|  |  |
| --- | --- |
| **Joint Collaborative Team on 3D Video Coding Extensions**  **of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11**  7th Meeting: San José, US, 11–17 Jan. 2014 | Document: JCT3V-G0034 |

|  |  |  |  |
| --- | --- | --- | --- |
| *Title:* | **CE3-related: Worst case complexity reduction for merge candidate construction** | | |
| *Status:* | Input Document to JCT-3V | | |
| *Purpose:* | Proposal | | |
| *Author(s) or Contact(s):* | Tomohiro Ikai  1-9-2 Nakase, Mihama-ku, Chiba-shi, Chiba 261-8520  JAPAN | Tel: Email: | +81-43-299-8526 [ikai.tomohiro@sharp.co.jp](mailto:ikai.tomohiro@sharp.co.jp) |
| *Source:* | Sharp Corporation | | |

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# Abstract

This contribution proposes to reduce worst case complexity for merge candidate derivation process in 3D-HEVC. The derivation process requires arbitrary inter-layer / inter-component access and increases checking process. It is asserted that this complexity increase in the worst case causes narrower bottle neck (in software) or increased implementation cost (in hardware). On the other hand, those additional candidates are not effective very much at the small blocks. This proposal omits additional 3D-HEVC candidates in 8x4 and 4x8 PU to reduce the worst case complexity. The experiment shows that the BD-rate gain is 0.0 %, 0.0 % and 0.0 % in video, total video and synthesis respectively.

# Introduction

The merge candidate construction in HEVC version 1 is quite simple in terms of memory access and parallelism. It emploies only static memory access (colPU) and uses the small number of neibouring check and pruning check. On the other hand, 3D-HEVC adds texture cand (T), inter-view cand (IvMC), disparity cand (IvDC), shifted inter view cand (IvMcShift), shifted disparity cand (IvDCShift) and VSP cand(VSP) and increase complexity in light of arbitrary memory access (DV base access) and check.

The consequence can be severe in 8x4 and 4x8 PU, where it is needed to apply all additional construction in the same way of larger PU. Besides, the additional merge candidates are not so efficient in these blocks because small block, which tends to be occlusion area or shape change area, has relatively little inter-view / inter component similarity.

Table 1 shows the summary of the merge candidate construction process

Table 1 Summary of merge candidate construction process

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Merge | Step | Access Type | Additional access | Additional pruning check / merge candidate check |
|  | **T** | colPU | **Yes** |  |
|  | **IvMC** | ivPU | **Yes** |  |
|  | A1 | neighPU |  | **Yes** |
|  | B1 | neighPU |  | **Yes** |
|  | B0 | neighPU |  |  |
|  | **IvDC** | mvDisp |  | **Yes** |
|  | **VSP** |  | **Yes** (MC part) |  |
|  | A0 | Left |  |  |
|  | B2 | Upper |  |  |
|  | **IvMCShift** | ivPU | **Yes** |  |
|  | **IvDCShift** | mergeCand | **Yes** |  |
|  | Col | colPU | **Yes (AltCol)** |  |
|  | combined-Bi | mergeCand |  |  |
|  | Zero | No |  |  |

# Proposal

If the PU size is 8x4 and 4x8, the additional 3D-HEVC candidates is not included in the merge mode candidate lists, otherwise those candidates are added (the same as HTM9).

