FINAL
DRAFT

INTERNATIONAL
STANDARD

**ISO/IEC
FDIS
23090-3**

ISO/IEC JTC **1**/SC **29**

Secretariat: JISC

Voting begins on:
2020-11-24

Voting terminates on:
2021-01-19

Information technology — Coded representation of immersive media —

Part 3:
Versatile video coding

Reference number
ISO/IEC FDIS 23090-3:2020(E)

ISO copyright office
CP 401 • Ch. de Blandonnet 8
CH-1214 Vernier, Geneva
Phone: +41 22 749 01 11
Email: copyright@iso.org
Website: www.iso.org

Published in Switzerland

Contents

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).

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) or the IEC list of patent declarations received (see 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.

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. The technically identical text is published as ITU-T H.266 (08/2020).

A list of all parts in the ISO/IEC 23090 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.

Introduction

Purpose

This document specifies a video coding technology known as versatile video coding. It has been designed with two primary goals. The first of these is to specify a video coding technology with a compression capability that is substantially beyond that of the prior generations of such standards, and the second is for this technology to be highly versatile for effective use in a broader range of applications than that addressed by prior standards. Some key application areas for the use of this document particularly include ultra-high-definition video (e.g., with 3840×2160 or 7620×4320 picture resolution and bit depth of 10 bits as specified in Rec. ITU-R BT.2100), video with a high dynamic range and wide colour gamut (e.g., with the perceptual quantization or hybrid log-gamma transfer characteristics specified in Rec. ITU-R BT.2100), and video for immersive media applications such as 360° omnidirectional video projected using a common projection format such as the equirectangular or cubemap projection formats, in addition to the applications that have commonly been addressed by prior video coding standards.

Profiles, tiers, and levels

This document is designed to be versatile in the sense that it serves a wide range of applications, bit rates, resolutions, qualities, and services. Applications include, but are not limited to, video coding for digital storage media, television broadcasting, video streaming services, real-time communication. In the course of creating this document, various requirements from typical applications have been considered, necessary algorithmic elements have been developed, and these have been integrated into a single syntax. Hence, this document is designed to facilitate video data interchange among different applications.

Considering the practicality of implementing the full syntax of this document, however, a limited number of subsets of the syntax are also stipulated by means of "profiles", "tiers", and "levels". These and other related terms are formally defined in Clause 3.

A "profile" is a subset of the entire bitstream syntax that is specified in this document. Within the bounds imposed by the syntax of a given profile it is still possible to require a very large variation in the performance of encoders and decoders depending upon the values taken by syntax elements in the bitstream such as the specified size of the decoded pictures. In many applications, it is currently neither practical nor economical to implement a decoder capable of dealing with all hypothetical uses of the syntax within a particular profile.

In order to deal with this problem, "tiers" and "levels" are specified within each profile. A level of a tier is a specified set of constraints imposed on values of the syntax elements in the bitstream. Some of these constraints are expressed as simple limits on values, while others take the form of constraints on arithmetic combinations of values (e.g. picture width multiplied by picture height multiplied by number of pictures decoded per second). A level specified for a lower tier is more constrained than a level specified for a higher tier.

Coded video content conforming to this document uses a common syntax. In order to achieve a subset of the complete syntax, flags, parameters, and other syntax elements are included in the bitstream that signal the presence or absence of syntactic elements that occur later in the bitstream.

Encoding process, decoding process, and use of VUI parameters and SEI messages

Any encoding process that produces bitstream data that conforms to the specified bitstream syntax format requirements of this document is considered to be in conformance with the requirements of this document. The decoding process is specified such that all decoders that conform to a specified combination of capabilities known as the profile, tier, and level will produce numerically identical cropped decoded output pictures when invoking the decoding process associated with that profile for a bitstream conforming to that profile, tier and level. Any decoding process that produces identical cropped decoded output pictures to those produced by the process described herein (with the correct output order or output timing, as specified) is considered to be in conformance with the requirements of this document.

Rec. ITU-T H.274 | ISO/IEC 23002-7 specifies the syntax and semantics of the video usability information (VUI) parameters and supplemental enhancement information (SEI) messages that do not affect the conformance specifications in Annex C. These VUI parameters and SEI messages may be used together with this document.

