When the value of nnpfgc\_id of an NNPFGC SEI message nnpfgcSeiA is equal to the value of nnpfgc\_id of another NNPFGC SEI message nnpfgcSeiB present in the same CLVS, nnpfgcSeiA and nnpfgcSeiB shall be identical.

**nnpfgc\_grouping\_type** equal to 0 indicates that this SEI message specifies a group of cascaded neural-network post-filters.

nnpfgc\_grouping\_type equal to 1 indicates that the NNPFs or NNPF groups identified by the nnpfgc\_member\_id[ i ] are alternatives to each other out of which the post-processor should select only one to be applied.

nnpfgc\_grouping\_type equal to 2 indicates that this SEI message specifies a group of NNPFs that are intended to be used jointly and are activated in an alternating manner so that at most one of these NNPFs is activate for any picture.

nnpfgc\_grouping\_type equal to 3 indicates that the NNPFs or NNPF groups identified by the nnpfgc\_member\_id[ i ] are intended to be used in parallel.

nnpfgc\_grouping\_type equal to 4 indicates that the NNPFs or NNPF groups identified by the nnpfgc\_member\_id[ i ] are optional, i.e., may or may not be applied by the post-processor.

The value of nnpfgc\_grouping\_type shall be in the range of 0 to 255, inclusive. Values of nnpfgc\_grouping\_type in the range of 5 to 255, inclusive, are reserved for future specification by ITU-T | ISO/IEC and shall not be present in bitstreams conforming to this edition of this document. Decoders conforming to this edition of this document shall ignore NNPFGC SEI messages with nnpfgc\_grouping\_type in the range of 5 to 255, inclusive.

**nnpfgc\_purpose** has the semantics of nnpfc\_purpose but with the exception that the semantics are specified for the NNPF group defined by this SEI message rather than the NNPF defined by an NNPFC SEI message.

**nnpfgc\_num\_members\_minus2** plus 2indicates the number of NNPFs or NNPF groups in the NNPF group that this SEI message defines.

**nnpfgc\_member\_id**[ i ] indicates the i-th member in the NNPF group defined by this SEI message as follows:

– If there is an NNPF with nnpfc\_id equal to nnpfgc\_member\_id[ i ] defined in the CLVS, the i-th member in the NNPF group defined by this SEI message is an NNPF that has nnpfc\_id equal to nnpfgc\_member\_id[ i ].

– Otherwise (there is no NNPF with nnpfc\_id equal to nnpfgc\_member\_id[ i ] defined in the CLVS), the i-th member in the NNPF group defined by this SEI message is an NNPF group with nnpfgc\_id equal to nnpfgc\_member\_id[ i ].

When an nnpfgc\_member\_id[ i ] value references an nnpfgc\_id value of an NNPFGC SEI message nnpfgcSei, it is a requirement of bitstream conformance that the NNPFGC SEI message nnpfgcSei shall have nnpfgc\_grouping\_type equal to 0. When nnpfgc\_grouping\_type is equal to 0 or 2, it is a requirement of bitstream conformance that there is an NNPF with nnpfc\_id value equal to nnpfgc\_member\_id[ i ] defined in the CLVS. When nnpfgc\_grouping\_type is equal to 1, 3, or 4, it is a requirement of bitstream conformance that there is an NNPF with nnpfc\_id value equal to nnpfgc\_member\_id[ i ] or an NNPF group with nnpfgc\_id value equal to nnpfgc\_member\_id[ i ] defined in the CLVS.

When nnpfgc\_grouping\_type is equal to 0, the NNPFs with nnpfc\_id equal to nnpfgc\_member\_id[ i ] are performed in cascade in increasing order of i, as activated by an NNPFGA SEI message with nnpfga\_target\_id equal to nnpfgc\_id.

nnpfgc\_complexity\_info\_present\_flag, nnpfgc\_parameter\_type\_idc, nnpfgc\_log2\_parameter\_bit\_length\_minus3, nnpfgc\_num\_parameters\_idc, nnpfgc\_num\_kmac\_operations\_idc, and nnpfgc\_total\_kilobyte\_size have the semantics of nnpfc\_complexity\_info\_present\_flag, nnpfc\_parameter\_type\_idc, nnpfc\_log2\_parameter\_bit\_length\_minus3, nnpfc\_num\_parameters\_idc, nnpfc\_num\_kmac\_operations\_idc, and nnpfc\_total\_kilobyte\_size, respectively, but with the exception that the semantics are specified for the NNPF group defined by this SEI message rather than the NNPF defined by an NNPFC SEI message. When nnpfgc\_grouping\_type is equal to 1, nnpfgc\_complexity\_info\_present\_flag shall be equal to 0.

* + 1. **Neural-network post-filter group activation SEI message**

[Ed. (MH): It is for further study if the extensions for the NNPFA SEI message that activate an NNPF group or the NNPF group activation SEI message is the best design choice, or if both would be justified.]

* + - 1. **Neural-network post-filter group activation SEI message syntax**

|  |  |
| --- | --- |
| nn\_post\_filter\_group\_activation( payloadSize ) { | **Descriptor** |
| **nnpfga\_target\_id** | ue(v) |
| **nnpfga\_cancel\_flag** | u(1) |
| if( !nnpfga\_cancel\_flag ) { |  |
| **nnpfga\_persistence\_flag** | u(1) |
| **nnpfga\_num\_filters\_minus2** | ue(v) |
| for( i = 0; i <= nnpfga\_num\_filters\_minus2 + 1; i++ ) { |  |
| **nnpfga\_target\_base\_flag**[ i ] | u(1) |
| **nnpfga\_input\_all\_pics\_flag**[ i ] | u(1) |
| if( !nnpfga\_input\_all\_pics\_flag[ i ] ) { |  |
| **nnpfga\_num\_input\_pics\_minus1**[ i ] | ue(v) |
| for( j = 0; j <= nnpfga\_num\_input\_pics\_minus1[ i ]; j++ ) |  |
| **nnpfga\_input\_pic\_skip\_count**[ i ][ j ] | ue(v) |
| } |  |
| **nnpfga\_num\_output\_entries**[ i ] | ue(v) |
| for( j = 0; j < nnpfga\_num\_output\_entries[ i ]; j++ ) |  |
| **nnpfga\_output\_flag**[ i ][ j ] | u(1) |
| } |  |
| } |  |
| } |  |

