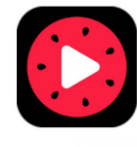


# JVET-P0400

## AhG16/Non-CE8: Removal of shared merge list

Y. Wang, L. Zhang, H. Liu, K. Zhang, J. Xu, Y. Wang  
ByteDance



# JVET-P0400: Background

- Motivation of shared merge list
  - *Improve the MV throughput of 4x4 blocks*
- Only one merge list is constructed for a shared merge region (SMR)

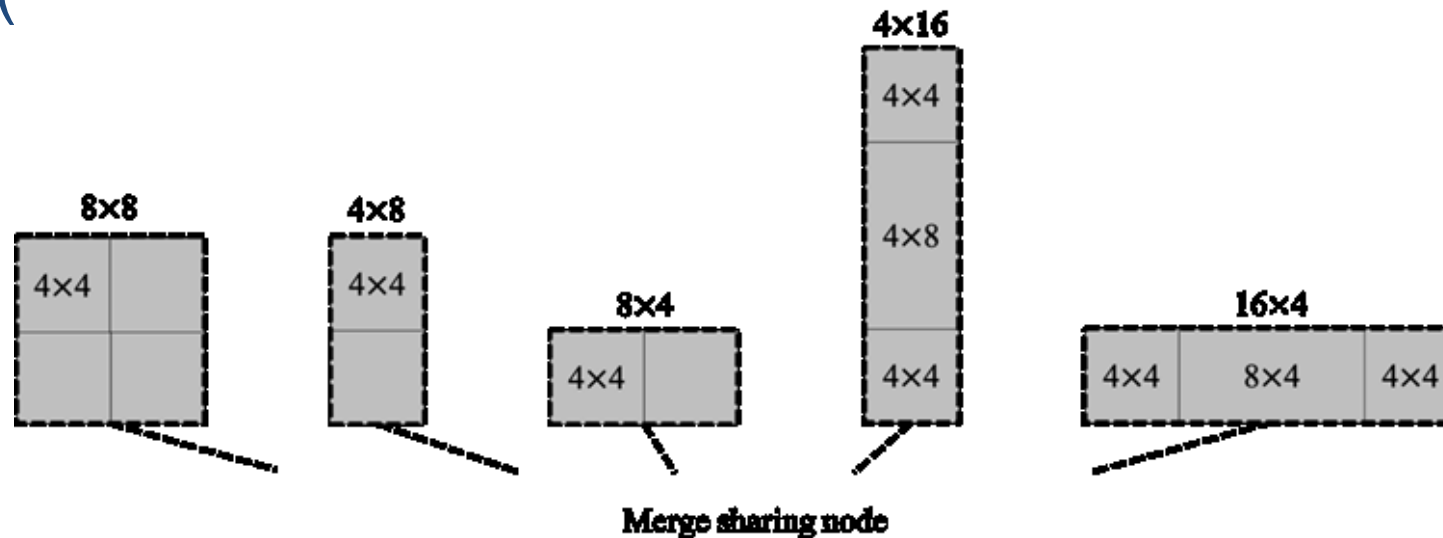


Figure 1. Examples of merge sharing nodes

# JVET-P0400: Problem statement

- One list per CU, as in current SW and Spec
  - *Latency of parsing*
    - The information (e.g. location, size) indicating whether current node is the merge sharing node must be saved.
  - *More complex logic and dependency on ancestor CUs*
    - All CUs must be checked whether it is one of the leaf CUs in a merge sharing node.
- One list per SMR
  - *Waste of computational complexity*
    - The BV candidate list is still constructed while all CUs under a merge sharing node are not coded with IBC mode.
    - For a small region containing one or more merge sharing nodes, BV candidate list may need to be constructed multiple times.

# JVET-P0400: Removal of shared merge list

## ■ Solution: No construction of BV list for 4x4 blocks

- *Shared merge list: removed*
- *Spatial merge candidates for 4x4 blocks: removed*
- *Updating of HMVP tables for 4x4 blocks: skipped*

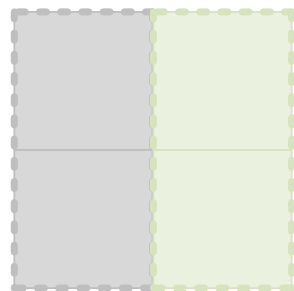
## ■ Simplest way to derive a BV candidate

- *Parse an index, get the BV from HMVP table*

# Complexity analysis

## ■ For a small region including multiple SMRs

| Region size                      | Methods  | Number of spatial neighbours to be accessed | Number of pruning | Number of BV list construction | Dependency on its neighbours |
|----------------------------------|----------|---|-------------------|--------------------------------|------------------------------|
| 8×8 in Figure 3(a)<br>worst case | VTM      | 2×2   | 3×2               | 1×2                            | Y                            |
|                                  | Proposed | 0   | 0                 | 0                              | N                            |
|                                  | Saving   | 100%  | 100%              | 100%                           |                              |
| 16×4 in Figure 3(b)              | VTM      | 2×2   | 3×2               | 1×2                            | Y                            |
|                                  | Proposed | 2   | 3                 | 1                              | N                            |
|                                  | Saving   | 50%   | 50%               | 50%                            |                              |