Versions of this document

This is the first edition of this document.

Overview of the design characteristics

The coded representation specified in the syntax is designed to enable a high compression capability for a desired image or video quality. The algorithm is typically not mathematically lossless, as the exact source sample values are typically not preserved through the encoding and decoding processes, although some modes are included that provide lossless coding capability. A number of techniques are specified to enable highly efficient compression. Encoding algorithms (not specified within the scope of this document) may select between inter, intra, intra block copy (IBC), and palette coding for block-shaped regions of each picture. Inter coding uses motion vectors for block-based inter-picture prediction to exploit temporal statistical dependencies between different pictures, intra coding uses various spatial prediction modes to exploit spatial statistical dependencies in the source signal within the same picture, and intra block copy coding uses block displacement vectors to reference previously decoded regions of the same picture to exploit statistical similarities among different areas of the same picture. Motion vectors, intra prediction modes, and IBC block vectors are specified for a variety of block sizes in the picture. The prediction residual can then be further compressed using a spatial transform to remove spatial correlation inside a block before it is quantized, producing a possibly irreversible process that typically discards less important visual information while forming a close approximation to the source samples. Finally, the motion vectors, intra prediction modes, and block vectors can also be further compressed using a variety of prediction mechanisms, and, after prediction, are combined with the quantized transform coefficient information and encoded using arithmetic coding.

How to read this document

It is suggested that the reader starts with Clause 1 and moves on to Clause 3. Clause 6 should be read for the geometrical relationship of the source, input, and output of the decoder. Clause 7 specifies the order to parse syntax elements from the bitstream. See subclauses 7.1 to 7.3 for syntactical order and subclause 7.4 for semantics; e.g. the scope, restrictions, and conditions that are imposed on the syntax elements. The actual parsing for most syntax elements is specified in Clause 9. Finally, Clause 8 specifies how the syntax elements are mapped into decoded samples. Throughout reading this document, the reader should refer to Clauses 2, 4, and 5 as needed. Annexes A through D also form an integral part of this document.

Annex A specifies profiles, each being tailored to certain application domains, and defines the so-called tiers and levels of the profiles. Annex B specifies syntax and semantics of a byte stream format for delivery of coded video as an ordered stream of bytes. Annex C specifies the hypothetical reference decoder, bitstream conformance, decoder conformance, and the use of the hypothetical reference decoder to check bitstream and decoder conformance. Annex D specifies syntax and semantics for supplemental enhancement information (SEI) message payloads that affect the conformance specifications in Annex C. Rec. ITU-T H.274 | ISO/IEC 23002-7 specifies the syntax and semantics of the video usability information (VUI) parameters as well as SEI messages that do not affect the conformance specifications in Annex C. These VUI parameters and SEI messages may be used together with this document.

Patent declarations

The International Organization for Standardization (ISO) and International Electrotechnical Commission (IEC) draw attention to the fact that it is claimed that compliance with this document may involve the use of patents.

ISO and IEC take no position concerning the evidence, validity and scope of these patent rights.

The holders of these patent rights have assured ISO and IEC that they are willing to negotiate licences under reasonable and non-discriminatory terms and conditions with applicants throughout the world. In this respect, the statements of the holders of these patent rights are registered with ISO and IEC. Information may be obtained from the patent database available at www.iso.org/patents.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights other than those in the patent database. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

Information technology — Coded representation of immersive media — Part 3: Versatile video coding

1 Scope

This document specifies a video coding technology known as versatile video coding (VVC), comprising a video coding technology with a compression capability that is substantially beyond that of the prior generations of such standards and with sufficient versatility for effective use in a broad range of applications.

Only the syntax format, semantics, and associated decoding process requirements are specified, while other matters such as pre-processing, the encoding process, system signalling and multiplexing, data loss recovery, post-processing, and video display are considered to be outside the scope of this document. Additionally, the internal processing steps performed within a decoder are also considered to be outside the scope of this document; only the externally observable output behaviour is required to conform to the specifications of this document.

This document is designed to be generic in the sense that it serves a wide range of applications, bit rates, resolutions, qualities and services. Applications include, but are not limited to, video coding for digital storage media, television broadcasting and real-time communication. In the course of creating This document, various requirements from typical applications have been considered, necessary algorithmic elements have been developed, and these have been integrated into a single syntax. Hence, this document is designed to facilitate video data interchange among different applications.