* + - 1. **Neural-network post-filter group activation SEI message semantics**

The neural-network post-filter group activation (NNPFGA) SEI message activates or de-activates the possible use of the target neural-network post-processing filter group (NNPFG) of NNPF groups, identified by nnpfga\_target\_id, for post-processing filtering of a set of pictures. nnpfgc\_grouping\_type for the identfied NNPF group shall be equal to 0 (cascade) or 1 (alternatives). When nnpfgc\_grouping\_type is equal to 1, each member of the group shall have the same number of input pictures and NNPF output pictures. For a particular picture for which the NNPFG is activated, the target NNPFG is the NNPFG specified by the last NNPFGC SEI message with nnpfgc\_id equal to nnpfga\_target\_id, that precedes the first VCL NAL unit of the current picture in decoding order and the NNPFs of the target NNPFG are defined by the NNPFC SEI messages that have nnpfc\_id equal to any nnpfgc\_member\_id[ i ] value of the target NNPFG and are present in the current picture unit or precede the current picture in decoding order.

Use of this SEI message requires the definition of the following variables:

– Input picture width and height in units of luma samples, denoted herein by InitCroppedWidth[ idx ] and InitCroppedHeight[ idx ], respectively, of the candidate input pictures with index idx in the range of 0 to numCandInputPics − 1, inclusive, that may be used as input for the NNPFG.

– Luma sample array InitCroppedYPic[ idx ] and chroma sample arrays InitCroppedCbPic[ idx ] and InitCroppedCrPic[ idx ], when present, of the candidate input pictures with index idx in the range of 0 to numCandInputPics − 1, inclusive, that may be used as input for the NNPFG.

– Bit depth BitDepthY for the luma sample array of the candidate input pictures.

– Bit depth BitDepthC for the chroma sample arrays, if any, of the candidate input pictures.

– A chroma format indicator, denoted herein by ChromaFormatIdc, as described in subclause ‎7.3.

– When nnpfc\_auxiliary\_inp\_idc is equal to 1, a filtering strength control value array StrengthControlVal[ idx ] that shall contain real numbers in the range of 0 to 1, inclusive, of the candidate input pictures with index idx in the range of 0 to numCandInputPics − 1, inclusive.

Candidate input picture with index 0 corresponds to the picture for which the NNPFG is activated by this NNPFGA SEI message. Candidate input picture with index i in the range of 1 to numCandInputPics − 1, inclusive, precedes the candidate input picture with index i − 1 in output order. Let candInputPicList[ 0 ] be the list of candidate input pictures in inverse output order.

**nnpfga\_target\_id** indicates the target NNPFG, which is specified by the NNPFGC SEI message that pertains to the current picture and have nnpfgc\_id equal to nnpfga\_target\_id.

The value of nnpfga\_target\_id shall be in the range of 0 to 232 − 2, inclusive.

An NNPFGA SEI message with a particular value of nnpfga\_target\_id shall not be present in a current PU unless there is an NNPFGC SEI message with nnpfgc\_id equal to the particular value of nnpfga\_target\_id and nnpfgc\_grouping\_type equal to 0 present in the current PU or in a PU that precedes the current PU in decoding order within the current CLVS.

When a PU contains both an NNPFGC SEI message with a particular value of nnpfgc\_id and an NNPFGA SEI message with nnpfga\_target\_id equal to the particular value of nnpfgc\_id, the NNPFGC SEI message shall precede the NNPFGA SEI message in decoding order.

**nnpfga\_cancel\_flag** equal to 1 indicates that the persistence of the target NNPFG established by any previous NNPFGA SEI message with the same nnpfga\_target\_id as the current SEI message is cancelled, i.e., the target NNPFG is no longer used unless it is activated by another NNPFGA SEI message with the same nnpfga\_target\_id as the current SEI message and nnpfga\_cancel\_flag equal to 0. nnpfga\_cancel\_flag equal to 0 indicates that the target NNPFG is activated for use.

**nnpfga\_persistence\_flag** specifies the persistence of the target NNPFG for the current layer.

nnpfga\_persistence\_flag equal to 0 specifies that the target NNPFG may be used for post-processing filtering for the current picture only.

nnpfga\_persistence\_flag equal to 1 specifies that the target NNPFG may be used for post-processing filtering for the current picture and all subsequent pictures of the current layer in output order until one or more of the following conditions are true:

– A new CLVS of the current layer begins.

– The bitstream ends.

– A picture in the current layer associated with a NNPFGA SEI message with the same nnpfga\_target\_id as the current SEI message that follows the current picture in output order.

NOTE – The target NNPFG is not applied for this subsequent picture in the current layer associated with a NNPFGA SEI message with the same nnpfga\_target\_id as the current SEI message.

Let the nnpfgcTargetPictures be the set of pictures to which the last NNPFGC SEI message with nnpfgc\_id equal to nnpfga\_target\_id that precedes the current NNPFGA SEI message in decoding order pertains. Let nnpfgaTargetPictures be the set of pictures for which the target NNPFG is activated by the current NNPFGA SEI message. It is a requirement of bitstream conformance that any picture included in nnpfgaTargetPictures shall also be included in nnpfgcTargetPictures.

**nnpfga\_num\_filters\_minus2** plus 2indicates the number of NNPFs in the NNPFG that this SEI message activates. The value of nnpfga\_num\_filters\_minus2 shall be equal to the value of nnpfgc\_num\_members\_minus2 in an NNPFGC SEI message with nnpfgc\_id equal to nnpfga\_target\_id.

