

JCTVC-J0183:

Syntax Issues

Kazushi Sato, Sony Corp.
Kazushi.Sato@jp.sony.com

Agenda

- Introduction
- max_temporal_layers_minus1 and temporal_id_nesting_flag
- log2_min_transform_block_size_minus2, diff_cu_qp_delta_depth and transform_skip_enable_flag
- Conclusion

max_temporal_layers_minus1 and temporal_id_nesting_flag [1/]

- Both VPS (Video Parameter Set) and SPS (Sequence Parameter Set) contain syntax elements “max_temporal_layers_minus1” and “temporal_id_nesting_flag”
- If the value of temporal_id_nesting_flag is 1, the constraint as below will be introduced:
 - *For each access unit auA with temporal_id equal to tldA, an access unit auB with temporal_id equal to tldB that is less than or equal to tldA shall not be referenced by inter prediction when there exists an access unit auC with temporal_id equal to tldC that is less than tldB, which follows the access unit auB and precedes the access unit auA in decoding order.*
- if the value of max_temporal_layers_minus1 is 0, it means that there is only one temporal layer, and in this case temporal_id_nesting_flag becomes meaningless.

max_temporal_layers_minus1 and temporal_id_nesting_flag [2/]

- Solution:
 - It is proposed to transmit temporal_id_nesting_flag only if the value of max_temporal_layers_minus1 is not equal to 0 both in VPS and SPS.

▪ video_parameter_set_rbsp() { [↵]	Descriptor [↵]	↵
▪ vps_max_temporal_layers_minus1 [↵]	u(3) [↵]	↵
▪ vps_max_layers_minus1 [↵]	u(5) [↵]	↵
▪ video_parameter_set_id [↵]	ue(v) [↵]	↵
▪ if(vps_max_temporal_layers_minus1 != 0) [↵]	[↵]	↵
▪ vps_temporal_id_nesting_flag [↵]	u(1) [↵]	↵
▪ for(i = 0; i <= vps_max_temporal_layers_minus1; i++) { [↵]	[↵]	↵
▪ vps_max_dec_pic_buffering[i] [↵]	ue(v) [↵]	↵
▪ vps_num_reorder_pics[i] [↵]	ue(v) [↵]	↵
▪ vps_max_latency_increase[i] [↵]	ue(v) [↵]	↵
▪ } [↵]	[↵]	↵
▪ vps_extension_flag [↵]	u(1) [↵]	↵
▪ if(vps_extension_flag) [↵]	[↵]	↵
▪ while(more_rbsp_data()) [↵]	[↵]	↵
▪ vps_extension_data_flag [↵]	u(1) [↵]	↵
▪ } [↵]	[↵]	↵
▪ rbsp_trailing_bits() [↵]	[↵]	↵
▪ } [↵]	[↵]	↵

seq_parameter_set_rbsp() {	Descriptor
...	u(3)
sps_max_temporal_layers_minus1	u(3)
pic_width_in_luma_samples	ue(v)
pic_height_in_luma_samples	ue(v)
pic_cropping_flag	u(1)
if(pic_cropping_flag) {	
pic_crop_left_offset	ue(v)
pic_crop_right_offset	ue(v)
pic_crop_top_offset	ue(v)
pic_crop_bottom_offset	ue(v)
}	
bit_depth_luma_minus8	ue(v)
bit_depth_chroma_minus8	ue(v)
...	u(1)
if(pcm_enabled_flag)	
pcm_loop_filter_disable_flag	u(1)
if(sps_max_temporal_layers_minus1 != 0)	
sps_temporal_id_nesting_flag	u(1)
...	
}	

log2_min_transform_block_size_minus
2 and diff_cu_qp_delta_depth [1/]

- Minimum TU size in a bitstream is specified by log2_min_transform_block_size_minus2 in SPS.
- There are 2 redundancies associated w/ this syntax element

log2_min_transform_block_size_minus
 2 and diff_cu_qp_delta_depth [2/]

- Redunancy -1
 - For example, minimum CU size specified by diff_cu_qp_delta_depth is 8x8, and minimum TU size specified by log2_min_transform_block_size_minus2 is 16x16.
 - In this case cu_qp_delta is never transmitted in the unit of 8x8, and it is redundant.
 - Therefore the value of diff_cu_qp_delta_depth should be restricted by the value of log2_min_transform_block_size_minus2.
 - log2_min_transform_block_size_minus2 should be transmitted in SPS but slice_granularity and diff_cu_qp_delta_depth is transmitted in PPS
 - **Usually slice_granularity and diff_cu_qp_delta_depth remains same within a sequence?**

`log2_min_transform_block_size_minus2` and `diff_cu_qp_delta_depth` [3/]

- Redundancy -2
 - If the value of `transform_skip_enable_flag` in SPS is 1, transform skip can be applied for intra 4x4 TUs with Y/Cb/Cr component.
 - However, if the value of `log2_min_transform_block_size_minus2` larger than 0, it means that there is no 4x4 TUs so it is not necessary to transmit `transform_skip_enable_flag`.

