|  |  |
| --- | --- |
| **Joint Collaborative Team on 3D Video Coding Extensions**  **of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11**  11th Meeting: Geneva, CH, 12 – 18 February 2015 | Document: JCT3V-K0052\_v2 |

|  |  |  |  |
| --- | --- | --- | --- |
| *Title:* | **Proposed TMVP fix and CU syntax cleanup for 3D-HEVC** | | |
| *Status:* | Input Document | | |
| *Purpose:* | Proposal | | |
| *Author(s) or Contact(s):* | Gerhard Tech  Karsten Sühring  Detlev Marpe | Email: | gerhard.tech@hhi.fraunhofer.de  karsten.suehring@hhi.fraunhofer.de  detlev.marpe@hhi.fraunhofer.de |
| *Source:* | Fraunhofer HHI | | |

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

# Abstract

Version 2 (\_v2) of this document includes specification text for alternative reference index modification only.

Fixes for following issues in 3D-HEVC are proposed:

* Conditions on picture types in TMVP
* Moving of syntax elements to the CU extension

# Temporal motion vector prediction

## Current design

3D-HEVC extends TMVP by two new features, which are:

* The selection of an alternative reference picture index, when the picture types (long-term or short-term) of the current reference picture and the collocated reference picture are not matching, such that the current reference picture is changed to a reference picture having the type of collocated reference picture.
* Motion vector scaling based on view id, when both reference pictures are long-term pictures.

Whether the tools are applied depends on the picture type and is based on the assumption that inter-view reference pictures are marked as "long-term pictures".

However, although all inter-view reference pictures are marked as long-term pictures, not all long-term pictures are inter-view reference pictures. Thus, although not explicitly done in the current specification, actually three types of reference pictures can be distinguished:

1. short-term reference pictures, which are always used for temporal inter prediction (ST-T)
2. long-term reference pictures, which are used for inter-view prediction (LT-IV)
3. long-term reference pictures, which are used for temporal inter prediction (LT-T)

Table 1 shows, depending on the picture type of the current reference picture and the collocated reference picture, whether the current reference picture is changed to the alternative reference picture and the picture type of the alternative reference picture.

Table 1: Alternative reference index approach, current design

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | | Current reference picture type | | |
| ST-T | LT-T | LT-IV |
| Collocated reference picture type | ST-T | no change | changed to ST-T | changed to ST-T |
| LT-T | changed to LT-T or LT-IV | no change | no change |
| LT-IV | changed to LT-T or LT-IV | no change | no change |

Similarly Table 2 shows whether motion vector scaling is applied and whether the TMVP is available.

Table 2: Motion vector scaling and TMVP availability, current design

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | | Current reference picture type | | |
| ST-T | LT-T | LT-IV |
| Collocated reference picture type | ST-T | temporal scaling | not available | not available |
| LT-T | not available | no scaling | view id based scaling |
| LT-IV | not available | view id based scaling | view id based scaling |

## Problem statement

### Alternative reference index

Design intent for using the alternative reference index is actually not matching reference picture types in a way that the collocated and the current reference picture are either a) both ST-T; or b) one LT-T and the other LT-IV, but to match them in a way they are either a) both ST-T; or b) both LT-IV.

As indicated in red in Table 1 this intention fails, when LT-T pictures are present e.g.:

* When the current reference picture is ST-T and collocated reference picture is LT-IV, the current reference picture might be changed to a LT-T reference picture instead of changing it to a LI-IV reference picture.
* When the current reference picture is LT-T and collocated reference picture is LT-IV, the current reference picture is not changed, although reference pictures types are not matching.

Thus, the current specification fails to match the design intent, when LT-T reference pictures are present.

### View id based scaling

As indicated in red in Table 2, view id based MV scaling is performed also in cases in which one reference picture is a LT-T reference picture and the other a LT-IV reference picture. Since the view id difference between the current or collocated picture and their respected LT-T reference pictures is equal to zero, such scaling leads either to a zero motion vector or division by zero and thus to undefined decoder state.

## Proposal

To resolve issue 2.2.1, it is proposed to change to:

1. Selected the alternative reference index, such that the alternative reference picture is a
   1. LT-IV reference picture when the current reference picture is a ST-T picture.
   2. LT-IV reference picture when the current reference picture is a LT-T picture.
   3. ST-T reference picture when the current reference picture is a LT-IV picture.
2. And to use the alternative picture as depicted in Table 3.