**nnpfga\_target\_base\_flag**[ i ] equal to 1 specifies that the i-th NNPF in the target NNPFG is the base NNPF with nnpfc\_id equal to nnpfgc\_member\_id[ i ] in an NNPFGC SEI message with nnpfgc\_id equal to nnpfga\_target\_id. nnpfga\_target\_base\_flag[ i ] equal to 0 specifies that the i-th NNPF in the target NNPFG is the NNPF specified by the last NNPFC SEI message that has nnpfc\_id equal to nnpfgc\_member\_id[ i ] in an NNPFGC SEI message with nnpfgc\_id equal to nnpfga\_target\_id, precedes the first VCL NAL unit of the current picture in decoding order, and is not a repetition of the NNPFC SEI message that contains the base NNPF.

**nnpfga\_input\_all\_pics\_flag**[ i ] equal to 1 specifies that the input pictures to the i-th NNPF are selected from the list of candidate input pictures candInputPicList[ i ] without skipping. nnpfga\_input\_all\_pics\_flag[ i ] equal to 0 specifies that the input pictures to the i-th NNPF are selected from the list of candidate input pictures candInputPicList[ i ] in a manner that some candidate input pictures are skipped.

**nnpfga\_num\_input\_pics\_minus1**[ i ] specifies the number of input pictures for the i-th NNPF in the target NNPFG. When present, nnpfga\_num\_input\_pics\_minus1[ i ] shall be equal to nnpfc\_num\_input\_pics\_minus1 for an NNPF with nnpfc\_id equal to nnpfgc\_member\_id[ i ] of an NNPFGC SEI message with nnpfgc\_id equal to nnpfga\_target\_id. When not present, nnpfga\_num\_input\_pics\_minus1[ i ] is inferred to be equal to nnpfc\_num\_input\_pics\_minus1 for an NNPF with nnpfc\_id equal to nnpfgc\_member\_id[ i ] in an NNPFGC SEI message with nnpfgc\_id equal to nnpfga\_target\_id.

**nnpfga\_input\_pic\_skip\_count**[ i ][ j ] specifies a j-th picture count that is skipped in the list of candidate input pictures candInputPicList[ i ] when selecting input pictures for the NNPF activated by the i-th loop entry. When nnpfga\_input\_pic\_skip\_count[ i ][ j ] is not present, it is inferred to be equal to 0 for all values of j in the range of 0 to nnpfga\_num\_input\_pics\_minus1[ i ], inclusive. The variable numCandInputPics, which indicates the number of candidate input pictures to the NNPFG, is derived as follows:

numCandInputPics = 0  
for( j = 0; j <= nnpfga\_num\_input\_pics\_minus1[ 0 ]; j++ )  
 numCandInputPics += 1 + nnpfga\_input\_pic\_skip\_count[ 0 ][ j ] (xx)

Let candInputPicList[ m ] for m in the range of 1 to nnpfga\_num\_filters\_minus2 + 1, inclusive, be a list of pictures in inverse output order that is initially empty and formed in decreasing order of n in the range of 0 to m − 1, inclusive, by including each picture that is output by the NNPF process of the n-th loop entry that has no corresponding picture already present in candInputPicList[ m ], and lastly including each picture present in candInputPicList[ 0 ] that has no corresponding picture already present in candInputPicList[ m ].

When a candidate input picture candInputPicList[ m ][ idx ] for any value of m in the range of 1 to nnpfga\_num\_filters\_minus2 + 1, inclusive, is an NNPF output picture of the n-th NNPF process with the value of n being less than the value of m, the width and height of the candidate input picture are respectively equal to nnpfcOutputPicWidth and nnpfcOutputPicHeight of the NNPF output picture.

The list of input pictures inputPicList[ m ] to the NNPF of the m-th loop entry is derived as follows:

for( k = 0, candIdx = 0; k <= nnpfga\_num\_input\_pics\_minus1[ m ]; k++, candIdx++ ) {  
 candIdx += nnpfga\_input\_pic\_skip\_count[ m ][ k ]  
 inputPicList[ m ][ k ] = candInputPicList[ m ][ candIdx ] (xx)  
}

It is a requirement of bitstream conformance that candIdx shall not exceed the number of pictures in candInputPicList[ m ].

It is a requirement of bitstream conformance that the pictures present in inputPicList[ m ], for any value of m in the range of 1 to nnpfga\_num\_filters\_minus2 + 1, inclusive, shall have the same width, height, bit depth, and chroma format.

For purposes of interpretation of the NNPFC SEI message with nnpfc\_id equal to nnpfgc\_member\_id[ i ] in an NNPFGC SEI message with nnpfgc\_id equal to nnpfga\_target\_id, the following variables are specified for the i-th loop entry:

– The variables BitDepthY, BitDepthC, and ChromaFormatIdc are used as provided for the interpretation of this SEI message.

– CroppedWidth and CroppedHeight are set equal to the width and height of the pictures in inputPicList[ i ], respectively, in units of luma samples.

– For each input picture k in the range of 0 to nnpfga\_num\_input\_pics\_minus1[ i ], inclusive, the following applies:

– CroppedYPic[ k ], CroppedCbPic[ k ], and CroppedCrPic[ k ], when present, are set equal to respective sample array of inputPicList[ i ][ k ]

– When nnpfc\_auxiliary\_inp\_idc is equal to 1 for the NNPF with nnpfc\_id equal to nnpfgc\_member\_id[ i ] in an NNPFGC SEI message with nnpfgc\_id equal to nnpfga\_target\_id, the following applies:

– It is a requirement of bitstream conformance that inputPicList[ i ][ k ] is the same as candInputPicList[ 0 ][ idx ] for any value of idx in the range of 0 to numCandInputPics − 1, inclusive.

– StrengthControlVal[ k ] is set equal to InitStrengthControlVal[ idx ].

**nnpfga\_num\_output\_entries**[ i ] specifies the number of nnpfga\_output\_flag[ i ][ j ] syntax elements present in the NNPFGA SEI message. The value of nnpfga\_num\_output\_entries[ i ] shall be in the range of 0 to NumInpPicsInOutputTensor, inclusive, for an NNPF with nnpfc\_id equal to nnpfgc\_member\_id[ i ] of an NNPFGC SEI message with nnpfgc\_id equal to nnpfga\_target\_id..