$\log_2_min_transform_block_size_minus2$ and $diff_cu_qp_delta_depth$ [4/]

- Solution

- minimum CU size for transmission of cu_qp_delta , specified by $diff_cu_qp_delta_depth$, is restricted by minimum TU size specified by $\log_2_min_transform_block_size_minus2$
 - **$slice_granularity$ and $diff_cu_qp_delta_depth$ is transmitted in SPS and can optionally be transmitted in PPS**
 - **$\log_2_min_transform_block_size_minus2$ can optionally be transmitted in PPS**
- $transform_skip_enable_flag$ is transmitted only if $\log_2_min_transform_block_size_minus2=0$

seq_parameter_set_rbsp() {,	Descriptor,
...	u(3),
log2_min_coding_block_size_minus3,	ue(v),
log2_diff_max_min_coding_block_size,	ue(v),
log2_min_transform_block_size_minus2,	ue(v),
log2_diff_max_min_transform_block_size,	ue(v),
if(pcm_enabled_flag) {,	,
log2_min_pcm_coding_block_size_minus3,	ue(v),
log2_diff_max_min_pcm_coding_block_size,	ue(v),
},	,
max_transform_hierarchy_depth_inter,	ue(v),
max_transform_hierarchy_depth_intra,	ue(v),
sps_slice_granularity,	u(2),
sps_diff_cu_qp_delta_depth,	ue(v),
scaling_list_enable_flag,	u(1),
if(scaling_list_enable_flag) {,	,
sps_scaling_list_data_present_flag,	u(1),
if(sps_scaling_list_data_present_flag),	,
scaling_list_param(),	,
},	,
chroma_pred_from_luma_enabled_flag,	u(1),
if(log2_min_transform_block_size_minus2==0),	,
transform_skip_enabled_flag,	u(1),
seq_loop_filter_across_slices_enabled_flag,	u(1),
asymmetric_motion_partitions_enabled_flag,	u(1),
...	,
},	,

pic_parameter_set_rbsp() {	Descriptor
... ⁴	ue(v)
constrained_intra_pred_flag	u(1)
pps_log2_min_transform_block_size_minus2_present_flag	u(1)
if(pps_log2_min_transform_block_size_minus2_present_flag)	,
pps_log2_min_transform_block_size_minus2	ue(v)
pps_slice_granularity_present_flag	u(1)
if(pps_slice_granularity_present_flag)	,
pps_slice_granularity	u(2)
if(pps_log2_min_transform_block_size_minus2_present_flag && pps_slice_granularity_present_flag)	,
pps_diff_cu_qp_delta_depth_flag	u(1)
else	,
pps_diff_cu_qp_delta_depth_flag = 0	,
if(pps_diff_cu_qp_delta_depth_flag)	,
pps_diff_cu_qp_delta_depth	ue(v)
cb_qp_offset	se(v)
cr_qp_offset	se(v)
weighted_pred_flag	u(1)
weighted_bipred_idc	u(2)
.... ⁵	,
}	,

Conclusion

- This contribution proposes 3 changes in syntax to remove redundancies as follow:
 - **temporal_id_nesting_flag shall not be transmitted if the value of max_temporal_layers_minus1 is equal to 0**
 - **minimum CU size for cu_qp_transmission specified by diff_cu_qp_delta_depth is restricted by the minimum transform size specified by log2_min_transform_block_size_minus2**
 - **transform_skip_enable_flag is transmitted only if log2_min_transform_block_size_minus2=0**
- It is recommended that these changes be adopted into HEVC DIS text.



"SONY" or "make.believe" is a registered trademark and/or trademark of Sony Corporation.

Names of Sony products and services are the registered trademarks and/or trademarks of Sony Corporation or its Group companies.

Other company names and product names are the registered trademarks and/or trademarks of the respective companies.