|  |  |
| --- | --- |
| **Joint Collaborative Team on 3D Video Coding Extensions**  **of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11**  9th Meeting: Sapporo, JP, 3 – 9 July 2014 | Document: JCT3V-J0045 |

|  |  |  |  |
| --- | --- | --- | --- |
| *Title:* | **Alignment of motion derivation from other component** | | |
| *Status:* | Input Document | | |
| *Purpose:* | Proposal | | |
| *Author(s) or Contact(s):* | Junghak Nam  Sehoon Yea  19, Yangjae-daero 11gil, Seocho-gu Seoul 137-130, Korea | Tel: Email: | +82-2-6912-6477 [junghak.nam@lge.com](mailto:junghak.nam@lge.com) |
| *Source:* | LG Electronics | | |

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

The newly added parts compared to 3D-HEVC working draft 3 are highlighted in green and the removed parts are marked with ~~strikethrough~~.

I.8.5.3.2.15 Derivation process for a texture merging candidate

This process is not invoked when mpi\_flag[ nuh\_layer\_id ] is equal to 0.

Inputs to this process are:

* a luma location ( xPb, yPb ) of the top-left sample of the current luma prediction block relative to the top-left luma sample of the current picture,
* two variables nPbW and nPbH specifying the width and the height of the current prediction block,
* a flag mvAccFlag specifying whether the output motion vectors have the same accuracy with texture motion vectors.

Outputs of this process are:

* a flag availableFlagT specifying whether the texture merging candidate is available,
* the prediction utilization flags predFlagL0T and predFlagL1T,
* the reference indices refIdxL0T and refIdxL1T (when availableFlagT is equal to 1),
* the motion vectors mvL0T and mvL1T (when availableFlagT is equal to 1).

The variable availableFlagT is set equal to 0. The variables predFlagL0T and predFlagL1T are set equal to 0. The variables refIdxL0T and refIdxL1T are set equal to −1. Both components of the motion vectors mvL0T and mvL1T are set equal to 0.

The texture luma location ( xRef, yRef ) is derived by:

* + - * 1. xRefFull = xPb + ( ( nPbW ~~− 1~~ ) >> 1 ) (‑156)
        2. yRefFull = yPb + ( ( nPbH ~~− 1~~ ) >> 1 ) (‑157)
        3. xRef = ( xRefFull >> 3 ) << 3 (‑158)
        4. yRef = ( yRefFull >> 3 ) << 3 (‑159)

Let textPb be the prediction block covering the position ( xRef, yRef ) in TexturePic.

For X in the range of 0 to 1, inclusive, the following applies:

* 1. The arrays textPredFlagLX[ x ][ y ], textRefIdxLX[ x ][ y ], and textMvLX[ x ][ y ], are set equal to the corresponding arrays of the texture picture specified by textPic, PredFlagLX[ x ][ y ], RefIdxLX[ x ][ y ], and MvLX[ x ][ y ].
  2. The list textRefPicListLX is set to be the reference picture list RefPicListX of the slice containing prediction block textPb in the picture textPic.
  3. The variable textRefPicListLX is set equal to the variable RefPicListX of textPic.
  4. The variable availableFlag is set equal to 0.
  5. When X is equal to 0 or the current slice is a B slice, for i in the range of 0 to NumRefPicsLX − 1, inclusive, the following applies:
     + When all of the following conditions are true, availableFlag is set equal to 1,
       - textPredFlagLX[ xRef ][ yRef ] is equal to 1
       - PicOrderCnt( RefPicListX[ i ] ) is equal to PicOrderCnt( textRefPicListX[ textRefIdxLX ])
       - ViewIdx( RefPicListX[ i ] ) is equal to ViewIdx( textRefPicListX[ textRefIdxLX ])
     + When predFlagLXT is equal to 0 and availableFlag is equal to 1, the texture merging candidate is derived as follows:
       - 1. mvLXT[ 0 ] = ( textMvLX[ xRef ][ yRef ][ 0 ] + 2 \* mvAccFlag ) >> ( 2 \* mvAccFlag ) (I‑160)
         2. mvLXT[ 1 ] = ( textMvLX[ xRef ][ yRef ][ 1 ] + 2 \* mvAccFlag ) >> ( 2 \* mvAccFlag ) (I‑161)
         3. refIdxLXT = i (I‑162)
         4. predFlagLXT = 1 (I‑163)
         5. availableFlagT = 1 (I‑164)

I.8.5.3.3.7.4 Derivation process for a motion vector from a reference block for residual prediction

Inputs to this process are:

* a luma location ( xP, yP ) of the top-left sample of the current luma prediction block relative to the top-left luma sample of the current picture,
* two variables nPbW and nPbH specifying the width and the height of the current luma prediction block,
* a reference picture refPic,
* a motion vector mvDisp

Outputs of this process are:

* a flag availFlag,
* a motion vector mvT,
* prediction list utilization variable Y.

The variable availFlag is set to 0 and the reference luma location ( xRef, yRef ) in refPicLX is derived by

~~xRef = Clip3( 0, PicWidthInSamples~~~~L~~~~− 1, xP + ( nPSW >> 1 ) + ( ( mvDisp[ 0 ] + 2 ) >> 2 ) ) (‑241)  
yRef = Clip3( 0, PicHeightInSamples~~~~L~~~~− 1, yP + ( nPSH >> 1 ) + ( ( mvDisp[ 1 ] + 2 ) >> 2 ) ) (‑242)~~

xRefFull = xP + ( nPSW >> 1 ) + ( ( mvDisp[ 0 ] + 2 ) >> 2 ) (‑241)

yRefFull = yP + ( nPSH >> 1 ) + ( ( mvDisp[ 1 ] + 2 ) >> 2 ) (‑242)

xRef = Clip3( 0, PicWidthInSamplesL − 1, ( xRefFull >> 3 ) << 3 ) (‑243)

yRef = Clip3( 0, PicHeightInSamplesL − 1, ( yRefFull >> 3 ) << 3 ) (‑244)

Let variable refCU and refPU be the coding unit and prediction unit that cover the luma location ( xRef, yRef ) in refPic, respectively.

When the variable CuPredMode for the coding unit refCU is equal to MODE\_SKIP or MODE\_INTER, the following applies for X in the range of 0 to 1, inclusive:

* + The variable refPredFlagLX is set equal to the prediction utilization flag predFlagLX of the prediction unit refPU.
  + When availFlag is equal to 0 and refPredFlagLX is equal to 1, the following applies:
    - Let refPicListRefX be the reference picture list X of refPic.
    - Let mvLX and refIdxLX be the motion vector and reference index of the prediction unit refPU corresponding to refPicListRefX, respectively.
    - When PicOrderCnt( refPicListRefX[ refIdxLX ] ) is not equal to PicOrderCnt( refPic ) and RpRefIdxLX is not equal to −1, availFlag is set to 1, Y is set equal to X and the residual prediction motion vector scaling process as specified in subclause I.8.5.3.3.7.3 is invoked with the prediction list utilization variable equal to X, the motion vector mvLX, and the reference picture refPicListRefX[ refIdxLX ] as inputs, and the output being mvT.