INTERNATIONAL STANDARD

ISO/IEC 23008-2

Third edition
2017-10-15

AMENDMENT 3
2018-07

Information technology — High efficiency coding and media delivery in heterogeneous environments —

Part 2: High efficiency video coding

AMENDMENT 3: Additional supplemental enhancement information

Technologies de l'information — Codage à haute efficacité et livraison des medias dans des environnements hétérogènes —

Partie 2: Codage vidéo à haute efficacité

AMENDEMENT 3: Informations additionnelles supplémentaires pour amélioration

Reference number
ISO/IEC 23008-2:2017/Amd.3:2018(E)

ISO copyright office
CP 401 • Ch. de Blandonnet 8
CH-1214 Vernier, Geneva
Phone: +41 22 749 01 11
Fax: +41 22 749 09 47
Email: copyright@iso.org
Website: www.iso.org

Published in Switzerland

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. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

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).

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

For an explanation on 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 the following URL: www.iso.org/iso/foreword.html.

This document was prepared by 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 identical text for ISO/IEC 23008-2 is published as ITU-T H.265.

A list of all parts in the ISO/IEC 23008 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.

Information technology — High efficiency coding and media delivery in heterogeneous environments —

Part 2:
High efficiency video coding

AMENDMENT 3: Additional supplemental enhancement information

General

Throughout the specification, replace all single-word instances of "nested" with "scalable-nested" (e.g., there is no single-word instance of "nested" in "non-nested"), and all instances of "non-nested" with "non-scalable-nested", except in 3.158 (which is kept unchanged) and Table F.4 in F.14.3.1 (for which a change is provided below).

Clause 3

Add the following term definitions (in alphabetical order within the current list) and correct all term numbering sequentially:

3.X
azimuth circle
circle on a sphere connecting all points with the same azimuth value

Note 1 to entry: An azimuth circle is always a *great circle* like a longitude line on the earth.

3.X
constituent picture
part of a spatially frame-packed stereoscopic video picture that corresponds to one view, or a picture itself when frame packing is not in use or the temporal interleaving frame packing arrangement is in use

3.X
elevation circle
circle on a sphere connecting all points with the same elevation value

Note 1 to entry: An elevation circle is similar to a lattitude line on the earth. Except when the elevation value is zero, an elevation circle is not a *great circle* like a longitude circle on the earth.

3.X
global coordinate axes
coordinate axes associated with *omnidirectional video* that are associated with an externally reference-able position and orientation

Note 1 to entry: The global coordinate axes may correspond to the position and orientation of a device or rig used for omnidirectional audio/video acquisition as well as the position of an observer's head in the three-dimensional space of the *omnidirectional video* rendering environment.

3.X
great circle
intersection of a sphere and a plane that passes through the centre point of the sphere

Note 1 to entry: A great circle is also known as an orthodrome or Riemannian circle.

3.X
local coordinate axes
coordinate axes having a specified rotation relationship relative to the *global coordinate axes*

3.X
omnidirectional video
video content in a format that enables rendering according to the user's viewing orientation, e.g., if viewed using a head-mounted device, or according to a user's desired *viewport*, reflecting a potentially rotated viewing position

3.X
packed region
region in a *region-wise packed picture* that is mapped to a *projected region* according to a *region-wise packing*

3.X
projected picture
picture that uses a *projection* format for *omnidirectional video*

3.X
projected region
region in a *projected picture* that is mapped to a *packed region* according to a *region-wise packing*

3.X
projection
specified correspondence between the colour samples of a *projected picture* and azimuth and elevation positions on a sphere

3.X
region-wise packed picture
decoded picture that contains one or more *packed regions*

Note 1 to entry: A packed picture may contain a *region-wise packing* of a *projected picture*.