**nnpfga\_output\_flag**[ i ][ j ] equal to 1 specifies that the NNPF-generated picture that corresponds to the input picture having index InpIdx[ j ] derived for the i-th NNPF of the target NNPFG is output by the NNPF process activated by this loop entry, where the NNPF process is specified in the semantics of the NNPFC SEI message. nnpfga\_output\_flag[ i ][ j ] equal to 0 specifies that the NNPF-generated picture that corresponds to the input picture having index InpIdx[ j ] derived for the i-th NNPF of the target NNPFG is not output by the NNPF process activated by this loop entry. When nnpfga\_num\_output\_entries[ i ] is less than NumInpPicsInOutputTensor derived for the i-th NNPF of the target NNPFG, nnpfga\_output\_flag[ i ][ j ] is inferred to be equal to 1 for each value of i in the range of nnpfga\_num\_output\_entries[ i ] to NumInpPicsInOutputTensor − 1, inclusive.

Let NnpfgaOutputPicList, which is the list of pictures output by NNPF process of the NNPFG in output order, be initially empty and formed in decreasing order of n in the range of 0 to nnpfga\_num\_filters\_minus2 + 1, inclusive, by including each picture that is output by the NNPF process of the n-th loop entry that has no corresponding picture already present in NnpfgaOutputPicList.

Renumber the current subclause 8.30 as 8.33 and add subclauses 8.30 to 8.32 as follows:

* 1. **Encoder optimization information SEI message**
     1. **Encoder optimization information SEI message syntax**

|  |  |
| --- | --- |
| encoder\_optimization\_info(payloadSize ) { | **Descriptor** |
| **optimization\_cancel\_flag** | u(1) |
| if( !optimization\_cancel\_flag ){ |  |
| **optimization\_persistence\_flag** | u(1) |
| **optimization\_for\_human\_viewing\_flag** | u(1) |
| **optimization\_for\_machine\_analysis\_flag** | u(1) |
| **optimization\_type** | u(16) |
| } |  |
| } |  |

* + 1. **Encoder optimization information SEI message semantics**

**optimization\_cancel\_flag** equal to 1 specifies that the persistence of the previous applied optimization is cancelled. optimization\_cancel\_flag equal to 0 indicates that optimization\_persistence\_flag and optimization\_method follow.

**optimization\_persistence\_flag** specifies the persistence of the optimization the current layer. optimization\_persistence\_flag equal to 0 specifies that the optimization applies for the current picture only. optimization\_persistence\_flag equal to 1 specifies that the optimization applies for the current picture and all subsequent pictures of the current layer in output order until one or more of the following conditions are true:

– A new CLVS of the current layer begins.

– The bitstream ends.

– A picture in the current layer associated with an encoder\_optimization\_info SEI message is output that follows the current picture in output order.

**optimization\_for\_human\_viewing\_flag** equal to 1 specifies that purposes for the applied optimization include human viewing. optimization\_for\_human\_viewing\_flag equal to 0 specifies that purposes for the applied optimization may or may not include human viewing.

**optimization\_for\_machine\_analysis\_flag** equal to 1 specifies that purposes for the applied optimization include machine analsysis. optimization\_for\_machine\_analysis\_flag equal to 0 specifies that purposes for the applied optimization may or may not include machine analysis.

**optimization\_type** indicates the types of optimization method as specified in Table 2.

**Table x – Definition of** **optimization\_type**

|  |  |
| --- | --- |
| **Value** | **Interpretation** |
| optimization\_type  = =  0 | May be used as determined by the application |
| optimization\_type > 0  && ( optimization\_type & 0x01 )  = =  0 | No object-based optimization |
| ( optimization\_type & 0x01 )  !=  0 | With object-based optimization; the pictures for which this SEI message persists have been pre-processed or encoded so that detected objects in the pictures are optimized with respect to other parts of the pictures for the indicated optimization purposes |
| optimization\_type > 0  && ( optimization\_type & 0x02 )  = =  0 | No temporal subsampling optimization |
| ( optimization\_type & 0x02 )  !=  0 | With temporal subsampling optimization |
| optimization\_type > 0  && ( optimization\_type & 0x04 )  = =  0 | No spatial subsampling optimization |
| ( optimization\_type & 0x04 )  !=  0 | With spatial subsampling optimization |
| optimization\_type > 0  && ( optimization\_type & 0x08 )  = =  0 | No temporal quality optimization |
| ( optimization\_type & 0x08 )  !=  0 | With temporal quality optimization |
| optimization\_type > 0  && ( optimization\_type & 0x10 )  = =  0 | No spatial quality optimization |
| ( optimization\_type & 0x10 )  !=  0 | With spatial quality optimization; the pictures for which this SEI message persists have been pre-processed or encoded to reduce unnecessary information or improve the quality of necessary information.(e.g to reduce the amount of noise and remove speckles at the picture-level) |

* 1. **Source picture timing information SEI message**
     1. **Source picture timing information SEI message syntax**

|  |  |
| --- | --- |
| source\_picture\_timing\_info( payloadSize ) { | **Descriptor** |
| **spti\_cancel\_flag** | u(1) |
| if( !spti\_cancel\_flag ) { |  |
| **spti\_persistence\_flag** | u(1) |
| **spti\_source\_picture\_timing\_type** | u(8) |
| **spti\_source\_timing\_equals\_output\_timing\_flag** | u(1) |
| if( !spti\_source\_timing\_equals\_output\_timing\_flag ) { |  |
| **spti\_time\_scale** | u(32) |
| **spti\_num\_units\_in\_elemental\_source\_picture\_interval** | u(32) |
| **spti\_max\_sublayers\_minus\_1** | u(3) |
| for( i = 0; i  <=  spti\_max\_sublayers\_minus1; i++ ) { |  |
| **spti\_sublayer\_source\_picture\_interval\_scale\_factor**[ i ] | ue(v) |
| **spti\_sublayer\_synthesized\_picture\_flag**[ i ] | u(1) |
| } |  |
| } |  |
| **}** |  |
| } |  |

* + 1. **Source picture timing information SEI message semantics**

The source picture timing information (SPTI) SEI message indicates the temporal distance between source pictures associated with the corresponding decoded output pictures prior to encoding, e.g., for camera-captured content, the temporal distance between source pictures is the difference between the time at which an image sensor was exposed to produce a source picture associated with the current decoded picture and the time at which the image sensor was exposed to produce the source picture associated with a previous decoded picture in output order.