Table 3: Alternative reference index approach, proposal

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | | Current reference picture type | | |
| ST-T | LT-T | LT-IV |
| Collocated reference picture type | ST-T | no change | no change | changed to ST-T |
| LT-T | no change | no change | changed to ST-T |
| LT-IV | changed to LT-IV | change to LT-IV | no change |

Thus, the alternative reference index approach is applied to match temporal and inter-view prediction. Note that, in the case that the current picture is a LT-IV reference picture and the collocated reference picture is a LT-T picture, changing the current reference picture to a ST-T reference picture has actually no effect, since TMVP would still be disabled. However, this change allows a simplified logic in the specification.

To resolve issue 2.2.2 it is proposed to disable view id based MV scaling when the current reference picture is a LT-IV reference picture and the collocated reference picture is a LT-T reference picture, or vice versa, as depicted in Table 4.

Table 4: Motion vector scaling and TMVP availability, proposal

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | | Current reference picture type | | |
| ST-T | LT-T | LT-IV |
| Collocated reference picture type | ST-T | temporal scaling | not available | not available |
| LT-T | not available | no scaling | not available |
| LT-IV | not available | not available | view id based scaling |

### Specification changes

**I.8.3.7 Derivation process for the alternative target reference index for TMVP in merge mode**

This process is invoked when the current slice is a P or B slice.

The variables AltRefIdxL0 and AltRefIdxL1 are set equal to −1 and the following applies for X in the range of 0 to 1, inclusive:

Let currPic be the current picture.

When X is equal to 0 or the current slice is a B slice the following applies:

~~zeroIdxLtFlag = RefPicListX[ 0 ] is a short-term reference picture ? 0 : 1~~ zeroIdxLtIvFlag = ( DiffPicOrderCnt( currPic, RefPicListX[ 0 ] ) = = 0 )  
 for( i = 1; i <= num\_ref\_idx\_lX\_active\_minus1 && AltRefIdxLX = = −1; i++ ) {  
 ~~if( ( zeroIdxLtFlag && RefPicListX[ i ] is a short-term reference picture ) | |  
 ( !zeroIdxLtFlag && RefPicListX[ i ] is a long-term reference picture ) )~~ candIdxLtIvFlag = ( DiffPicOrderCnt( currPic, RefPicListX[ 0 ] ) = = 0 )  
 candIdxStFlag = ( RefPicListX[ i ] is a short-term reference picture ) ? 1 : 0  
 if( ( zeroIdxLtIvFlag && candIdxStFlag ) | | ( !zeroIdxLtIvFlag && candIdxLtIvFlag ) )  
 AltRefIdxLX = i

I.8.5.3.2.9 Derivation process for collocated motion vectors in merge mode

NOTE – This process is only invoked if merge\_flag of the current PU is equal to 1. Otherwise, (merge\_flag of the current PU is equal to 0), the derivation process for collocated motion vectors as specified in clause 8.5.3.2.9 might be invoked.

Inputs to this process are:

* a variable currPb specifying the current prediction block,
* a variable colPb specifying the collocated prediction block inside the collocated picture specified by ColPic,
* a luma location ( xColPb, yColPb ) specifying the top-left sample of the collocated luma prediction block specified by colPb relative to the top-left luma sample of the collocated picture specified by ColPic,
* a reference index refIdxLX, with X being 0 or 1.

Outputs of this process are:

* the motion vector prediction mvLXCol,
* the availability flag availableFlagLXCol.

The variable currPic specifies the current picture.

The arrays predFlagL0Col[ x ][ y ], mvL0Col[ x ][ y ], and refIdxL0Col[ x ][ y ] are set equal to PredFlagL0[ x ][ y ], MvL0[ x ][ y ], and RefIdxL0[ x ][ y ], respectively, of the collocated picture specified by ColPic, and the arrays predFlagL1Col[ x ][ y ], mvL1Col[ x ][ y ], and refIdxL1Col[ x ][ y ] are set equal to PredFlagL1[ x ][ y ], MvL1[ x ][ y ], and RefIdxL1[ x ][ y ], respectively, of collocated the picture specified by ColPic.