3.X
region-wise packing
transformation, resizing, and relocation of *packed regions* of a *region-wise packed picture* to remap the *packed regions* to *projected regions* of a *projected picture*

3.X
sphere coordinates
azimuth and elevation angles identifying a location of a point on a sphere

3.X
sphere region
region on a sphere, specified either by four *great circles* or by two *azimuth circles* and two *elevation circles*, or such a region on a rotated sphere after applying yaw, pitch, and roll rotations

3.X
tilt angle
angle indicating the amount of tilt of a *sphere region*, measured as the amount of rotation of a *sphere region* along the axis originating from the sphere origin passing through the centre point of the *sphere region*, where the angle value increases clockwise when looking from the origin towards the positive end of the axis

3.X
viewport
region of *omnidirectional video* content suitable for display and viewing by the user

Clause 4

Add the following to the list of abbreviations (in alphabetical order):

MCTS motion-constrained tile set

5.8

Add the following function definitions:

Asin(x) the trigonometric inverse sine function, operating on an argument x that is in the range of −1.0 to 1.0, inclusive, with an output value in the range of −π ÷ 2 to π ÷ 2, inclusive, in units of radians (5-2)

Atan(x) the trigonometric inverse tangent function, operating on an argument x, with an output value in the range of −π ÷ 2 to π ÷ 2, inclusive, in units of radians (5-3)

$$\text{Atan2}(y,x)=\begin{cases}\text{Atan}\left(\frac{y}{x}\right) & ;\ \text{if } x > 0 \\ \text{Atan}\left(\frac{y}{x}\right)+\pi & ;\ \text{if } x < 0 \ \&\ \&\ y >= 0 \\ \text{Atan}\left(\frac{y}{x}\right)-\pi & ;\ \text{if } x < 0 \ \&\ \&\ y < 0 \\ +\frac{\pi}{2} & ;\ \text{if } x == 0 \ \&\ \&\ y >= 0 \\ -\frac{\pi}{2} & ;\ \text{otherwise}\end{cases} \quad (5\text{-}4)$$

Renumber the prior Formulae 5-2 through 5-15 as 5-5 to 5-18 to account for the added formulae.

Add the following function definition:

Sin(x) the trigonometric sine function operating on an argument x in units of radians (5-19)

Renumber the prior Formulae 5-19 through 5-20 as 5-20 to 5-21 to account for the added formulae.

Add the following function definition:

Tan(x) the trigonometric tangent function operating on an argument x in units of radians (5-22)

7.4.2.4.4, NOTE 2

Delete the sentence that says “Consequently, hypothetical reference decoder (HRD) parameters carried in non-nested buffering period, picture timing and decoding unit information SEI messages apply to access units based on such access unit boundary detection.”

7.4.4

Replace paragraph 6 (directly after NOTE 1) and NOTE 2 with the following:

general_non_packed_constraint_flag equal to 1 specifies that there are no frame packing arrangement SEI messages, segmented rectangular frame packing arrangement SEI messages, omnidirectional projection indication SEI messages, or cubemap projection SEI messages present in the CVS. general_non_packed_constraint_flag equal to 0 indicates that there may or may not be one or more frame packing arrangement SEI messages, segmented rectangular frame packing arrangement

SEI messages, omnidirectional projection indication SEI messages, or cubemap projection SEI messages present in the CVS.

NOTE 2 Decoders could ignore the value of general_non_packed_constraint_flag, as there are no decoding process requirements associated with the presence or interpretation of frame packing arrangement SEI messages, segmented rectangular frame packing arrangement SEI messages, equirectangular projection SEI messages, or cubemap projection SEI messages present in the CVS.

8.7.2.1

Replace paragraph 4 (directly after the NOTE) with the following:

The deblocking filter process is applied to all prediction block edges and transform block edges of a picture, except the following types of edges:

— Edges that are at the boundary of the picture;

— Edges that coincide with tile boundaries when loop_filter_across_tiles_enabled_flag is equal to 0;

— Edges that coincide with upper or left boundaries of slices with slice_loop_filter_across_slices_enabled_flag, equal to 0 or slice_deblocking_filter_disabled_flag equal to 1;

— Edges within slices with slice_deblocking_filter_disabled_flag equal to 1;

— Edges that do not correspond to 8 × 8 sample grid boundaries of the considered component;

— Edges within chroma components for which both sides of the edge use inter prediction;

— Edges of chroma transform blocks that are not edges of the associated transform unit.