**spti\_cancel\_flag** equal to 1 indicates that the SPTI SEI message cancels the persistence of any previous SPTI SEI message in output order that applies to the current layer. spti\_cancel\_flag equal to 0 indicates that source picture timing information follows.

**spti\_persistence\_flag** specifies the persistence of the SPTI SEI message for the current layer.

spti\_persistence\_flag equal to 0 specifies that the SPTI SEI message applies to the current decoded picture only.

spti\_persistence\_flag equal to 1 specifies that the SPTI SEI message applies to the current decoded picture and persists for all subsequent pictures of the current layer in output order until one or more of the following conditions are true:

– A new CLVS of the current layer begins.

– The bitstream ends.

– A picture in the current layer in an AU associated with an SPTI SEI message is output that follows the current picture in output order.

**spti\_source\_picture\_timing\_type** indicates the timing relationship between source pictures and corresponding decoded output pictures as specified in Table 1, where ( spti\_source\_picture\_timing\_type & bitMask ) not equal to 0 indicates that the timing relationship has the interpretation associated with the bitMask value in Table 1. When spti\_source\_picture\_timing\_type is greater than 0 and ( spti\_source\_picture\_timing\_type & bitMask ) is equal to 0, the interpretation associated with the bitMask value is not applicable to the SPTI SEI message. When spti\_source\_picture\_timing\_type is equal to 0, the timing relationship may be specified by the application.

The value of spti\_source\_picture\_timing\_type shall be in the range of 0 to 127, inclusive, in bitstreams conforming to this edition of this document. Values of 128 to 255, inclusive, for spti\_source\_picture\_timing\_type are reserved for future use by ITU-T | ISO/IEC and shall not be present in bitstreams conforming to this edition of this document. Decoders conforming to this edition of this document shall ignore SPTI SEI messages with spti\_source\_picture\_timing\_type in the range of 128 to 255, inclusive.

**Table 1 – Interpretation of spti\_source\_picture\_timing\_type**

|  |  |
| --- | --- |
| **bitMask** | **Interpretation** |
| 0x01 | Slow motion |
| 0x02 | Frame rate conversion |
| 0x04 | High-speed imaging |
| 0x08 | Time-lapse imaging |
| 0x10 | Temporal reversal |
| 0x20 | Still image / freeze frame |
| 0x40 | Sporadic or event-driven |

The variable temporalReversalFlag is equal to ( spti\_source\_picture\_timing\_type & 0x10 )? 1 : 0.

**spti\_source\_timing\_equals\_output\_timing\_flag** equal to 1 indicates the timing of source pictures is the same as the timing of corresponding decoded output pictures. spti\_source\_timing\_equals\_output\_timing\_flag equal to 0 indicates the timing of source pictures might not be the same as the timing of corresponding decoded output pictures.

When spti\_source\_timing\_equals\_output\_timing\_flag is equal to 1 and a picture timing SEI message is present for the current picture, source picture timing could be determined from information conveyed in the picture timing SEI message.

**spti\_time\_scale** specifies the number of time units that pass in one second. The value of spti\_time\_scale shall not be equal to 0. For example, a time coordinate system that measures time using a 27 MHz clock has an spti\_time\_scale of 27,000,000.

**spti\_num\_units\_in\_elemental\_source\_picture\_interval** specifies the number of time units of a clock operating at the frequency spti\_time\_scale Hz that corresponds to the indicated elemental source picture interval of consecutive pictures in output order in the CLVS.

The indicated elemental source picture interval, also to be denoted by the variable ElementalSourcePictureInterval, in units of seconds, is equal to the quotient of spti\_num\_units\_in\_elemental\_source\_picture\_interval divided by spti\_time\_scale. For example, to represent an elemental source picture interval equal to 0.04 seconds, spti\_time\_scale may be equal to 27,000,000 and spti\_num\_units\_in\_elemental\_source\_picture\_interval may be equal to 1,080,000.

**spti\_max\_sublayers\_minus\_1** plus 1 specifies the maximum number of temporal sublayers that may be present in the CLVS.

**spti\_sublayer\_source\_picture\_interval\_scale\_factor**[ i ], when present, specifies a scale factor used in determining the source picture interval of corresponding consecutive pictures in output order in the CLVS having TemporalId less than or equal to i. The value 0 may be used to indicate that the source picture corresponding to the current decoded output picture is identical to the source picture corresponding to the previous decoded output picture.

The indicated source picture interval associated with output pictures having TemporalId less than or equal to i, denoted by the variable SourcePictureInterval[ i ], in units of seconds, is derived as follows:

SourcePictureInterval[ i ] = ElementalSourcePictureInterval \* spti\_sublayer\_source\_picture\_interval\_scale\_factor[ i ]

When picture n is a picture that is output having TemporalId less than or equal to i and is not the first picture of the bitstream that is output, the value of the variable SourcePictureTime[ n ] is derived as follows:

– If temporalReversalFlag is equal to 0, SourcePictureTime[ n ] = SourcePictureTime[ previousPicInOutputOrder ] + SourcePictureInterval[ i ]

– Otherwise, temporalReversalFlag is equal to 1, SourcePictureTime[ n ] = SourcePictureTime[ previousPicInOutputOrder ] - SourcePictureInterval[ i ]

where previousPicInOutputOrder is the last picture that is output having TemporalId less than or equal to i that precedes picture n in output order (if any). When the value of SourcePictureTime[ 0 ] is not provided by external means not specified in this document, the value of SourcePictureTime[ 0 ] is inferred to be 0.

