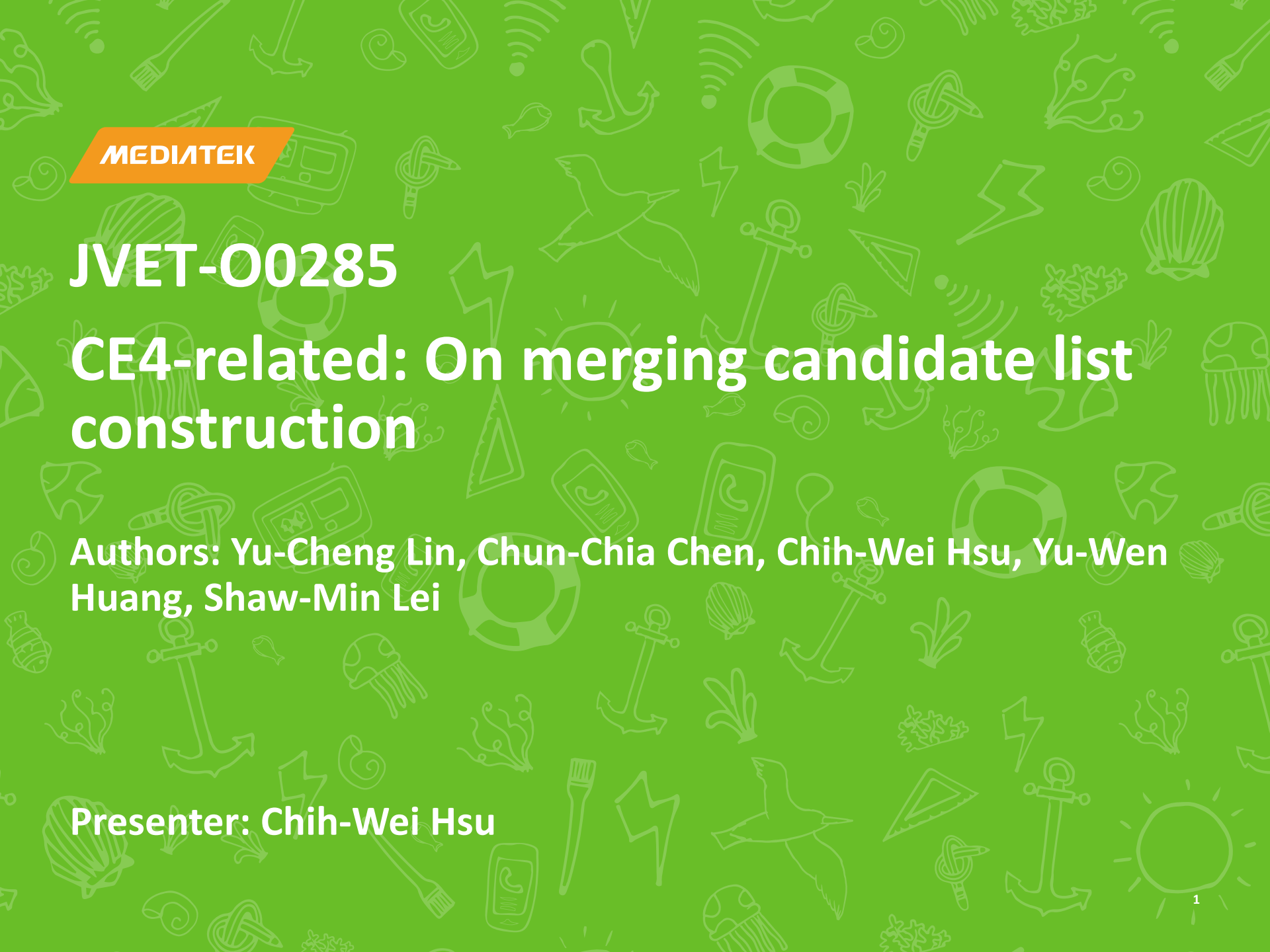




MEDIATEK

JVET-00285

CE4-related: On merging candidate list construction

Authors: Yu-Cheng Lin, Chun-Chia Chen, Chih-Wei Hsu, Yu-Wen Huang, Shaw-Min Lei

Presenter: Chih-Wei Hsu

Overall Summary

- In current VVC
 - MaxNumMergeCand variable is used in merging candidate list construction and HMVP table size
 - Decoder has to support different values of MaxNumMergeCand
- Proposed method
 - Replace the MaxNumMergeCand with a fixed value (equal to 6) for merging candidate list construction
 - Change the HMVP table size to constant (equal to 5)
- Simulation results

