

JVET-N0390 - AHG16/non-CE3: Study of CCLM restrictions in case of separate luma/chroma tree



Problem: in AhG16 on Implementation studies, concerns raised on CCLM in case of Separate luma/chroma partitioning tree

- Latency issue for processing the chroma block - in worst case, need to wait for reconstruction of the 64x64 collocated luma block before being able processing the chroma block

Goal of this proposal

- Investigating constraints to be applied to CCLM activation when separate luma/chroma tree is used

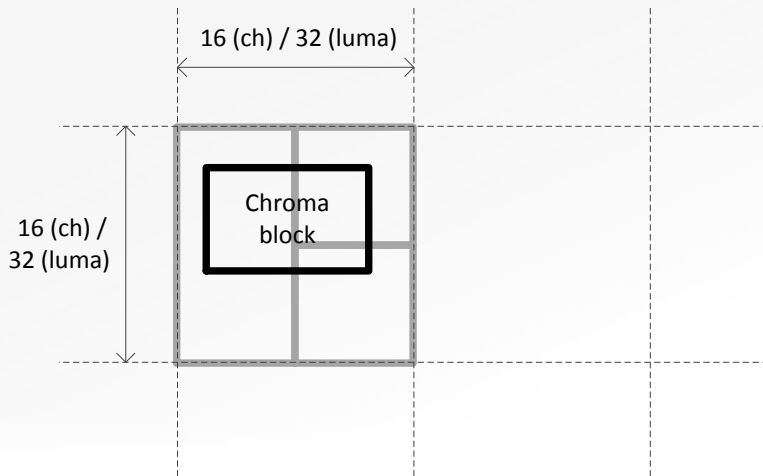
De-activation of CCLM or SepT has strong performance impact (cfg AHG13 report)

Abbreviation	AI				
	BDR-Y	BDR-U	BDR-V	EncTime	DecTime
CST	0.29%	11.28%	12.41%	121%	101%
CCLM	2.08%	19.05%	19.23%	99%	99%

Abbreviation	RA				
	BDR-Y	BDR-U	BDR-V	EncTime	DecTime
CST	0.16%	4.90%	6.35%	102%	99%
CCLM	0.91%	17.37%	17.31%	101%	102%

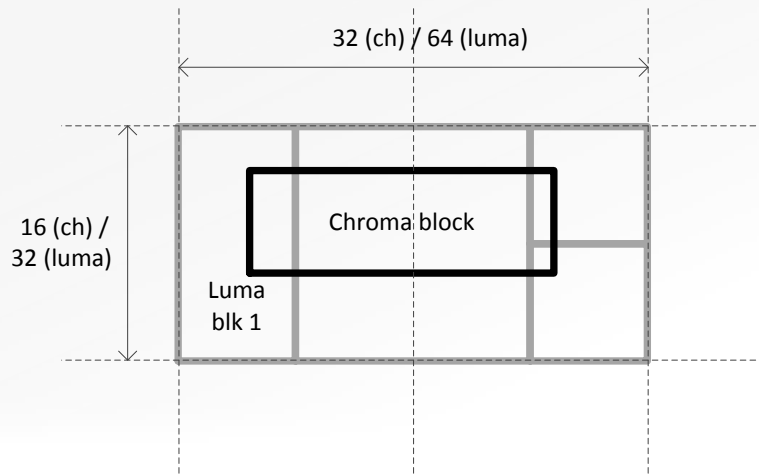
Restriction1 (“restrict16x16”): CCLM enables when

- Chroma block is inside 16x16 area
- Collocated luma blocks are inside collocated 32x32 area
- Max latency - processing of 32x32 luma samples



Restriction2 (“restrict16x32or32x16”): CCLM enables when

- Chroma block is inside 16x32 or 32x16 area
- Collocated luma blocks are inside the collocated 32x64 or 64x32 area
- Max latency - processing of 32x64 or 64x32 luma block



To compensate for loss due to the restrictions

- **enableBT** - enable BT split for 64x64 luma blks
- **forceQTorBT** - force QT or BT split for 64x64 luma blks

Test results - AI configuration - VTM4.0 as anchor (CCLM on)

no CCLM

Y	U	V	EncT	DecT
2.08%	19.05%	19.23%	99%	99%

enableBT

Y	U	V	EncT	DecT
-0.23%	-0.26%	-0.28%	111%	107%

forceQTorBT

Y	U	V	EncT	DecT
-0.06%	0.35%	0.33%	113%	107%

restrict16x16

Y	U	V	EncT	DecT
0.28%	2.57%	2.56%	99%	98%

enableBT_restrict16x16

Y	U	V	EncT	DecT
0.11%	2.68%	2.81%	110%	107%

forceQTorBT_restrict16x16

Y	U	V	EncT	DecT
0.21%	2.80%	2.91%	114%	106%

enableBT_restrict16x32or32x16

Y	U	V	EncT	DecT
0.04%	2.18%	2.18%	111%	107%

forceQTorBT_restrict16x32or32x16

Y	U	V	EncT	DecT
0.09%	1.98%	1.87%	115%	105%

Test results - AI configuration - VTM4.0 without CCLM as anchor

CCLM gain remains significant when applying the restrictions

no CCLM

Y	U	V	EncT	DecT
-1.98%	-14.46%	-15.05%	101%	101%

enableBT

Y	U	V	EncT	DecT
-2.20%	-14.66%	-15.28%	112%	108%

forceQTorBT

Y	U	V	EncT	DecT
-2.04%	-14.21%	-14.76%	114%	108%

restrict16x16

Y	U	V	EncT	DecT
-1.71%	-12.47%	-13.03%	100%	99%

enableBT_restrict16x16

Y	U	V	EncT	DecT
-1.88%	-12.39%	-12.85%	111%	108%

forceQTorBT_restrict16x16

Y	U	V	EncT	DecT
-1.78%	-12.26%	-12.70%	115%	108%

enableBT_restrict16x32or32x16

Y	U	V	EncT	DecT
-1.94%	-12.80%	-13.34%	112%	108%

forceQTorBT_restrict16x32or32x16

Y	U	V	EncT	DecT
-1.90%	-12.92%	-13.52%	116%	106%

reasonable impact of CCLM restrictions in RA

	Y	U	V	EncT	DecT
restrict16x16	0.05%	1.25%	1.04%	99%	100%
enableBT_restrict16x16	0.03%	1.25%	1.06%	101%	100%
forceQTorBT_restrict16x32or32x16	0.00%	0.82%	0.65%	101%	100%

Different options investigated to reduce the latency for processing chroma samples coded in CCLM, in case of separate luma/chroma tree

Preferred option (restrict16x16): enable CCLM if

- chroma block inside 16x16 area
- collocated luma blocks inside 32x32 collocated areas

versus VTM4.0 CCLM on

AI			RA		
Y	U	V	Y	U	V
0.28%	2.57%	2.56%	0.05%	1.25%	1.04%

AI - versus VTM4.0 CCLM off

no CCLM			restrict16x16		
Y	U	V	Y	U	V
-1.98%	-14.46%	-15.05%	-1.71%	-12.47%	-13.03%

Thanks to Huawei for cross-checking (JVET-N0792)