**H.8.5.3.2.1** **Derivation process for luma motion vectors for merge mode**

* 1. The derivation process for the base merge candidate list as specified in subclause xxx is invoked with the luma location ( xCb, yCb ), the luma location ( xPb, yPb ), the variables nCbS, nPbW, nPbH, and the partition index partIdx as inputs, and the output being a modified luma location ( xPb, yPb ), the modified variables nPbW and nPbH, the modified variable partIdx, the luma location ( xOrigP, yOrigP ), the variables nOrigPbW and nOrigPbH, the merge candidate list baseMergeCandList, the luma motion vectors mvL0N and mvL1N, the reference indices refIdxL0N and refIdxL1N, and the prediction list utilization flags predFlagL0N and predFlagL1N, with N being replaced by all elements of baseMergeCandList.
  2. For N being replaced by A1, B1, B0, A0 and B2, the following applies:
     + If N is an element in baseMergeCandList, availableFlagN is set equal to 1.
     + Otherwise (N is not an element in baseMergeCandList), availableFlagN is set equal to 0.
  3. Depending on iv\_mv\_pred\_flag[ nuh\_layer\_id ], the following applies:
     + If iv\_mv\_pred\_flag[ nuh\_layer\_id ] is equal to 0, the flags availableFlagIvMC, availableIvMCShift and availableFlagIvDC are set equal to 0.
     + Otherwise (iv\_mv\_pred\_flag[ nuh\_layer\_id ] is equal to 1), the derivation process for the inter-view merge candidates as specified in subclause H.8.5.3.2.10 is invoked with the luma location ( xPb, yPb ), the variables nPbW and nPbH, as the inputs and the output is assigned to the availability flags availableFlagIvMC, availableIvMCShift and availableFlagIvDC, the reference indices refIdxLXIvMC, refIdxLXIvMCShift and refIdxLXIvDC, the prediction list utilization flags predFlagLXIvMC, predFlagLXivMCShift and predFlagLXIvDC, and the motion vectors mvLXIvMC, mvLXIvMCShift and mvLXIvDC (with X being 0 or 1, respectively).
  4. Depending on view\_synthesis\_pred\_flag[ nuh\_layer\_id ], the following applies:
     + If view\_synthesis\_pred\_flag[ nuh\_layer\_id ] is equal to 0, the flag availableFlagVSP is set equal to 0.
     + Otherwise (view\_synthesis\_pred\_flag[ nuh\_layer\_id ] is equal to 1), the derivation process for a view synthesis prediction merge candidate as specified in subclause H.8.5.3.2.13 is invoked with the luma locations ( xCb, yCb ) as input and the outputs are the availability flag availableFlagVSP, the reference indices refIdxL0VSP and refIdxL1VSP, the prediction list utilization flags predFlagL0VSP and predFlagL1VSP, and the motion vectors mvL0VSP and mvL1VSP.
  5. Depending on mpi\_flag[ nuh\_layer\_id ], the following applies:
     + If mpi\_flag[ nuh\_layer\_id ] is equal to 0, the variable availableFlagT is set equal to 0.
     + Otherwise (mpi\_flag[ nuh\_layer\_id ] is equal to 1), the derivation process for the texture merging candidate as specified in subclause H.8.5.3.2.14 is invoked with the luma location ( xPb, yPb ), the variables nPbW and nPbH as the inputs and the outputs are the flag availableFlagT, the prediction utilization flags predFlagL0T and predFlagL1T, the reference indices refIdxL0T and refIdxL1T, and the motion vectors mvL0T and mvL1T.
  6. The merging candidate list, extMergeCandList, is constructed as follows:

i = 0  
 if( availableFlagT && ( nOrigPbW + nOrigPbH ) > 12))  
 extMergeCandList[ i++ ] = T  
 if( availableFlagIvMC && ( !availableFlagT  | |  differentMotion( T, IvMC ) ) && ( nOrigPbW + nOrigPbH ) > 12))  
 extMergeCandList[ i++ ] = IvMC  
 N = DepthFlag ? T : IvMC  
 if( availableFlagA1 && ( !availableFlagN  | |  differentMotion( N, A1 ) ) )  
 extMergeCandList[ i++ ] = A1 if( availableFlagB1 && ( !availableFlagN  | |  differentMotion( N, B1 ) ) )  
 extMergeCandList[ i++ ] = B1 if( availableFlagB0 )  
 extMergeCandList[ i++ ] = B0 (H‑99) if( availableFlagIvDC && ( !availableFlagA1  | |  differentMotion( A1, IvDC ) ) &&   
 ( !availableFlagB1  | |  differentMotion( B1, IvDC ) ) && ( nOrigPbW + nOrigPbH ) > 12))  
 extMergeCandList[ i++ ] = IvDC  
 if( availableFlagVSP && !ic\_flag && iv\_res\_pred\_weight\_idx = = 0 )  
 extMergeCandList[ i++ ] = VSP  
 if( availableFlagA0 )  
 extMergeCandList[ i++ ] = A0 if( availableFlagB2 )  
 extMergeCandList[ i++ ] = B2 if( availableFlagIvMCShift && i < ( 5 + NumExtraMergeCand ) &&  
 ( !availableFlagIvMC  | |  differentMotion( IvMC, IvMCShift ) ) && ( nOrigPbW + nOrigPbH ) > 12))  
 extMergeCandList[ i++ ] = IvMCShift