The variables mvLXCol and availableFlagLXCol are derived as follows:

* If colPb is coded in an intra prediction mode, both components of mvLXCol are set equal to 0 and availableFlagLXCol is set equal to 0.
* Otherwise, the motion vector mvCol, the reference index refIdxCol, and the reference list identifier listCol are derived as follows:
  + If predFlagL0Col[ xColPb ][ yColPb ] is equal to 0, mvCol, refIdxCol, and listCol are set equal to mvL1Col[ xColPb ][ yColPb ], refIdxL1Col[ xColPb ][ yColPb ], and L1, respectively.
  + Otherwise, if predFlagL0Col[ xColPb ][ yColPb ] is equal to 1 and predFlagL1Col[ xColPb ][ yColPb ] is equal to 0, mvCol, refIdxCol, and listCol are set equal to mvL0Col[ xColPb ][ yColPb ], refIdxL0Col[ xColPb ][ yColPb ], and L0, respectively.
  + Otherwise (predFlagL0Col[ xColPb ][ yColPb ] is equal to 1 and predFlagL1Col[ xColPb ][ yColPb ] is equal to 1), the following assignments are made:
    - If NoBackwardPredFlag is equal to 1, mvCol, refIdxCol, and listCol are set equal to mvLXCol[ xColPb ][ yColPb ], refIdxLXCol[ xColPb ][ yColPb ], and LX, respectively.
    - Otherwise, mvCol, refIdxCol, and listCol are set equal to mvLNCol[ xColPb ][ yColPb ], refIdxLNCol[ xColPb ][ yColPb ], and LN, respectively, with N being the value of collocated\_from\_l0\_flag.

and mvLXCol and availableFlagLXCol are derived as follows:

* + The variables curSt~~Iv~~Flag and colSt~~Iv~~Flag, specifying whether the reference pictures of the ~~inter-view prediction is utilized for the~~ current and collocated PU are short-term reference pictures derived as:
    - 1. curSt~~Iv~~Flag = !LongTermRefPic( currPic, currPb, refIdxLX, LX ) (I−128)
      2. colSt~~Iv~~Flag = !LongTermRefPic( ColPic, colPb, refIdxCol, listCol ) (I−129)
  + The variable refPicListCol[ refIdxCol ] is set to be the picture with reference index refIdxCol in the reference picture list listCol of the slice containing prediction block currPb in the collocated picture specified by ColPic.
  + The variables curLtIvFlag and colLtIvFlag, specifying whether inter−view prediction is utilized for the current and collocated PU are derived as:
    - 1. curLtIvFlag = ( DiffPicOrderCnt( currPic, RefPicListX[ refIdxLX ] ) = = 0 ) (I−128)
      2. colLtIvFlag = ( DiffPicOrderCnt( ColPic, refPicListCol[ refIdxCol ] ) = = 0 ) (I−129)
  + When curLtIvFlag is not equal to colLtIvFlag, and AltRefIdxLX is not equal to −1, the variables, AltRefFlagLX, refIdxLX and curIvFlag are modified as follows:
    - 1. AltRefFlagLX = 1 (I−130)
      2. refIdxLX = AltRefIdxLX (I−130)
      3. ~~curIvFlag = LongTermRefPic( currPic, currPb, refIdxLX, LX ) (I−131)~~
      4. curLtIvFlag = ( DiffPicOrderCnt( currPic, RefPicListX[ refIdxLX ] ) = = 0 ) (I−131)
  + The motion vector mvLXCol is modified as follows:
    - If curLtIvFlag is not equal to colLtIvFlag, or curStFlag is not equal to colStFlag, both components of mvLXCol are set equal to 0 and availableFlagLXCol is set equal to 0.
    - Otherwise, the variable availableFlagLXCol is set equal to 1, ~~refPicListCol[ refIdxCol ] is set to be the picture with reference index refIdxCol in the reference picture list listCol of the slice containing prediction block currPb in the collocated picture specified by ColPic,~~ and the following applies:
      * + The variables colDiff and currDiff specifying a POC or ViewId difference are ~~derived~~ as follows:

~~If curIvFlag is equal to 0 or ( ( ViewIdx  !=  0 )  &&  IvMvScalingFlag ) is equal to 0, the following applies:~~

If curStFlag is equal to 1, the following applies:

colDiff = DiffPicOrderCnt( ColPic, refPicListCol[ refIdxCol ] ) (I−132)

currDiff = DiffPicOrderCnt( currPic, RefPicListX[ refIdxLX ] ) (I−133)

~~Otherwise, (curIvFlag is equal to 1 and ( ( ViewIdx  !=  0  )  &&   IvMvScalingFlag ) ise equal to 1), the following applies:~~

Otherwise, if curLtIvFlag, ( ViewIdx  !=  0  ) , and IvMvScalingFlag are equal to 1, the following applies:

colDiff = DiffViewId( ColPic, refPicListCol[ refIdxCol ] ) (I−134)

currDiff = DiffViewId( currPic, RefPicListX[ refIdxLX ] ) (I−135)

Otherwise, colDiff and currDiff are set equal to 0.

