JCTVC-S0094:
QP OFFSET FOR ADAPTIVE COLOUR TRANSFORM

Rickard Sjöberg
Jonatan Samuelsson

QP OFFSET PROPOSAL

› Adaptive Color Transform (ACT) was adopted at the 18th JCT-VC meeting and can be used by an encoder to transform GBR residual data to YCoCg

	GBR	YCoCg
L sample array	Green	Luma
Cb sample array	Blue	Chroma Green
Cr sample array	Red	Chroma Orange

› Currently, chroma QP offsets is applied on sample arrays
  - E.g. The QP offset is shared between Blue in GBR and and Chroma Green in YCoCg

› We propose to add separate QP offsets for YCoCg residual data to increase flexibility
  - Similar to proposals in JCTVC-S0040 and JCTVC-S0144

SYNTAX CHANGES

[In an extension to the PPS:]

pps_scc_extension( ) {	Descriptor	Comment
pps_transformed_y_qp_offset	se(v)	Offset for Luma in YCoCg
pps_transformed_cb_qp_offset	se(v)	Offset for Chroma Green in YCoCg
pps_transformed_cr_qp_offset	se(v)	Offset for Chroma Orange in YCoCg
pps_slice_transformed_qp_offsets_present_flag	u(1)	Gating flag for slice syntax
}		

[In the slice segment header:]

slice_segment_header( ) {	Descriptor	Comment
…		
slice_qp_delta	se(v)	
if(pps_slice_chroma_qp_offsets_present_flag) {		
slice_cb_qp_offset	se(v)	Offset for Blue in GBR
slice_cr_qp_offset	se(v)	Offset for Red in GBR
}		
if(pps_slice_transformed_qp_offsets_present_flag) {		
slice_transformed_y_qp_offset	se(v)	Offset for Luma in YCoCg
slice_transformed_cb_qp_offset	se(v)	Offset for Chroma Green in YCoCg
slice_transformed_cr_qp_offset	se(v)	Offset for Chroma Orange in YCoCg
}		
…		

QP RANGE PROBLEM

› In the current SCC draft text, the quantization parameter qP can become negative when ACT is enabled due to the (-5, -5, -3) QP offsets used for normalizing the transform
  – This problem is also identified in other contributions to this meeting, e.g. JCTVC-S0086, JCTVC-S0140, and JCTVC-S0144

› If QP offsets for ACT is adopted, we need to ensure that qP parameters are within the allowed range, both negative and too large values must be prohibited/disallowed

QP RANGE PROPOSAL

[In the derivation process for quantization parameters (Section 8.6.1):]

When ChromaArrayType is not equal to 0, the following applies.

– If cu_residual_act_flag[xTbY][yTbY] is equal to 0, the variables qP_{Cb} and qP_{Cr} are derived as follows:

$\qquad qPi_{Cb} = Clip3(-QpBdOffset_C, 57, Qp_Y + pps\_cb\_qp\_offset + slice\_cb\_qp\_offset + CuQpOffset_{Cb})$

$\qquad qPi_{Cr} = Clip3(-QpBdOffset_C, 57, Qp_Y + pps\_cr\_qp\_offset + slice\_cr\_qp\_offset + CuQpOffset_{Cr})$

– Otherwise (cu_residual_act_flag[xTbY][yTbY] is equal to 1), the variables qP_{Cb} and qP_{Cr} are derived as follows:

$\qquad qPi_{Cb} = Clip3(-QpBdOffset_C, 57, Qp_Y + pps\_transformed\_cb\_qp\_offset + slice\_transformed\_cb\_qp\_offset\ -5 + CuQpOffset_{Cb})$

$\qquad qPi_{Cr} = Clip3(-QpBdOffset_C, 57, Qp_Y + pps\_transformed\_cr\_qp\_offset + slice\_transformed\_cr\_qp\_offset\ -3 + CuQpOffset_{Cr})$

[In the scaling and transformation process (Section 8.6.2)]

The quantization parameter qP is derived as follows:

– If cIdx is equal to 0,

$\qquad qP = Clip3(0, 51 + QpBdOffset_Y, Qp'_Y + (cu\_residual\_act\_flag\ [\ xTbY\ ][\ yTbY\ ]\ ?$
$\qquad pps\_transformed\_y\_qp\_offset + slice\_transformed\_y\_qp\_offset\ -5 : 0)$
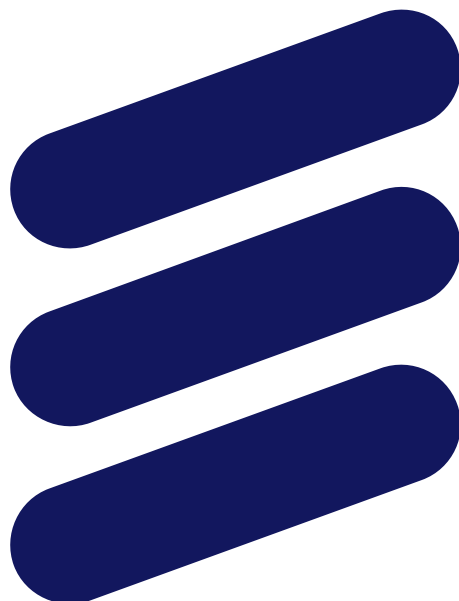
– Otherwise, if cIdx is equal to 1,

$\qquad qP = Qp'_{Cb} + \sout{(cu\_residual\_act\_flag\ [\ xTbY\ ][\ yTbY\ ]\ ?\ -5 : 0)}$

– Otherwise (cIdx is equal to 2),

$\qquad qP = Qp'_{Cr} + \sout{(cu\_residual\_act\_flag\ [\ xTbY\ ][\ yTbY\ ]\ ?\ -3 : 0)}$

> Adding Chroma Green and Chroma Orange QP offset and transform normalization offsets without new clipping

> Adding new clipping for Luma
>
> (since Q_{PY} is used for deblocking and QP prediction we can not use existing clipping as above)