**spti\_sublayer\_synthesized\_picture\_flag**[ i ], when present, equal to 1 indicates that decoded output pictures belonging to the ith temporal sublayer are synthesized and do not correspond to unmodified original source pictures. spti\_sublayer\_synthesized\_picture\_flag[ i ] equal to 0 provides no such indication. When not present, the value of spti\_sublayer\_synthesized\_picture\_flag[ i ] is inferred to be equal to 0.

* 1. **Object mask information SEI message**
     1. **Object mask information SEI message syntax**

|  |  |
| --- | --- |
| object\_mask\_info( payloadSize ) { | **Descriptor** |
| **omi\_cancel\_flag** | u(1) |
| if( !om\_cancel\_flag ) { |  |
| **omi\_aux\_id\_minus128** | ue(v) |
| **omi\_num\_primary\_pic\_layer\_minus1** | ue(v) |
| for(i = 0; i <= omi\_num\_primary\_pic\_layer\_minus1; i++ ) |  |
| **omi\_primary\_pic\_layer\_id**[ i ] | ue(v) |
| **omi\_mask\_id\_length\_minus8** | ue(v) |
| **omi\_mask\_confidence\_info\_present\_flag** | u(1) |
| if( omi\_mask\_confidence\_info\_present\_flag ) |  |
| **omi\_mask\_confidence\_length\_minus1** | u(4) |
| **omi\_mask\_depth\_info\_present\_flag** | u(1) |
| if( omi\_mask\_depth\_info\_present\_flag ) |  |
| **omi\_mask\_depth\_length\_minus1** | u(4) |
| **omi\_mask\_label\_info\_present\_flag** | u(1) |
| if( omi\_mask\_label\_info\_present\_flag ) { |  |
| **omi\_mask\_label\_language\_present\_flag** | u(1) |
| if( omi\_mask\_label\_language\_present\_flag ) { |  |
| while( !byte\_aligned( ) ) |  |
| **omi\_bit\_equal\_to\_zero** | f(1) |
| **omi\_mask\_lable\_language** | st(v) |
| } |  |
| } |  |
| for( i = 0; i <= omi\_num\_primary\_pic\_layer\_minus1; i++ ) { |  |
| for( j = 0; j < numAuxLayer[ omi\_primary\_pic\_layer\_id[ i ] ]; j++ ){ |  |
| **omi\_mask\_pic\_update\_flag**[ i ][ j ] | f(1) |
| if( omi\_mask\_pic\_update\_flag[ i ][ j ]) { |  |
| **omi\_num\_mask\_in\_pic\_update**[ i ][ j ] | ue(v) |
| for( k = 0; k < omi\_num\_mask\_in\_pic\_update[ i ][ j ]; k++ ) { |  |
| **omi\_mask\_id**[ i ][ j ][ k ] | u(v) |
| **omi\_mask\_top**[ i ][ j ][ k ] | ue(v) |
| **omi\_mask\_left**[ i ][ j ][ k ] | ue(v) |
| **omi\_mask\_width**[ i ][ j ][ k ] | ue(v) |
| **omi\_mask\_height**[ i ][ j ][ k ] | ue(v) |
| if( maskIdExist[ i ][ j ][ omi\_mask\_id[ i ][ j ][ k ] ] ) { |  |
| **omi\_mask\_cancel**[ i ][ j ][ k ] | u(1) |
| maskIdExist[ i ][ j ][ omi\_mask\_id[ i ][ j ][ k ] ] =  !omi\_mask\_cancel[ i ][ j ][ k ] |  |
| } |  |
| else |  |
| maskIdExist[ i ][ j ][ omi\_mask\_id[ i ][ j ][ k ] ] = 1 |  |
| if( maskIdExist[ i ][ j ][ omi\_mask\_id[ i ][ j ][ k ] ] ) { |  |
| if( omi\_mask\_confidence\_info\_present\_flag ) |  |
| **omi\_mask\_confidence**[ i ][ j ][ k ] | u(v) |
| if( omi\_mask\_depth\_info\_present\_flag ) |  |
| **omi\_mask\_depth**[ i ][ j ][ k ] | u(v) |
| while( !byte\_aligned( ) ) |  |
| **omi\_bit\_equal\_to\_zero** | f(1) |
| if( omi\_mask\_label\_info\_present\_flag **)** |  |
| **omi\_mask\_label**[ i ][ j ][ k ] | st(v) |
| } |  |
| } |  |
| } |  |
| } |  |
| } |  |

* + 1. **Object mask information information SEI message semantics**

The object mask information (OMI) SEI message provides information about object mask pictures coded as auxiliary pictures. Object mask auxiliary pictures have nuh\_layer\_id equal to sdi\_layer\_id[ i ] and sdi\_aux\_id [ i ] in the range of 128 to 159, inclusive, for any value of i in range of 0 to sid\_max\_layers\_minus1, inclusive.

NOTE 1 – Each object mask auxiliary picture layer is associated with one primary picture layer and one primary picture layer may be associated with one or more object mask auxiliary picture layers.

When an access unit contains an auxiliary picture picA in a layer, with nuh\_layer\_id equal to nuhLayerIdA, that is indicated as an object mask auxiliary layer by an OMI SEI message, and a primary picture picB in a layer, with nuh\_layer\_id equal to nuhLayerIdB, that is indicated as a primary layer by the OMI SEI message, OMI SEI message persists in output order until one or more of the following conditions are true:

– A CLVS containing the auxiliary picture picA ends.

– A CLVS containing the primary picture picB ends.

– A CVS ends.

– The bitstream ends.

**omi\_cancel\_flag** equal to 1 indicates that the SEI message cancels the persistence of any previous object mask information SEI message in output order that is associated with one or more primary picture layers to which this SEI applies. om\_cancel\_flag equal to 0 indicates that object mask information follows.

