



MEDIATEK

JVET-P0149

AHG18: Disabling LMCS for lossless coding

Authors: Zhi-Yi Lin, Tzu-Der Chuang, Ching-Yeh Chen, Chih-Wei Hsu, Yu-Wen Huang, Shaw-Min Lei

Presenter: Zhi-Yi Lin

Overall Summary

- Two methods are proposed to achieve lossless coding with LMCS enabled at sequence level
- Method 1: disable LMCS **at slice level**
 - Do not signal slice_lmcs_enabled_flag and infer it to 0 if pps_transquant_bypass_enable_flag is true
- Method 2: disable LMCS **at CU level**
 - Disable forward mapping, inverse mapping, and chroma scaling in blocks coded with lossless mode

Introduction

- Lossless coding cannot be achieved when LMCS is applied
 - With the forward and inverse luma mappings, the reconstructed samples cannot be perfectly restored
 - With the chroma residual scaling, the residuals of a lossless block are changed

Proposed Method 1

- Disable LMCS at slice level if the transquant bypass mode is enabled at picture level

slice_header() {	
...	
if(sps_lmcs_enabled_flag && ! pps_transquant_bypass_enabled_flag) {	
slice_lmcs_enabled_flag	u(1)
if(slice_lmcs_enabled_flag) {	
slice_lmcs_aps_id	u(2)
if(ChromaArrayType != 0)	
slice_chroma_residual_scale_flag	u(1)
}	
}	
...	

Proposed Method 2

- Disable LMCS at CU level if a CU is coded in lossless mode
 - The forward mapping for inter prediction is disabled
 - The inverse mapping for reconstructed samples before loop filtering is disabled
 - The chroma scaling is disabled
- For those blocks not coded with lossless mode, LMCS can still be applied

Conclusion

- To achieve lossless coding when LMCS is enabled in SPS, different methods are proposed to disable LMCS
- Method 1: Disable LMCS at slice level
 - Do not signal slice_lmcs_enabled_flag and infer it to 0 if pps_transquant_bypass_enable_flag is true
- Method 2: Disable LMCS at CU level
 - Disable forward mapping, inverse mapping, and chroma scaling in blocks coded with lossless mode



MEDIATEK

Thank you!