* + - * + If colDiff is equal to currDiff, mvLXCol is derived as follows:

mvLXCol = mvCol (I−136)

* + - * + Otherwise, mvLXCol is derived as a scaled version of the motion vector mvCol as follows:

tx = ( 16384 + ( Abs( td )  >>  1 ) ) / td (I−137)

distScaleFactor = Clip3( −4096, 4095, ( tb \* tx + 32 )  >>  6 ) (I−138)

mvLXCol = Clip3( −32768, 32767, Sign( distScaleFactor \* mvCol ) \*   
 ( ( Abs( distScaleFactor \* mvCol ) + 127 )  >>  8 ) ) (I−139)

where td and tb are derived as follows:

td = Clip3( −128, 127, colDiff ) (I−140)

tb = Clip3( −128, 127, currDiff ) (I−141)

# CU extension

## Current design

Currently dbbp\_flag and sdc\_flag are signaled in CU syntax structure following the part\_mode syntax element. To avoid the number of places where the CU syntax is modified, other 3D-HEVC CU syntax elements are collectively signalled in the CU extension (if possible).

## Problem statement

The dbbp\_flag and sdc\_flag are currently not signalled in the CU extension, although possible.

## Proposed solution

It is proposed to clean up the syntax and move dbbp\_flag and sdc\_flag syntax elements to the CU extension.

### Syntax changes

|  |  |
| --- | --- |
| coding\_unit( x0, y0, log2CbSize , ctDepth) { | **Descriptor** |
| ... |  |
| if( !cu\_skip\_flag[ x0 ][ y0 ] && !single\_sample\_mode\_flag[ x0 ][ y0 ] ) { |  |
| if( slice\_type != I ) |  |
| **pred\_mode\_flag** | ae(v) |
| if( ( CuPredMode[ x0 ][ y0 ] != MODE\_INTRA | |   log2CbSize = = MinCbLog2SizeY ) && !predPartModeFlag ) |  |
| **part\_mode** | ae(v) |
| ~~if( DepthBasedBlkPartFlag && DispAvailFlag && ( log2CbSize > 3 ) &&  ( PartMode = = PART\_2NxN  | |  PartMode = = PART\_Nx2N ) )~~ |  |
| **~~dbbp\_flag~~**~~[ x0 ][ y0 ]~~ | ~~ae(v)~~ |
| ~~if( sdcEnableFlag )~~ |  |
| **~~sdc\_flag~~**~~[ x0 ][ y0 ]~~ | ~~ae(v)~~ |
| if( CuPredMode[ x0 ][ y0 ] = = MODE\_INTRA ) { |  |
| if( PartMode = = PART\_2Nx2N && pcm\_enabled\_flag &&   log2CbSize >= Log2MinIpcmCbSizeY &&  log2CbSize <= Log2MaxIpcmCbSizeY ) |  |
| **pcm\_flag**[ x0 ][ y0 ] | ae(v) |
| if( pcm\_flag[ x0 ][ y0 ] ) { |  |
| while( !byte\_aligned( ) ) |  |
| **pcm\_alignment\_zero\_bit** | f(1) |
| pcm\_sample( x0, y0, log2CbSize ) |  |
| } else { |  |
| pbOffset = ( PartMode = = PART\_NxN ) ? ( nCbS / 2 ) : nCbS |  |
| log2PbSize = log2CbSize – ( ( PartMode = = PART\_NxN ) ? 1 : 0 ) |  |
| for( j = 0; j < nCbS; j = j + pbOffset ) |  |
| for( i = 0; i < nCbS; i = i + pbOffset ) { |  |
| if( IntraSdcWedgeFlag  | |  IntraContourFlag ) |  |
| intra\_mode\_ext( x0 + i ,  y0+ j , log2PbSize ) |  |
| if( dim\_not\_present\_flag[ x0 + i ][ y0 + j ] ) |  |
| **prev\_intra\_luma\_pred\_flag**[ x0 + i ][ y0 + j ] | ae(v) |
| } |  |
| for( j = 0; j < nCbS; j = j + pbOffset ) |  |
| for( i = 0; i < nCbS; i = i + pbOffset ) |  |
| if( dim\_not\_present\_flag[ x0 + i ][ y0 + j ] ) { |  |
| if( prev\_intra\_luma\_pred\_flag[ x0 + i ][ y0 + j ] ) |  |
| **mpm\_idx**[ x0 + i ][ y0 + j ] | ae(v) |
| else |  |
| **rem\_intra\_luma\_pred\_mode**[ x0 + i ][ y0 + j ] | ae(v) |
| } |  |
| ~~if( !sdc\_flag[ x0 ][ y0 ] ) {  [Ed. (GT) This aligns spec with current SW. However, since ChromaArrayType~~  ~~can only be 0 for depth maps, this is purely editorial.]~~ |  |
| if( ChromaArrayType = = 3 ) |  |
| for( j = 0; j < nCbS; j = j + pbOffset ) |  |
| for( i = 0; i < nCbS; i = i + pbOffset ) |  |
| **intra\_chroma\_pred\_mode**[ x0 + 1 ][ y0 + j ] | ae(v) |
| else if( ChromaArrayType != 0 ) |  |
| ... | ae(v) |