**omi\_aux\_id\_minus128** plus 128 indicates the value of sdi\_aux\_id of object mask auxiliary picture layer. om\_aux\_id\_minus128 shall be in the range of 0 to 31, inclusive.

When a CVS does not contain an SDI SEI message with sdi\_aux\_id[ i ] equal to omi\_aux\_id\_minus128 + 128 for at least one value of i, no picture in the CVS shall be associated with an OMI SEI message.

When an AU contains both an SDI SEI message with sdi\_aux\_id[ i ] equal to omi\_aux\_id\_minus128 + 128 for at least one value of i and an OMI SEI message, the SDI SEI message shall precede the OMI SEI message in decoding order.

**omi\_num\_primary\_pic\_layer\_minus1** plus 1indicates the number of primary picture layers associated with the object mask auxiliary picture layers to which this SEI message applies. The value of omi\_num\_primary\_pic\_layer\_minus1 shall be in the range of 0 to sdi\_max\_layers\_minus1.

**omi\_primary\_pic\_layer\_id**[ i ] specifies the nuh\_layer\_id value of the i-th primary picture layer to which this OMI SEI message applies. The value of sdi\_aux\_id[ j ] shall be equal to 0 for any value of j in the range of 0 to sid\_max\_layers\_minus1, inclusive, if sdi\_layer\_id[ j ] equal to omi\_primary\_pic\_layer\_id[ i ]

**omi\_mask\_id\_length\_minus8** plus 8 specifies the length, in bits, of omi\_mask\_id[ i ][ j ][ k ] syntax elements.

**omi\_mask\_confidence\_info\_present\_flag** equal to 1 indicates that omi\_mask\_confidence[ i ][ j ][ k ] syntax elements are present. omi\_mask\_confidence\_info\_present\_flag equal to 0 indicates that omi\_mask\_confidence[ i ][ j ][ k ]syntax elements are not present. It is a requirement of bitstream conformance that the value of omi\_mask\_confidence\_info\_present\_flag shall be the same for all object\_mask\_info( ) syntax structures within a CLVS.

**omi\_mask\_confidence\_length\_minus1** plus 1 specifies the length, in bits, of the omi\_mask\_confidence[ i ][ j ][ k ] syntax elements. It is a requirement of bitstream conformance that the value of omi\_mask\_confidence\_length\_minus1 shall be the same for all object\_mask\_info( ) syntax structures within a CLVS.

**omi\_object\_depth\_info\_present\_flag** equal to 1 indicates that omi\_object\_depth[ i ][ j ][ k ] syntax elements are present. omi\_object\_depth\_info\_present\_flag equal to 0 indicates that omi\_object\_depth[ i ][ j ][ k ]syntax elements are not present. It is a requirement of bitstream conformance that the value of omi\_object\_depth\_info\_present\_flag shall be the same for all object\_mask\_info( ) syntax structures within a CLVS.

**omi\_object\_depth\_length\_minus1** plus 1 specifies the length, in bits, of the omi\_object\_depth[ i ][ j ][ k ] syntax elements. It is a requirement of bitstream conformance that the value of omi\_object\_depth\_length\_minus1 shall be the same for all object\_mask\_info( ) syntax structures within a CLVS.

**omi\_mask\_label\_info\_present\_flag** equal to 1 indicates that omi\_mask\_label\_language\_present\_flag and omi\_mask\_label[ i ][ j ][ k ] syntax elements are present. omi\_mask\_label\_info\_present\_flag equal to 0 indicates that omi\_mask\_label\_language\_present\_flag and omi\_mask\_label[ i ][ j ][ k ] syntax elements are not present.

**omi\_mask\_label\_language\_present\_flag** equal to 1 indicates that omi\_mask\_lable\_language syntax element is present. omi\_mask\_label\_language\_present\_flag equal to 0 indicates that omi\_mask\_lable\_language syntax element is not present.

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

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

**omi\_mask\_pic\_update\_flag**[ i ][ j ] equal to 1 indicates the mask information of j-th object mask auxiliary picture associated with i-th primary picture is signalled. omi\_mask\_pic\_update\_flag[ i ][ j ] equal to 0 indicates the mask information of j-th object mask auxiliary picture associated with i-th primary picture is not signalled. When the mask information of j-th object mask auxiliary picture associated with i-th primary picture is not present, the persistence mechanism is used, that is the information is inherited from the last OMI SEI message which signals the mask information of j-th object mask auxiliary picture associated with i-th primary picture.

**omi\_num\_mask\_in\_pic\_update**[ i ][ j ] indicates the number of object masks of which the information to be signalled in the j-th auxiliary picture associated with i-th primary picture. omi\_num\_mask\_in\_pic\_update [ i ][ j ] shall be in the range of 0 to (1<<BitDepthY) – 1, inclusive, where BitDepthY is the bit depth for the samples of the luma component. The variable omiNumMaskInPic[ i ][ j ] indicating the number of object masks in the j-th auxiliary picture associated with i-th primary picture is set to omi\_num\_mask\_in\_pic\_update[ i ][ j ] when the current SEI message is the first OMI SEI message in the current CLVS.

The variable numAuxLayer[ primaryLayerId ] indicates the number of the auxiliary picture layers associated with primary picture layer with nuh\_layer\_id equal to primaryLayerId. The variable associatedAuxLayerId[ primaryLayerId ][ i ] indicates the value of nuh\_layer\_id of the i-th auxiliary picture layer associated with primary picture layer with nuh\_layer\_id equal to primaryLayerId. numAuxLayer[ primaryLayerId ] and associatedAuxLayerId[ primaryLayerId ][ i ] are derived as follows.

for( i = 0; i <= sdi\_max\_max\_layers\_minus1; i++ )  
 numAuxLayer[ sdi\_layer\_id[ i ] ] = 0;  
for( i = 0; i <= sdi\_max\_layers\_minus1; i++ ){  
 if( sdi\_aux\_id[ i ] == omi\_aux\_id\_minus128 + 128 ){  
 for( j = 0; j <= sdi\_num\_associated\_primary\_layers\_minus1[ i ]; j++ ) { (xx)  
 primaryLayerId = sdi\_layer\_id[ sdi\_associated\_primary\_layer\_idx[ i ][ j ] ];  
 associatedAuxLayerId[ primaryLayerId ][ numAuxLayer[ primaryLayerId ] ] = sdi\_layer\_id[ i ];  
 numAuxLayer[ primaryLayerId ]++;  
 }  
 }  
}

**omi\_mask\_id**[ i ][ j ][ k ] indicates the identifier of k-th object mask in the j-th object mask auxiliary picture associated with the i-th primary picture..