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

Rec. ITU-T H.274 | ISO/IEC 23002-7, *Versatile supplemental enhancement information messages for coded video bitstreams*

Rec. ITU-T T.35:2000, Procedure for the allocation of ITU-T defined codes for non standard facilities.

3 Terms and definitions

For the purposes of this document, the following terms and definitions apply.

ISO and IEC maintain terminological databases for use in standardization at the following addresses:

— ISO Online browsing platform: available at https://www.iso.org/obp

— IEC Electropedia: available at http://www.electropedia.org/

3.1
access unit
set of *PUs* that belong to different *layers* and contain *coded pictures* associated with the same time for output from the *DPB*

3.2
adaptive colour transform
cross-*component transform* applied to the decoded *residual* of a *coding unit* in the 4:4:4 colour format prior to reconstruction and loop filtering

3.3
adaptive loop filter
filtering process that is applied as part of the *decoding process* and is controlled by parameters conveyed in an *APS*

3.4
AC transform coefficient
transform coefficient for which the *frequency index* in at least one of the two dimensions is non-zero

3.5
ALF APS
APS that controls the *ALF* process

3.6
adaptation parameter set
syntax structure containing *syntax elements* that apply to zero or more *slices* as determined by zero or more *syntax elements* found in *slice headers*

3.7
associated GDR picture
previous *GDR picture* (when present) in *decoding order*, for a particular picture with nuh_layer_id equal to a particular value layerId, that has nuh_layer_id equal to layerId and between which and the particular *picture* in *decoding order* there is no *IRAP picture* with nuh_layer_id equal to layerId

3.8
associated GDR subpicture
previous *GDR subpicture* (when present) in *decoding order*, for a particular subpicture with nuh_layer_id equal to a particular value layerId and subpicture index equal to a particular value subpicIdx, that has nuh_layer_id equal to layerId and subpicture index equal to subpicIdx and between which and the particular sub*picture* in *decoding order* there is no *IRAP subpicture* with nuh_layer_id equal to layerId and subpicture index equal to subpicIdx

3.9
associated IRAP picture
previous *IRAP picture* (when present) in *decoding order*, for a particular picture with nuh_layer_id equal to a particular value layerId, that has nuh_layer_id equal to layerId and between which and the particular *picture* in *decoding order* there is no *GDR picture* with nuh_layer_id equal to layerId

3.10
associated IRAP subpicture
previous *IRAP subpicture* (when present) in *decoding order*, for a particular subpicture with nuh_layer_id equal to a particular value layerId and subpicture index equal to a particular value subpicIdx, that has nuh_layer_id equal to layerId and subpicture index equal to subpicIdx and between which and the particular sub*picture* in *decoding order* there is no *GDR subpicture* with nuh_layer_id equal to layerId and subpicture index equal to subpicIdx

3.11
associated non-VCL NAL unit
non-VCL NAL unit (when present) for a *VCL NAL unit* where the *VCL NAL unit* is the *associated VCL NAL unit* of the *non-VCL NAL unit*

3.12
associated VCL NAL unit
preceding *VCL NAL unit* in *decoding order* for a *non-VCL NAL unit* with nal_unit_type equal to EOS_NUT, EOB_NUT, SUFFIX_APS_NUT, SUFFIX_SEI_NUT, FD_NUT, RSV_NVCL_27, UNSPEC_30, or UNSPEC_31; or otherwise the next *VCL NAL unit* in *decoding order*

3.13
bin
bit of a *bin string*

3.14
binarization
set of *bin strings* for all possible values of a *syntax element*

3.15
binarization process
unique mapping process of all possible values of a *syntax element* onto a set of *bin strings*

3.16
binary split
split of a rectangular MxN *block* of samples into two *blocks* where a vertical split results in a first (M / 2)xN *block* and a second (M / 2)xN *block*, and a horizontal split results in a first Mx(N / 2) *block* and a second Mx(N / 2) *block*

3.17
bin string
intermediate binary representation of values of *syntax elements* from the *binarization* of the *syntax element*

