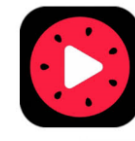


JVET-N0277

NON-CE4: PARALLEL-FRIENDLY PAIRWISE AVERAGE MERGE CANDIDATE DERIVATION

Kai Zhang, Li Zhang, Hongbin Liu, Jizheng Xu, Yue Wang
Bytedance Inc.



Problem

- In VTM-4.0, the pairwise average merge candidate is built as the average of the first two candidate in the merge candidate list.
- In the worst case, it cannot be built until the merge candidate list prior to it has been constructed, which is not friendly in the parallel-processing design.

Proposed methods

■ Method 1:

- *The pairwise average merge candidate is always built with motion information from the left and above neighbouring blocks.*
- *If only one set of motion information is available from these two, that set of motion information is copied to the pair-wise average candidate, with MVs right-shifted by 1*

■ Method 2:

- *The pair-wise average merge candidate is always built with motion information from the left and above neighbouring blocks, and TMVP.*
- *The first two available sets from these three are averaged to build the pair-wise average candidate.*
- *If only one set of motion information is available from these three, that motion information is copied to the pairwise average candidate, with MVs right-shifted by 1.*

Simulations results on VTM-4.0

■ Test 1:

	RA					LB				
	Y	U	V	EncT	DecT	Y	U	V	EncT	DecT
Class A1	0.02%	-0.01%	0.04%	100%	100%					
Class A2	0.02%	0.00%	-0.03%	100%	99%					
Class B	0.03%	-0.01%	0.00%	99%	99%	-0.01%	-0.14%	-0.40%	99%	99%
Class C	0.03%	0.01%	-0.09%	98%	95%	-0.04%	0.37%	-0.04%	101%	103%
Class E						0.03%	0.00%	-0.15%	101%	98%
Overall	0.02%	0.00%	-0.02%	99%	98%	-0.01%	0.07%	-0.22%	100%	100%
Class D	0.03%	0.02%	0.10%	98%	93%	-0.07%	0.13%	-0.66%	100%	99%
Class F	-0.02%	0.07%	-0.02%	98%	98%	-0.02%	-0.19%	0.55%	98%	97%

■ Test 2:

	RA					LB				
	Y	U	V	EncT	DecT	Y	U	V	EncT	DecT
Class A1	-0.02%	-0.06%	0.09%	100%	100%					
Class A2	-0.01%	0.02%	-0.03%	100%	99%					
Class B	0.02%	-0.10%	0.04%	99%	99%	-0.01%	-0.09%	0.10%	99%	100%
Class C	0.02%	0.01%	-0.10%	99%	98%	-0.02%	0.27%	-0.10%	100%	103%
Class E						0.00%	0.35%	0.17%	100%	100%
Overall	0.00%	-0.04%	0.00%	100%	99%	-0.01%	0.14%	0.05%	100%	101%
Class D	0.00%	-0.02%	0.06%	99%	100%	-0.03%	0.43%	-0.81%	99%	98%
Class F	0.00%	-0.01%	-0.02%	98%	101%	-0.11%	-0.21%	-0.12%	98%	98%

Conclusion

- Both the two solutions can make the pairwise average merge candidate derivation process more friendly to parallel processing while keeping the coding performance almost unchanged
- It is recommended to adopt method 1 into VVC.