**Coding unit extension syntax**

|  |  |
| --- | --- |
| cu\_extension( x0 , y0 , log2CbSize ) { | **Descriptor** |
| if( single\_sample\_mode\_flag[ x0 ][ y0 ] ) |  |
| **single\_sample\_flag**[ x0 ][ y0 ] | ae(v) |
| else { |  |
| if( !cu\_skip\_flag[ x0 ][ y0 ] { |  |
| if( DepthBasedBlkPartFlag && DispAvailFlag && ( log2CbSize > 3 ) &&  ( PartMode = = PART\_2NxN  | |  PartMode = = PART\_Nx2N ) ) |  |
| **dbbp\_flag**[ x0 ][ y0 ] | ae(v) |
| if( sdcEnableFlag ) |  |
| **sdc\_flag**[ x0 ][ y0 ] | ae(v) |
| } |  |
| if ( rpEnableFlag ) |  |
| **iv\_res\_pred\_weight\_idx**[ x0 ][ y0 ] | ae(v) |
| if ( icEnableFlag && iv\_res\_pred\_weight\_idx[ x0 ][ y0 ] = = 0 ) |  |
| **ic\_flag**[ x0 ][ y0 ] | ae(v) |
| if( cuDepthDcPresentFlag ) { |  |
| pbOffset = ( PartMode = = PART\_NxN &&   CuPredMode[ x0 ][ y0 ] = = MODE\_INTRA ) ? ( nCbS / 2 ) : nCbS |  |
| for( j = 0; j < nCbS; j = j + pbOffset ) |  |
| for( k = 0; k < nCbS; k = k + pbOffset ) |  |
| if( DmmFlag[ x0 + k ][ y0 + j ] | | sdc\_flag[ x0 ][ y0 ] ) { |  |
| if( CuPredMode[ x0 ][ y0 ] = = MODE\_INTRA && sdc\_flag[ x0 ][ y0 ] ) |  |
| **depth\_dc\_flag**[ x0 + k ][ y0 + j ] | ae(v) |
| dcNumSeg = DmmFlag[ x0 + k ][ y0 + j ] ? 2 : 1 |  |
| if( depth\_dc\_flag[ x0 + k ][ y0 + j ] ) |  |
| for( i = 0; i < dcNumSeg; i++ ) { |  |
| **depth\_dc\_abs**[ x0 + k ][ y0 + j ][ i ] | ae(v) |
| if( ( depth\_dc\_abs[ x0 + k ][ y0 + j ][ i ] − dcNumSeg + 2 ) > 0 ) |  |
| **depth\_dc\_sign\_flag**[ x0 + k ][ y0 + j ][ i ] | ae(v) |
| } |  |
| } |  |
| } |  |
| } |  |
| } |  |

# Patent rights declaration(s)

Fraunhofer-Gesellschaft zur Förderung der angewandten Forschung e.V. 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).