The variable maskId[ i ][ j ][ k ] specifying the object mask identifier of k-th object mask in the j-th object mask auxiliary picture associated with i-th primary picture in the SEI message is derived as follows:

for( i = 0; i <= omi\_num\_primary\_pic\_layer\_minus1; i++ ) {  
 for( j = 0; j < numAuxLayer[omi\_primary\_pic\_layer\_id[ i ]]; j++ ) {   
 for( k = 0; k < omiNumMaskInPic[ i ][ j ]; k++ ) {   
 maskId[ i ][ j ][ k ] = omi\_mask\_id[ i ][ j ][ k ] + (1<<BitDepthY)\*j (xx)  
 }  
 }  
}

**omi\_mask\_top**[ i ][ j ][ k ], **omi\_mask\_left**[ i ][ j ][ k ], **omi\_mask\_width**[ i ][ j ][ k ], and **omi\_mask\_height**[ i ][ j ][ k ] indicate the coordinates of the top-left corner and the width and height, respectively, of the bounding box of the object mask with identifier equal to omi\_mask\_id[ i ][ j ][ k ].

The value of omi\_mask\_left[ i ][ j ][ k ] shall be in the range of 0 to ( CroppedWidth / SubWidthC – 1 ), inclusive, CroppedWidth and SubWidthC being associated to the the j-th object mask auxiliary picture associated with i-th primary picture.

The value of omi\_mask\_top[ i ][ j ][ k ] shall be in the range of 0 to ( CroppedHeight / SubHeightC – 1 ), inclusive, CroppedHeight  and SubHeightC  being associated to the the j-th object mask auxiliary picture associated with i-th primary picture.

The value of omi\_mask\_width[ i ][ j ][ k ] shall be in the range of 0 to ( CroppedWidth / SubWidthC − omi\_mask\_left[ i ][ j ][ k ] ), inclusive.

The value of omi\_mask\_height[ i ][ j ][ k ] shall be in the range of 0 to ( CroppedHeight / SubHeightC − omi\_mask\_top[ i ][ j ][ k ] ), inclusive.

Variable I[ i ][ j ][ x ][ y ] is the decoded value of the sample at the relative sample location (x, y) in the j-th object mask auxiliary picture associated with the i-th primary picture. The following process is to determine each mask region in each auxiliary picture.

for( i = 0; i <= omi\_num\_primary\_pic\_layer\_minus1; i++ ) {  
 for( j = 0; j < numAuxLayer[omi\_primary\_pic\_layer\_id[ i ]]; j++ ) {   
 for( k = 0; k < omiNumMaskInPic[ i ][ j ]; k++ ) { (xx)  
 if( pI[ i ][ j ][ x ][ y ] == omi\_mask\_id[ i ][ j ][ k ]  
 && x >= omi\_mask\_left[ i ][ j ][ k ]   
 && x < omi\_mask\_left[ i ][ j ][ k ] + omi\_mask\_width[ i ][ j ][ k ]  
 && y >= omi\_mask\_top[ i ][ j ][ k ]   
 && y < omi\_mask\_top[ i ][ j ][ k ] + omi\_mask\_height[ i ][ j ][ k ] )  
 The sample at location (x, y) in the j-th object mask auxiliary picture associated with   
 the i-th primary picture is associated with the object mask with the identifier of  
 maskId[ i ][ j ][ k ]  
 }  
 }  
}

**omi\_mask\_cancel**[ i ][ j ][ k ] equal to 1 cancels the persistence scop of object mask with identifier equal to om\_mask\_id[ i ][ j ][ k ]. omi\_mask\_cancel[ i ][ j ][ k ] equal to 0 indicates the information of object mask with identifier equal to omi\_mask\_id[ i ][ j ] is signalled.

The variable maskIdExist[ i ][ j ][ id ] equal to 1 indicates the object mask with identifier equal to id in the j-th object mask auxiliary picture associated with i-th primary picture exists. The variable maskIdExist[ i ][ j ][ id ] equal to 0 indicates the object mask with identifier equal to id in the j-th object mask auxiliary picture associate with i-th primary picture does not exist. maskIdExist[ i ][ j ][ id ] is initialized with 0 before decoding the current CVS.

**omi\_mask\_confidence**[ i ][ j ][  ] indicates the degree of confidence associated with the k-th object mask in the j-th object mask auxiliary picture associated with i-th primary picture, in units of 2-( omi\_mask\_confidence\_length\_minus1 + 1 ), such that a higher value of omi\_mask\_confidence[ i ][ j ][ k ] indicates a higher degree of confidence. The length of the omi\_mask\_confidence[ i ][ j ][ k ] syntax element is omi\_mask\_confidence\_length\_minus1 + 1 bits.

**omi\_mask\_depth**[ i ][ j ][ k ] indicates the object depth associated with the k-th object mask in the j-th object mask auxiliary picture associated with i-th primary picture. A smaller value of omi\_mask\_depth indicates a shorter distance to the object. The length of the omi\_mask\_depth[ i ][ j ][ k ] syntax element is omi\_object\_depth\_length\_minus1 + 1 bits.

**omi\_mask\_label**[ i ][ j ][ k ] specifies the contents of the label associated with k-th object mask in the j-th object mask auxiliary picture associated with i-th primary picture. The length of the omi\_mask\_label[ i ][ j ][ k ] syntax element shall be less than or equal to 255 bytes, not including the null termination byte.