3.18
bi-predictive slice
B slice
slice that is decoded using *intra prediction* or using *inter prediction* with at most two *motion vectors* and *reference indices* to *predict* the sample values of each *block*

3.19
bitstream
sequence of bits, in the form of a *NAL unit stream* or a *byte stream*, that forms the representation of a sequence of *AUs* forming one or more coded video sequences *(CVSs)*

3.20
block
MxN (M-column by N-row) array of samples, or an MxN array of *transform coefficients*

3.21
block vector
two-dimensional vector that provides an offset from the coordinates of the current *coding block* to the coordinates of the reference block in the same decoded *slice*

3.22
byte
sequence of 8 bits, within which, when written or read as a sequence of bit values, the left-most and right-most bits represent the most and least significant bits, respectively

3.23
byte-aligned
<*bitstream*> positioned an integer multiple of 8 bits from the position of the first bit in the *bitstream*

3.24
byte-aligned
<bit or *byte* or *syntax element*> position at which it appears in a *bitstream* is byte-aligned

3.25
byte stream
encapsulation of a *NAL unit stream* into a series of *bytes* containing *start code prefixes* and *NAL units*

3.26
chroma
sample array or single sample representing one of the two colour difference signals related to the primary colours, represented by the symbols Cb and Cr

Note 1 to entry: The term chroma is used rather than the term chrominance in order to avoid the implication of the use of linear light transfer characteristics that is often associated with the term chrominance.

3.27
CRA PU
PU in which the *coded picture* is a *CRA picture*

3.28
CRA picture
IRAP picture for which each *VCL NAL unit* has nal_unit_type equal to CRA_NUT

Note 1 to entry: A CRA picture does not use inter prediction in its decoding process, and could be the first picture in the bitstream in decoding order, or could appear later in the bitstream. A CRA picture could have associated RADL or RASL pictures. When a CRA picture has NoOutputBeforeRecoveryFlag equal to 1, the associated RASL pictures are not output by the decoder, because they might not be decodable, as they could contain references to pictures that are not present in the bitstream.

3.29
CRA subpicture
IRAP subpicture for which each *VCL NAL unit* has nal_unit_type equal to CRA_NUT

3.30
coded layer video sequence:
sequence of *PUs* with the same value of nuh_layer_id that consists, in *decoding order*, of a *CLVSS PU*, followed by zero or more *PUs* that are not *CLVSS PUs*, including all subsequent *PUs* up to but not including any subsequent *PU* that is a *CLVSS PU*

Note 1 to entry: A CLVSS PU could be an IDR PU, a CRA PU, or a GDR PU. The value of NoOutputBeforeRecoveryFlag is equal to 1 for each IDR PU, and each CRA PU that has HandleCraAsClvsStartFlag equal to 1, and each CRA or GDR PU that is the first PU in

the layer of the bitstream in decoding order or the first PU in the layer of the bitstream that follows an EOS NAL unit in the layer in decoding order.

3.31
CLVSS PU
PU in which the *coded picture* is a *CLVSS picture*

3.32
CLVSS picture
coded picture that is an *IRAP picture* with NoOutputBeforeRecoveryFlag equal to 1 or a *GDR picture* with NoOutputBeforeRecoveryFlag equal to 1

3.33
coded picture
coded representation of a *picture* comprising *VCL NAL units* with a particular value of nuh_layer_id within an *AU* and containing all *CTUs* of the *picture*

3.34
coded picture buffer
first-in first-out buffer containing *DUs* in *decoding order* specified in the *hypothetical reference decoder*

Note 1 to entry: The hypothetical reference decoder is specified in Annex C.