* 1. The variable availableFlagIvDCShift is set equal to 0, and when DepthFlag is equal to 0, availableFlagIvMCShift is equal to 0, and i is less than ( 5 + NumExtraMergeCand ) and ( nOrigPbW + nOrigPbH ) is larger than 12), the derivation process for the shifted disparity merging candidate as specified in subclause H.8.5.3.2.15 is invoked with the luma location ( xPb, yPb ), the variables nPbW and nPbH, and the availability flags availableFlagN, the reference indices refIdxL0N and refIdxL1N, the prediction list utilization flags predFlagL0N and predFlagL1N, the motion vectors mvL0N and mvL1N, of every candidate N being in extMergeCandList, extMergeCandList, and i as the inputs and the outputs are the flag availableFlagIvDCShift, the prediction utilization flags predFlagL0IvDCShift and predFlagL1IvDCShift, the reference indices refIdxL0IvDCShift and refIdxL1IvDCShift, and the motion vectors mvL0IvDCShift and mvL1IvDCShift.
  2. The merging candidate list, extMergeCandList, is constructed as follows:

if( availableFlagIvDCShift )  
 extMergeCandList[ i++ ] = IvDCShift  
 j = 0  
 while( i < MaxNumMergeCand ) {(H‑99)  
 N = baseMergeCandList[ j++ ]  
 if( N != A1 && N != B1 && N != B0 && N != A0 && N != B2 )  
 extMergeCandList[ i++ ] = N  
 }

* 1. The variable N is set equal to extMergeCandList[ merge\_idx[ xOrigP ][ yOrigP ] ].
  2. The variable subPbMotionFlag is set equal to ( N = = IvMC ).
  3. The following assignments are made with X being replaced by 0 or 1:
     + 1. mvLX = subPbMotionFlag ? 0 : mvLXN (H‑99)
       2. refIdxLX = subPbMotionFlag ? −1 : refIdxLXN (H‑101)
       3. predFlagLX = subPbMotionFlag ? 0 : predFlagLXN (H‑102)
  4. When predFlagL0 is equal to 1 and predFlagL1 is equal to 1, and ( nOrigPbW + nOrigPbH ) is equal to 12, the following applies
     + 1. refIdxL1 = −1 (H‑103)
       2. predFlagL1 = 0 (H‑104)
  5. The derivation process for a view synthesis prediction flag as specified in subclause xxx is invoked with the luma location ( xPb, yPb ), the variables nPbW and nPbH, the merge candidate indicator N as the inputs, and the output is the mergeCandIsVspFlag.
  6. The variable vspModeFlag is derived as specified in the following:
     1. vspModeFlag = mergeCandIsVspFlag && !ic\_flag && ( iv\_res\_pred\_weight\_idx = = 0 ) (H‑105)

# Simulation results

Experiment result based on HTM90r1 is shown in Table 2

Table 2 Experimental results (anchor: HTM90r1)

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | video 0 | video 1 | video 2 | video PSNR / video bitrate | video PSNR / total bitrate | synth PSNR / total bitrate | enc time | dec time | ren time |
| Balloons | 0.0% | -0.2% | 0.1% | 0.00% | -0.01% | -0.02% | 99.9% | 97.9% | 100.5% |
| Kendo | 0.0% | -0.2% | -0.2% | -0.07% | -0.06% | -0.06% | 97.8% | 99.3% | 99.9% |
| Newspaper\_CC | 0.0% | -0.1% | 0.0% | 0.00% | -0.02% | 0.00% | 97.6% | 99.2% | 100.1% |
| GT\_Fly | 0.0% | -0.1% | -0.1% | 0.01% | 0.01% | -0.02% | 99.4% | 99.1% | 99.8% |
| Poznan\_Hall2 | 0.0% | 0.0% | -0.2% | 0.00% | 0.01% | 0.01% | 99.9% | 100.1% | 100.7% |
| Poznan\_Street | 0.0% | -0.1% | -0.1% | -0.01% | -0.01% | -0.02% | 100.1% | 97.9% | 99.3% |
| Undo\_Dancer | 0.0% | -0.1% | -0.1% | 0.00% | 0.00% | 0.00% | 99.8% | 100.9% | 99.9% |
| Shark | 0.0% | 0.0% | 0.0% | 0.01% | 0.00% | 0.01% | 99.3% | 99.0% | 100.3% |
| 1024x768 | 0.0% | -0.1% | 0.0% | -0.02% | -0.03% | -0.03% | 98.4% | 98.8% | 100.2% |
| 1920x1088 | 0.0% | 0.0% | -0.1% | 0.00% | 0.00% | 0.00% | 99.7% | 99.4% | 100.0% |
| **average** | **0.0%** | **-0.1%** | **-0.1%** | **-0.01%** | **-0.01%** | **-0.01%** | **99.2%** | **99.2%** | **100.1%** |

# Conclusion

This proposal omits additional 3D-HEVC candidates in 8x4 and 4x8 PU. It is recommended to adopt this method in 3D-HEVC.

# Patent rights declaration(s)

**Sharp Corporation may have current or pending patent rights relating to the technology described in this contribution and, conditioned on reciprocity, is prepared to grant licenses under reasonable and non-discriminatory terms as necessary for implementation of the resulting ITU-T Recommendation | ISO/IEC International Standard (per box 2 of the ITU-T/ITU-R/ISO/IEC patent statement and licensing declaration form).**