

JVET-N0477

On luma mapping with chroma scaling

Yan Ye, Jie Chen, Ru-Ling Liao (Alibaba)



Problem statement

- Luma mapping with chroma scaling (LMCS) was adopted into VVC draft 4
- Chroma residual scaling issues in VTM4:
 - Depends on the average value of the entire corresponding luma block
 - Invoked without checking chroma cbf
- Averaging the entire luma block is complex and unnecessary:
 - Complex: luma block can be up to 64x64
 - Unnecessary: small deviation in luma value often does not lead to change in piecewise linear index
- Invoking the scaling factor derivation is unnecessary if only zero chroma residual



Simplified luma averaging

- Method 1: Up to 4x4 luma samples are used to calculate the average
 - Let S = size in dimension X (X could be horizontal or vertical)
 - If $S \leq 4$, use all sample positions in X
 - Otherwise, use only 4 evenly-spaced sample positions in that dimension
 - **Benefit: reduce computational complexity**
- Method 2:
 - If intra coded block, use the top-left reference sample in intra prediction
 - Otherwise, use method 1
 - **Benefit: further reduced computation + reduced latency**



Simulation results (method 1)

	Random Access				
Class A1	-0.02%	0.00%	-0.03%	100%	101%
Class A2	0.00%	-0.02%	-0.04%	100%	101%
Class B	0.00%	-0.05%	-0.05%	100%	100%
Class C	0.01%	0.00%	0.06%	100%	100%
Class E					
Overall	0.00%	-0.02%	-0.01%	100%	100%
Class D	0.01%	-0.04%	-0.08%	100%	100%
Class F	0.00%	0.02%	-0.07%	100%	100%

	Low Delay B				
Class A1					
Class A2					
Class B	-0.01%	0.09%	-0.01%	100%	100%
Class C	-0.05%	0.24%	0.14%	100%	102%
Class E	-0.06%	0.31%	-0.28%	99%	101%
Overall	-0.04%	0.20%	-0.03%	100%	101%
Class D	0.01%	-0.03%	-0.33%	100%	99%
Class F	-0.04%	-0.09%	0.03%	100%	100%



Simulation results (method 2)

	Random Access				
Class A1	0.01%	-0.13%	0.05%	99%	99%
Class A2	-0.01%	-0.14%	-0.04%	99%	100%
Class B	0.01%	-0.05%	-0.05%	99%	98%
Class C	0.00%	0.05%	-0.05%	100%	100%
Class E					
Overall	0.00%	-0.06%	-0.03%	99%	99%
Class D	-0.03%	0.19%	0.05%	100%	97%
Class F	0.00%	0.03%	-0.02%	100%	99%

	Low Delay B				
Class A1					
Class A2					
Class B	0.00%	0.04%	-0.01%	99%	99%
Class C	-0.06%	0.42%	-0.04%	99%	95%
Class E	-0.03%	0.35%	-0.59%	99%	95%
Overall	-0.03%	0.24%	-0.16%	99%	97%
Class D	-0.03%	0.18%	-0.85%	100%	93%
Class F	-0.04%	-0.11%	0.10%	99%	97%



Simulation results: summary

		Y	U	V	EncT	DecT
Method 1	RA	0.00%	-0.02%	-0.01%	100%	100%
	LD	-0.04%	0.20%	-0.03%	100%	101%
Method 2	RA	0.00%	-0.06%	-0.03%	99%	99%
	LD	-0.03%	0.24%	-0.16%	99%	97%

Chroma cbf condition

- Chroma scaling process in VTM4:
 1. Calculate average of entire luma block
 2. Identify piecewise linear index, identify chroma scaling factor accordingly
 3. Scale chroma residual
- Proposal: condition steps 1 & 2 on tu_cb_cbf and tu_cr_cbf (step 3 already conditioned)
- No performance impact (bit exact results)



Conclusion

- Two methods to simplify the luma averaging process
 - Method 2 provides better complexity vs. performance tradeoff
- Additionally, condition the first two steps using chroma cbf's
- Suggest to adopt into VVC draft 5
 - Use top-left reference sample if intra coded
 - Condition using chroma cbf's