3.35
coded representation
data element as represented in its coded form

3.36
coded video sequence
sequence of *AUs* that consists, in *decoding order*, of a *CVSS AU*, followed by zero or more *AUs* that are not *CVSS AUs*, including all subsequent *AUs* up to but not including any subsequent *AU* that is a *CVSS AU*

3.37
CVSS AU
IRAP AU or *GDR AU* for which the *coded picture* in each *PU* is a *CLVSS picture*

3.38
coding block
MxN *block* of samples for some values of M and N such that the division of a *CTB* into *coding blocks* is a *partitioning*

3.39
coding tree block
N×N *block* of samples for some value of N such that the division of a *component* into *CTBs* is a *partitioning*

3.40
coding tree unit
CTB of *luma* samples, two corresponding *CTBs* of *chroma* samples of a *picture* that has three sample arrays, or a *CTB* of samples of a monochrome *picture*, and *syntax structures* used to code the samples

3.41
coding unit
coding block of *luma* samples, two corresponding *coding blocks* of *chroma* samples of a *picture* that has three sample arrays in the single tree mode, or a *coding block* of *luma* samples of a *picture* that has three sample arrays in the dual tree mode, or two *coding blocks* of *chroma* samples of a *picture* that has three sample arrays in the dual tree mode, or a *coding block* of samples of a monochrome *picture*, and *syntax structures* used to code the samples

3.42
component
array or single sample from one of the three arrays (*luma* and two *chroma*) that compose a *picture* in 4:2:0, 4:2:2, or 4:4:4 colour format or the array or a single sample of the array that compose a *picture* in monochrome format

3.43
context variable
variable specified for the *adaptive binary arithmetic decoding process* of a *bin* by an equation containing recently decoded *bins*

3.44
deblocking filter
filtering process that is applied as part of the *decoding process* in order to minimize the appearance of visual artefacts at the boundaries between *blocks*

3.45
decoded picture
picture produced by applying the *decoding process* to a *coded picture*

3.46
decoded picture buffer
buffer holding *decoded pictures* for reference, output reordering, or output delay specified for the *hypothetical reference decoder*

3.47
decoder
embodiment of a *decoding process*

3.48
decoding order
order in which *syntax elements* are processed by the *decoding process*

3.49
decoding process
process specified in this document that reads a *bitstream* and derives *decoded pictures* from it

3.50
decoding unit
AU if DecodingUnitHrdFlag is equal to 0 or a subset of an *AU* otherwise, consisting of one or more *VCL NAL units* in an *AU* and the *associated non-VCL NAL units*

3.51
emulation prevention byte
byte equal to 0x03 that is present within a *NAL unit* when the *syntax elements* of the *bitstream* form certain patterns of *byte* values in a manner that ensures that no sequence of consecutive *byte-aligned bytes* in the *NAL unit* can contain a *start code prefix*

3.52
encoder
embodiment of an *encoding process*

3.53
encoding process
process not specified in this document that produces a *bitstream* conforming to this document

3.54
filler data NAL units
NAL units with nal_unit_type equal to FD_NUT

3.55
flag
variable or single-bit *syntax element* that can take one of the two possible values: 0 and 1

3.56
frequency index
one-dimensional or two-dimensional index associated with a *transform coefficient* prior to the application of a *transform* in the *decoding process*

3.57
GDR AU
AU in which there is a *PU* for each *layer* present in the *CVS* and the *coded picture* in each present *PU* is a *GDR picture*

3.58
GDR PU
PU in which the *coded picture* is a *GDR picture*

3.59
GDR picture
picture for which each VCL NAL unit has nal_unit_type equal to GDR_NUT

Note 1 to entry: The value of pps_mixed_nalu_types_in_pic_flag for a GDR picture is equal to 0. When pps_mixed_nalu_types_in_pic_flag is equal to 0 for a picture, and any slice of the picture has nal_unit_type equal to GDR_NUT, all other slices of the picture have the same value of nal_unit_type, and the picture is known to be a GDR picture after receiving the first slice.

3.60
GDR subpicture
subpicture for which each VCL NAL unit has nal_unit_type equal to GDR_NUT

3.61
hypothetical reference decoder
hypothetical *decoder* model that specifies constraints on the variability of conforming *NAL unit streams* or conforming *byte streams* that an encoding process may produce

3.62
hypothetical stream scheduler
hypothetical delivery mechanism used for checking the conformance of a *bitstream* or a *decoder* with regards to the timing and data flow of the input of a *bitstream* into the *hypothetical reference decoder*

3.63
IDR PU
PU in which the *coded picture* is an *IDR picture*

3.64
IDR picture
IRAP picture for which each *VCL NAL unit* has nal_unit_type equal to IDR_W_RADL or IDR_N_LP

Note 1 to entry: An IDR picture does not use inter prediction in its decoding process, and could be the first picture in the bitstream in decoding order, or could appear later in the bitstream. Each IDR picture is the first picture of a CVS in decoding order. When an IDR picture for which each VCL NAL unit has nal_unit_type equal to IDR_W_RADL, it could have associated RADL pictures. When an IDR picture for which each VCL NAL unit has nal_unit_type equal to IDR_N_LP, it does not have any associated leading pictures. An IDR picture does not have associated RASL pictures.