(a) 8×8 region with vertical BT and each 8×4 sub-region with horizontal BT



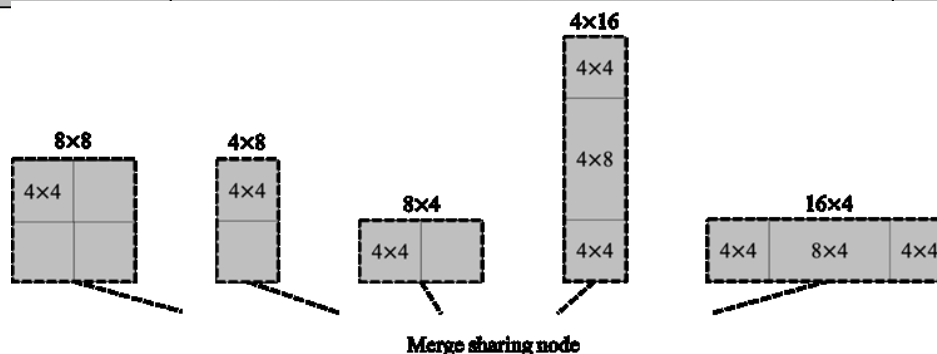
(b) 16×4 region with vertical BT and left 8×4 sub-region with vertical BT

Figure 2. 8×8/16×4 region with certain partitions

# Complexity analysis

## ■ For a SMR

| CU size, split type | Methods       | Number of spatial neighbours to be accessed | Number of pruning | Number of BV list construction |
|---------------------|---------------|---|-------------------|--------------------------------|
| 8×8, QT             | VTM           | 2   | 3                 | 1                              |
|                     | Proposed      | 0   | 0                 | 0                              |
|                     | <b>Saving</b> | 100%  | 100%              | 100%                           |
| 8×4/4×8, BT         | VTM           | 2   | 3                 | 1                              |
|                     | Proposed      | 0   | 0                 | 0                              |
|                     | <b>Saving</b> | 100%  | 100%              | 100%                           |
| 16×4/4×16, TT       | VTM           | 2   | 3                 | 1                              |
|                     | Proposed      | 2   | 3                 | 1                              |
|                     | <b>Saving</b> | 0%  | 0%                | 0%                             |



# Simulation results

## ■ CTC on top of VTM6.0

|                    | All Intra Main10 |        |       |      |      |
|--------------------|------------------|--------|-------|------|------|
|                    | Over VTM-6.0     |        |       |      |      |
|                    | Y                | U      | V     | EncT | DecT |
| Class F (optional) | 0.01%            | -0.01% | 0.03% | 99%  | 96%  |
| Class SCC 1080p    | 0.06%            | 0.04%  | 0.04% | 100% | 101% |

|                    | Random Access Main 10 |        |        |      |      |
|--------------------|-----------------------|--------|--------|------|------|
|                    | Over VTM-6.0          |        |        |      |      |
|                    | Y                     | U      | V      | EncT | DecT |
| Class F (optional) | -0.01%                | 0.05%  | 0.07%  | 99%  | 99%  |
| Class SCC 1080p    | 0.02%                 | -0.03% | -0.06% | 100% | 99%  |

|                    | Low delay B Main10 |        |        |      |      |
|--------------------|--------------------|--------|--------|------|------|
|                    | Over VTM-6.0       |        |        |      |      |
|                    | Y                  | U      | V      | EncT | DecT |
| Class F (optional) | -0.03%             | 0.02%  | 0.64%  | 100% | 101% |
| Class SCC 1080p    | -0.11%             | -0.12% | -0.03% | 98%  | 94%  |

# Conclusions

## ■ Benefits

- *Increase the MV throughput, especially for the worst case*
- *Reduce computational complexity by totally removing pruning process*
- *No need to construct a BV candidate list for a 4x4 block*

■ It is recommended adopting the proposed method

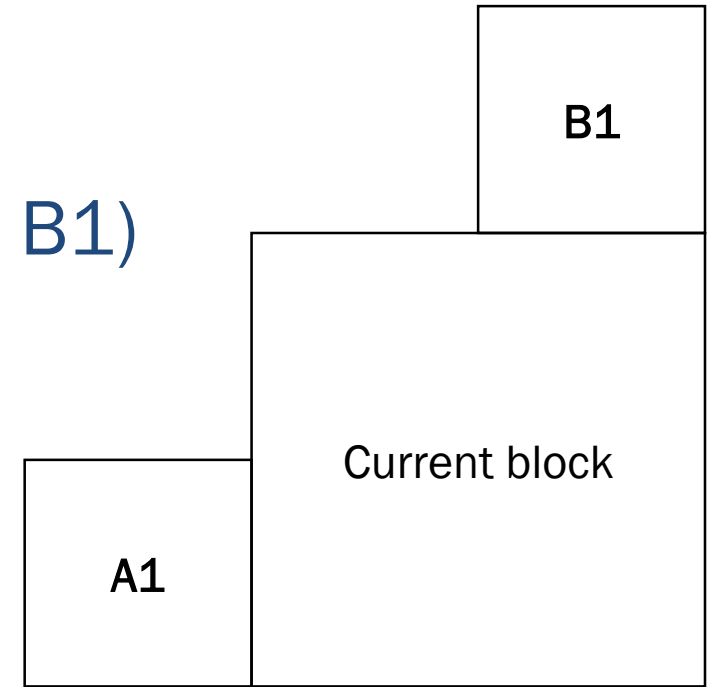
Thanks Alibaba for crosschecking!



# JVET-P0400: Background

## ■ Current design

- *BV candidate construction process*
  - 2 spatial neighbouring positions (A1, B1)
  - 5 HMVP candidates
  - Zero vectors by default
- *Prunings (3 times per block)*
  - B1 vs A1
  - Last 1 HMVP candidate compared to 2 spatial candidates
- *Shared merge list*



# JVET-P0400: Background

- Higher probability of HMVP candidates in a HMVP table with the spatial merge candidates