C.1

Add the following NOTE 1 immediately before the sentence that says “Figure C.1 shows the types of bitstream conformance points checked by the HRD.”, and renumber the existing NOTEs in the clause accordingly:

NOTE 1 Decoders conforming to profiles specified in Annex A do not use NAL units with nuh_layer_id greater than 0 (e.g., access unit delimiter NAL units with nuh_layer_id greater than 0) for access unit boundary detection, except for identification of whether a NAL unit is a VCL or non-VCL NAL unit. Consequently, hypothetical reference decoder (HRD) parameters carried in non-scalable-nested buffering period, picture timing and decoding unit information SEI messages apply to access units that are identified based on such access unit boundary detection.

D.2.1

Replace this subclause with the following:

D.2.1 General SEI message syntax

sei_payload(payloadType, payloadSize) {	**Descriptor**
if(nal_unit_type = = PREFIX_SEI_NUT)	
if(payloadType = = 0)	
buffering_period(payloadSize)	
else if(payloadType = = 1)	
pic_timing(payloadSize)	
else if(payloadType = = 2)	
pan_scan_rect(payloadSize)	
else if(payloadType = = 3)	
filler_payload(payloadSize)	
else if(payloadType = = 4)	

user_data_registered_itu_t_t35(payloadSize)	
else if(payloadType = = 5)	
user_data_unregistered(payloadSize)	
else if(payloadType = = 6)	
recovery_point(payloadSize)	
else if(payloadType = = 9)	
scene_info(payloadSize)	
else if(payloadType = = 15)	
picture_snapshot(payloadSize)	
else if(payloadType = = 16)	
progressive_refinement_segment_start(payloadSize)	
else if(payloadType = = 17)	
progressive_refinement_segment_end(payloadSize)	
else if(payloadType = = 19)	
film_grain_characteristics(payloadSize)	
else if(payloadType = = 22)	
post_filter_hint(payloadSize)	
else if(payloadType = = 23)	
tone_mapping_info(payloadSize)	
else if(payloadType = = 45)	
frame_packing_arrangement(payloadSize)	
else if(payloadType = = 47)	
display_orientation(payloadSize)	
else if(payloadType = = 56)	
green_metadata(payloadsize) /* specified in ISO/IEC 23001-11 */	
else if(payloadType = = 128)	
structure_of_pictures_info(payloadSize)	
else if(payloadType = = 129)	
active_parameter_sets(payloadSize)	
else if(payloadType = = 130)	
decoding_unit_info(payloadSize)	
else if(payloadType = = 131)	
temporal_sub_layer_zero_index(payloadSize)	
else if(payloadType = = 133)	
scalable_nesting(payloadSize)	
else if(payloadType = = 134)	
region_refresh_info(payloadSize)	
else if(payloadType = = 135)	
no_display(payloadSize)	
else if(payloadType = = 136)	
time_code(payloadSize)	
else if(payloadType = = 137)	
mastering_display_colour_volume(payloadSize)	
else if(payloadType = = 138)	

segmented_rect_frame_packing_arrangement(payloadSize)	
else if(payloadType = = 139)	
temporal_motion_constrained_tile_sets(payloadSize)	
else if(payloadType = = 140)	
chroma_resampling_filter_hint(payloadSize)	
else if(payloadType = = 141)	
knee_function_info(payloadSize)	
else if(payloadType = = 142)	
colour_remapping_info(payloadSize)	
else if(payloadType = = 143)	
deinterlaced_field_identification(payloadSize)	
else if(payloadType = = 144)	
content_light_level_info(payloadSize)	
else if(payloadType = = 145)	
dependent_rap_indication(payloadSize)	
else if(payloadType = = 146)	
coded_region_completion(payloadSize)	
else if(payloadType = = 147)	
alternative_transfer_characteristics(payloadSize)	
else if(payloadType = = 148)	
ambient_viewing_environment(payloadSize)	
else if(payloadType = = 149)	
content_colour_volume(payloadSize)	
else if(payloadType = = 150)	
equirectangular_projection(payloadSize)	
else if(payloadType = = 151)	
cubemap_projection(payloadSize)	
else if(payloadType = = 154)	
sphere_rotation(payloadSize)	
else if(payloadType = = 155)	
regionwise_packing(payloadSize)	
else if(payloadType = = 156)	
omni_viewport(payloadSize)	
else if(payloadType = = 157)	
regional_nesting(payloadSize)	
else if(payloadType = = 158)	
mcts_extraction_info_sets(payloadSize)	
else if(payloadType = = 159)	
mcts_extraction_info_nesting(payloadSize)	
else if(payloadType = = 160)	
layers_not_present(payloadSize) /* specified in Annex F */	
else if(payloadType = = 161)	
inter_layer_constrained_tile_sets(payloadSize) /* specified in Annex F */	
else if(payloadType = = 162)	