3.65
IDR subpicture
IRAP subpicture for which each *VCL NAL unit* has nal_unit_type equal to IDR_W_RADL or IDR_N_LP

3.66
inter-layer reference picture
picture in the same *AU* with the current *picture*, with nuh_layer_id less than the nuh_layer_id of the current *picture*, and is marked as "used for long-term reference"

3.67
inter coding
coding of a *coding block*, *slice*, or *picture* that uses *inter prediction*

3.68
inter prediction
prediction derived from blocks of sample values of one or more *reference pictures* as determined by motion vectors

3.69
intra block copy prediction
prediction derived from blocks of sample values of the same decoded *slice* as determined by block vectors

3.70
intra coding
coding of a *coding block, slice*, or *picture* that uses *intra prediction*

3.71
intra prediction
prediction derived from neighbouring sample values of the same decoded *slice*

3.72
IRAP AU
AU in which there is a *PU* for each layer present in the CVS and the *coded picture* in each *PU* is an *IRAP picture*

3.73
IRAP PU
PU in which the *coded picture* is an *IRAP picture*

3.74
IRAP picture
coded picture for which all *VCL NAL units* have the same value of nal_unit_type in the range of IDR_W_RADL to CRA_NUT, inclusive

Note 1 to entry: An IRAP picture could be a CRA picture or an IDR picture. An IRAP picture does not use inter prediction from reference pictures in the same layer in its decoding process. The first picture in the bitstream in decoding order is an IRAP or GDR picture. For a single-layer bitstream, provided the necessary parameter sets are available when they need to be referenced, the IRAP picture and all subsequent non-RASL pictures in the CLVS in decoding order are correctly decodable without performing the decoding process of any pictures that precede the IRAP picture in decoding order.

Note 2 to entry: The value of pps_mixed_nalu_types_in_pic_flag for an IRAP picture is equal to 0. When pps_mixed_nalu_types_in_pic_flag is equal to 0 for a picture, and any slice of the picture has nal_unit_type in the range of IDR_W_RADL to CRA_NUT, inclusive, all other slices of the picture have the same value of nal_unit_type, and the picture is known to be an IRAP picture after receiving the first slice.

3.75
IRAP subpicture
subpicture for which all *VCL NAL units* have the same value of nal_unit_type in the range of IDR_W_RADL to CRA_NUT, inclusive

3.76
intra slice
I slice
slice that is decoded using *intra prediction* only

3.77
layer
set of *VCL NAL units* that all have a particular value of nuh_layer_id and the *associated non-VCL NAL units*

3.78
leading picture
picture that precedes the *associated IRAP picture* in *output order*

3.79
leading subpicture
subpicture that precedes the *associated IRAP subpicture* in *output order*

3.80
leaf
terminating node of a tree that is a root node of a tree of depth 0

3.81
level <constraints>
defined set of constraints on the values that may be taken by the *syntax elements* and variables of this document

Note 1 to entry: The same set of levels is defined for all profiles, with most aspects of the definition of each level being in common across different profiles. Individual implementations could, within the specified constraints, support a different level for each supported profile.

3.82
level <transform coefficient value>
value of a *transform coefficient* prior to *scaling*

3.83
list 0 (list 1) motion vector
motion vector associated with a *reference index* pointing into *reference picture list 0* (*list 1*)

3.84
list 0 (list 1) prediction
inter prediction of the content of a *slice* using a *reference index* pointing into *reference picture list 0* (*list 1*)

3.85
LMCS APS
APS that controls the *LMCS* process

3.86
long-term reference picture
picture with nuh_layer_id equal to the nuh_layer_id of the current *picture* and marked as "used for long-term reference"

3.87
luma
sample array or single sample representing the monochrome signal related to the primary colours, represented by the symbol or subscript Y or L