



MEDIATEK

JVET-Q0182

AHG9: Allowing slice-level scaling list and LMCS

Authors: Chen-Yen Lai, Tzu-Der Chuang, Olena Chubach, Ching-Yeh Chen, Yu-Wen Huang, Shaw-Min Lei



Presenter: Chen-Yen Lai

Overall Summary

- The current design of Scaling List (SL)/LMCS lacks flexibility.
 - No slice-level SL/LMCS: cannot efficiently encode one picture including different scenes.
 - All slices in one picture shall be coded by the same SL/LMCS settings: difficult to merge bitstreams with different SL/LMCS settings.
- Propose to allow different slices within one coded picture refer to different SL/LMCS APSs.

Method 1: Signal SL/LMCS Syntax Elements in either PH or SH

- In picture header, referencing the syntax design for ALF

if(sps_lmcs_enabled_flag) {	
pic_lmcs_enabled_present_flag 	Add LMCS present flag in PH
if (pic_lmcs_enabled_present_flag) {	
pic_lmcs_enabled_flag	u(1)
if(pic_lmcs_enabled_flag) {	
pic_lmcs_aps_id	u(2)
if(ChromaArrayType != 0)	
pic_chroma_residual_scale_flag	u(1)
}	
}	
}	
if(sps_scaling_list_enabled_flag) {	
pic_scaling_list_enabled_present_flag 	Add SL present flag in PH
if (pic_scaling_list_enabled_present_flag) {	
pic_scaling_list_present_flag	u(1)
if(pic_scaling_list_present_flag)	
pic_scaling_list_aps_id	u(3)
}	
}	

Method 1: Signal SL/LMCS Syntax Elements in either PH or SH

- In slice header,

When the LMCS syntax elements **are not** present in PH

```
if( sps_lmcs_enabled_flag && ! pic_lmcs_enabled_present_flag ) {  
    slice_lmcs_enabled_flag  
    if(slice_lmcs_enabled_flag) {  
        slice_lmcs_aps_id  
        if( ChromaArrayType != 0 )  
            slice_chroma_residual_scale_flag  
    }  
}  
  
if( sps_scaling_list_enabled_flag && ! pic_scaling_list_enabled_present_flag ) {  
    slice_scaling_list_present_flag  
    if(slice_scaling_list_present_flag) {  
        slice_scaling_list_aps_id  
    }  
}
```

When the SL syntax elements **are not** present in PH

Not efficient if we want to share some settings between some slices and allow different settings for other slices.

Method 2: Allow Overriding PH SL/LMCS in SH

- In slice header,

if(sps_lmcs_enabled_flag) {	
slice_lmcs_override_flag	→ Add a LMCS override flag:
if(slice_lmcs_override_flag) {	1: Signal LMCS syntax elements in slice header
slice_lmcs_enabled_flag	0: Follow LMCS settings in PH
if(slice_lmcs_enabled_flag) {	
slice_lmcs_aps_id	
if(ChromaArrayType != 0)	
slice_chroma_residual_scale_flag	
}	
}	
}	
if(sps_scaling_list_enabled_flag) {	
slice_scaling_list_override_flag	→ Add a SL override flag:
if(slice_scaling_list_override_flag) {	1: Signal SL syntax elements in slice header
slice_scaling_list_present_flag	0: Follow SL setting in PH
if(slice_scaling_list_present_flag)	
slice_scaling_list_aps_id	
}	
}	

Comparison

Method	# of allowed reference SL/LMCS APs in one picture	Syntax signalled in PH	Syntax signalled in SH
VVC Draft 7	One	V	X
Method 1	More than one	V (the syntax elements either in PH or SH)	
Method 2	More than one	Same as VVC Draft 7	V (allow overriding PH)

Method 2: Can achieve **sharing the same settings** between some slices and **allowing different setting** for other slices at the same time.

Conclusion

- It is suggested to allow different slices within one coded picture to refer to different SL/LMCS APSs.
- Two proposed methods:
 - Signal SL/LMCS syntax elements in either PH or SH
 - Allow overriding PH SL/LMCS in SH



Thank You