bsp_nesting(payloadSize) /* specified in Annex F */	
else if(payloadType = = 163)	
bsp_initial_arrival_time(payloadSize) /* specified in Annex F */	
else if(payloadType = = 164)	
sub_bitstream_property(payloadSize) /* specified in Annex F */	
else if(payloadType = = 165)	
alpha_channel_info(payloadSize) /* specified in Annex F */	
else if(payloadType = = 166)	
overlay_info(payloadSize) /* specified in Annex F */	
else if(payloadType = = 167)	
temporal_mv_prediction_constraints(payloadSize) /* specified in Annex F */	
else if(payloadType = = 168)	
frame_field_info(payloadSize) /* specified in Annex F */	
else if(payloadType = = 176)	
three_dimensional_reference_displays_info(payloadSize) /* specified in Annex G */	
else if(payloadType = = 177)	
depth_representation_info(payloadSize) /* specified in Annex G */	
else if(payloadType = = 178)	
multiview_scene_info(payloadSize) /* specified in Annex G */	
else if(payloadType = = 179)	
multiview_acquisition_info(payloadSize) /* specified in Annex G */	
else if(payloadType = = 180)	
multiview_view_position(payloadSize) /* specified in Annex G */	
else if(payloadType = = 181)	
alternative_depth_info(payloadSize) /* specified in Annex I */	
else	
reserved_sei_message(payloadSize)	
else /* nal_unit_type = = SUFFIX_SEI_NUT */	
if(payloadType = = 3)	
filler_payload(payloadSize)	
else if(payloadType = = 4)	
user_data_registered_itu_t_t35(payloadSize)	
else if(payloadType = = 5)	
user_data_unregistered(payloadSize)	
else if(payloadType = = 17)	
progressive_refinement_segment_end(payloadSize)	
else if(payloadType = = 22)	
post_filter_hint(payloadSize)	
else if(payloadType = = 132)	
decoded_picture_hash(payloadSize)	
else if(payloadType = = 146)	
coded_region_completion(payloadSize)	
else	
reserved_sei_message(payloadSize)	

if(more_data_in_payload( )) {	
if(payload_extension_present( ))	
reserved_payload_extension_data	u(v)
payload_bit_equal_to_one /* equal to 1 */	f(1)
while(!byte_aligned( ))	
payload_bit_equal_to_zero /* equal to 0 */	f(1)
}	
}	

D.2.40

Renumber D.2.40 (Reserved SEI message syntax) as D.2.45.

Add new subclauses D.2.40 through D.2.44, as follows:

D.2.40 Content colour volume SEI message syntax

content_colour_volume(payloadSize) {	Descriptor
ccv_cancel_flag	u(1)
if(!ccv_cancel_flag) {	
ccv_persistence_flag	u(1)
ccv_primaries_present_flag	u(1)
ccv_min_luminance_value_present_flag	u(1)
ccv_max_luminance_value_present_flag	u(1)
ccv_avg_luminance_value_present_flag	u(1)
ccv_reserved_zero_2bits	u(2)
if(ccv_primaries_present_flag)	
for(c = 0; c < 3; c++) {	
ccv_primaries_x[c]	i(32)
ccv_primaries_y[c]	i(32)
}	
if(ccv_min_luminance_value_present_flag)	
ccv_min_luminance_value	u(32)
if(ccv_max_luminance_value_present_flag)	
ccv_max_luminance_value	u(32)
if(ccv_avg_luminance_value_present_flag)	
ccv_avg_luminance_value	u(32)
}	
}	