Over VTM 5.0 @ full length Anchor VTM5.0						
		Y	U	V	EncT	DecT
Proposed method	RA	0.00%	0.00%	0.00%	100%	99%
	LB	0.00%	0.00%	0.00%	100%	101%

Conclusions

- Proposed methods
 - Replace the MaxNumMergeCand with a fixed value (equal to 6)
 - Change the HMVP table size to constant (equal to 5)
- Decoder can construct the merging candidate list by a constant MaxNumMergeCand
 - Simply derive the whole list of merge candidates and select one pointed to by a given merge index
 - No need of different circuits for different MaxNumMergeCand
- No BD-rate change
- Thanks Huawei for cross-checking



everyday genius

Appendix – proposed WD change

Proposed WD Change (JVET-N1001 v8)

8.5.2.2 Derivation process for luma motion vectors for merge mode

...

6. The variable numCurrMergeCand and numOrigMergeCand are set equal to the number of merging candidates in the mergeCandList.↵
7. When numCurrMergeCand is less than ~~5(MaxNumMergeCand-1)~~ and NumHmvpCand is greater than 0, the following applies:↵
 - The derivation process of history-based merging candidates as specified in 8.5.2.6 is invoked with mergeCandList and numCurrMergeCand as inputs, and modified mergeCandList and numCurrMergeCand as outputs.↵
 - numOrigMergeCand is set equal to numCurrMergeCand.↵
8. When numCurrMergeCand is less than ~~6MaxNumMergeCand~~ and greater than 1, the following applies:↵
 - The derivation process for pairwise average merging candidate specified in clause 8.5.2.4 is invoked with mergeCandList, the reference indices refIdxL0N and refIdxL1N, the prediction list utilization flags predFlagL0N and predFlagL1N, the motion vectors mvL0N and mvL1N of every candidate N in mergeCandList, and numCurrMergeCand as inputs, and the output is assigned to mergeCandList, numCurrMergeCand, the reference indices refIdxL0avgCand and refIdxL1avgCand, the prediction list utilization flags predFlagL0avgCand and predFlagL1avgCand and the motion vectors mvL0avgCand and mvL1avgCand of candidate avgCand being added into mergeCandList. The bi-prediction weight index bcwIdx of candidate avgCand being added into mergeCandList is set equal to 0. ↵
 - numOrigMergeCand is set equal to numCurrMergeCand.↵

Proposed WD Change (JVET-N1001 v8)

8.5.2.5 Derivation process for zero motion vector merging candidates

... When numCurrMergeCand is less than **6MaxNumMergeCand**, the variable numInputMergeCand is set equal to numCurrMergeCand, the variable zeroIdx is set equal to 0 and the following ordered steps are repeated until numCurrMergeCand is equal to **6MaxNumMergeCand**:

1. For the derivation of the reference indices, the prediction list utilization flags and the motion vectors of the zero motion vector merging candidate, the following applies:
 - If slice_type is equal to P, the candidate zeroCand_m with m equal to (numCurrMergeCand – numInputMergeCand) is added at the end of mergeCandList, i.e., mergeCandList[numCurrMergeCand] is set equal to zeroCand_m, and the reference indices, the prediction list utilization flags and the motion vectors of zeroCand_m are derived as follows and numCurrMergeCand is incremented by 1:

$$\text{refIdxL0zeroCand}_m = (\text{zeroIdx} < \text{numRefIdx}) ? \text{zeroIdx} : 0 \quad (8-317)$$

$$\text{refIdxL1zeroCand}_m = -1 \quad (8-318)$$

$$\text{predFlagL0zeroCand}_m = 1 \quad (8-319)$$

$$\text{predFlagL1zeroCand}_m = 0 \quad (8-320)$$

$$\text{mvL0zeroCand}_m[0] = 0 \quad (8-321)$$

$$\text{mvL0zeroCand}_m[1] = 0 \quad (8-322)$$

$$\text{mvL1zeroCand}_m[0] = 0 \quad (8-323)$$

$$\text{mvL1zeroCand}_m[1] = 0 \quad (8-324)$$

$$\text{numCurrMergeCand} = \text{numCurrMergeCand} + 1 \quad (8-325)$$

Simulation result for MaxNumMergeCand = 2

Over VTM 5.0 @ cpq full length Anchor VTM5.0						
		Y	U	V	EncT	DecT
MaxNumMergeCand = 2	RA	1.05%	1.04%	1.11%	94%	97%
	LB	0.96%	1.06%	0.70%	91%	101%