D.2.41 Syntax of omnidirectional video specific SEI messages

D.2.41.1 Equirectangular projection SEI message syntax

equirectangular_projection(payloadSize) {	Descriptor
erp_cancel_flag	u(1)
if(!erp_cancel_flag) {	
erp_persistence_flag	u(1)
erp_guard_band_flag	u(1)
erp_reserved_zero_2bits	u(2)
if(erp_guard_band_flag = = 1) {	
erp_guard_band_type	u(3)
erp_left_guard_band_width	u(8)
erp_right_guard_band_width	u(8)
}	
}	
}	

D.2.41.2 Cubemap projection SEI message syntax

cubemap_projection(payloadSize) {	Descriptor
cmp_cancel_flag	u(1)
if(!cmp_cancel_flag)	
cmp_persistence_flag	u(1)
}	

D.2.41.3 Sphere rotation SEI message syntax

sphere_rotation(payloadSize) {	Descriptor
sphere_rotation_cancel_flag	u(1)
if(!sphere_rotation_cancel_flag) {	
sphere_rotation_persistence_flag	u(1)
sphere_rotation_reserved_zero_6bits	u(6)
yaw_rotation	i(32)
pitch_rotation	i(32)
roll_rotation	i(32)
}	
}	

D.2.41.4 Region-wise packing SEI message syntax

regionwise_packing(payloadSize) {	Descriptor
rwp_cancel_flag	u(1)
if(!rwp_cancel_flag) {	
rwp_persistence_flag	u(1)
constituent_picture_matching_flag	u(1)
rwp_reserved_zero_5bits	u(5)
num_packed_regions	u(8)
proj_picture_width	u(32)
proj_picture_height	u(32)
packed_picture_width	u(16)
packed_picture_height	u(16)
for(i = 0; i < num_packed_regions; i++) {	
rwp_reserved_zero_4bits[i]	u(4)
rwp_transform_type[i]	u(3)
rwp_guard_band_flag[i]	u(1)
proj_region_width[i]	u(32)
proj_region_height[i]	u(32)
proj_region_top[i]	u(32)
proj_region_left[i]	u(32)
packed_region_width[i]	u(16)
packed_region_height[i]	u(16)
packed_region_top[i]	u(16)
packed_region_left[i]	u(16)
if(rwp_guard_band_flag[i]) {	

rwp_left_guard_band_width[i]	u(8)
rwp_right_guard_band_width[i]	u(8)
rwp_top_guard_band_height[i]	u(8)
rwp_bottom_guard_band_height[i]	u(8)
rwp_guard_band_not_used_for_pred_flag[i]	u(1)
for(j = 0; j < 4; j++)	
rwp_guard_band_type[i][j]	u(3)
rwp_guard_band_reserved_zero_3bits[i]	u(3)
}	
}	
}	
}	

D.2.41.5 Omnidirectional viewport SEI message syntax

omni_viewport(payloadSize) {	**Descriptor**
omni_viewport_id	u(10)
omni_viewport_cancel_flag	u(1)
if(!omni_viewport_cancel_flag) {	
omni_viewport_persistence_flag	u(1)
omni_viewport_cnt_minus1	u(4)
for(i = 0; i <= omni_viewport_cnt_minus1; i++) {	
omni_viewport_azimuth_centre[i]	i(32)
omni_viewport_elevation_centre[i]	i(32)
omni_viewport_tilt_centre[i]	i(32)
omni_viewport_hor_range[i]	u(32)
omni_viewport_ver_range[i]	u(32)
}	
}	
}	

D.2.42 Regional nesting SEI message syntax

regional_nesting(payloadSize) {	**Descriptor**
regional_nesting_id	u(16)
regional_nesting_num_rect_regions	u(8)
for(i = 0; i < regional_nesting_num_rect_regions; i++) {	
regional_nesting_rect_region_id[i]	u(8)
regional_nesting_rect_left_offset[i]	u(16)
regional_nesting_rect_right_offset[i]	u(16)
regional_nesting_rect_top_offset[i]	u(16)
regional_nesting_rect_bottom_offset[i]	u(16)
}	
num_sei_messages_in_regional_nesting_minus1	u(8)
for(i = 0; i <= num_sei_messages_in_regional_nesting_minus1; i++) {	
num_regions_for_sei_message[i]	u(8)