G.8.5 Decoding process for coding units coded in inter prediction mode

The specifications in subclause 8.5 apply with the following modification:

– If use\_mvi\_flag is equal to 1 and ( PredMode is equal to MODE\_SKIP or PredMode is equal to MODE\_INTER and merge\_flag[ xP ][ yP ] is equal to 1 ) and merge\_idx[ xP ][ yP ] is equal to 0, the following ordered steps apply:

1. The variable TextureModeDepth[ x0 >> Log2MinCbSize ][ y0 >> Log2MinCbSize ] is set equal to the current value of the variable cbDepth.
2. The co-located region of the corresponding texture picture is set as the luma prediction block covering the modified location given by ( ( x0 >> 3 ) << 3, ( y0 >> 3 ) << 3 ) inside the corresponding texture picture
3. The syntax elements split\_coding\_unit\_flag[  ][  ] as well as the motion vectors and reference indices are inherited from the co-located region of the corresponding texture picture.

– All invocations of the process specified in subclause 8.5.1 are replaced with invocations of the process specified in subclause .

G.8.5.2.1.12 Derivation process for a temporal inter-view motion vector predictor candidate

This process is not invoked when multi\_view\_mv\_pred\_flag is equal to 0.

Inputs to this process are:

– a luma location ( xP, yP ) of the top-left luma sample of the current prediction unit relative to the top-left luma sample of the current picture,

– variables nPSW and nPSH specifying the width and the height, respectively, of the current prediction unit,

– a reference index refIdxLX (with X being equal to 0 or 1) specifying a reference picture in the reference picture list RefPicListLX,

– a view identifier refViewIdx specifying a reference view.

Outputs of this process are:

– a flag availableFlagLXInterView specifying whether the inter-view motion vector candidate is available,

– a motion vector candidate mvLXInterView (if availableFlagLXInterView is equal to 1).

The derivation process as specified in subclause G.8.5.2.1.13 is invoked with the luma location ( xP, yP ), and the variables nPSW, nPSH, as the inputs and a flag availableDV and a disparity vector mvDisp as the outputs.

Set the variable X = ( ( xP + ( ( nPSW – 1 ) >> 1 ) + ( ( mvDisp[0] + 2 ) >> 2 ) ) >>3 ) <<3

Y = ( (yP + ( ( nPSH – 1 ) >> 1 ) + ( ( mvDisp[1] + 2 ) >> 2 ) ) >>3 ) <<3

The reference layer luma location ( xRef, yRef ) is derived by

xRef = Clip3( 0, PicWidthInSamplesL – 1, X) (G‑)  
yRef = Clip3( 0, PicHeightInSamplesL – 1, Y) (G‑)

Let refCU be the coding unit that covers the luma location ( xRef, yRef ) in the view component with ViewIdx equal to refViewIdx.

The flag availableFlagLXInterView is set equal to 0.

When the variable PredMode for the coding unit refCU is equal to MODE\_SKIP or MODE\_INTER, the following ordered steps are specified:

1. Let refPredFlagLY, with Y being replaced by 0 and 1, be the variables predFlagLY for the prediction unit refPU. Let refRefIdxLY, with Y being replaced by 0 and 1, be the variables refIdxLY for the prediction unit refPU. Let refMvLY, with Y being replaced by 0 and 1, be the variables mvLY for the prediction unit refPU. Let refRefPicListLY, with Y being replaced by 0 and 1, be the reference picture list RefPicListLY for the prediction unit refPU in the view component with ViewIdx equal to refViewIdx.
2. When refPredFlagL0 is equal to 1 and the picture order count of the picture refRefPicListL0[ refRefIdxL0 ] is equal to the picture order count of the picture RefPicListLX[ refIdxLX ], the flag availableFlagLXInterView is set equal to 1 and the motion vector mvLXInterView is derived by

mvLXInterView[ 0 ] = refMvL0[ 0 ] (G‑87)  
mvLXInterView[ 1 ] = refMvL0[ 1 ] (G‑88)

1. When availableFlagLXInterView is equal to 0 and refPredFlagL1 is equal to 1 and the picture order count of the picture refRefPicListL1[ refRefIdxL1 ] is equal to the picture order count of the picture RefPicListLX[ refIdxLX ], the flag availableFlagLXInterView is set equal to 1 and the motion vector mvLXInterView is derived by

mvLXInterView[ 0 ] = refMvL1[ 0 ] (G‑89)  
mvLXInterView[ 1 ] = refMvL1[ 1 ] (G‑90)

**G.8.5.2.1.15 Derivation process for a temporal inter-view motion vector merging candidate**

This process is not invoked when multi\_view\_mv\_pred\_flag is equal to 0.

Inputs to this process are:

– a luma location ( xP, yP ) of the top-left luma sample of the current prediction unit relative to the top-left luma sample of the current picture,

– variables nPSW and nPSH specifying the width and the height, respectively, of the current prediction unit,

– a prediction list indication X.

Outputs of this process are:

– a flag availableFlagLXInterView specifying whether the inter-view motion vector candidate is available,

– a motion vector candidate mvLXInterView (if availableFlagLXInterView is equal to 1).

– a reference index refIdxLX (with X being equal to 0 or 1) specifying a reference picture in the reference picture list RefPicListLX,

The derivation process for a disparity vector as specified in subclause G.8.5.2.1.13 is invoked with the luma location ( xP, yP ), and the variables nPSW and nPSH, as the inputs and a flag availableDV and a disparity vector mvDisp and as the outputs.

Set the variable X = ( ( xP + ( ( nPSW – 1 ) >> 1 ) + ( ( mvDisp[0] + 2 ) >> 2 ) ) >>3 ) <<3

Y = ( (yP + ( ( nPSH – 1 ) >> 1 ) + ( ( mvDisp[1] + 2 ) >> 2 ) ) >>3 ) <<3

The reference layer luma location ( xRef, yRef ) is derived by

xRef = Clip3( 0, PicWidthInSamplesL – 1, X) (G‑)  
yRef = Clip3( 0, PicHeightInSamplesL – 1,Y) (G‑)

Let refCU be the coding unit that covers the luma location ( xRef, yRef ) in the view component with ViewIdx equal to refViewIdx.

The flag availableFlagLXInterView is set equal to 0.

When the variable PredMode for the coding unit refCU is equal to MODE\_SKIP or MODE\_INTER, the following ordered steps apply:

1. Let refPredFlagLY, with Y being replaced by 0 and 1, be the variables predFlagLY for the prediction unit refPU. Let refRefIdxLY, with Y being replaced by 0 and 1, be the variables refIdxLY for the prediction unit refPU. Let refMvLY, with Y being replaced by 0 and 1, be the variables mvLY for the prediction unit refPU. Let refRefPicListLY, with Y being replaced by 0 and 1, be the reference picture list RefPicListLY for the prediction unit refPU in the view component with ViewIdx equal to refViewIdx.
2. When refPredFlagLX is equal to 1, the following apply for each i from 0 to num\_ref\_idx\_lX\_active\_minus1, inclusive
   * When availableFlagLXInterView is 0, and the picture order count of the picture refRefPicListLX[ refRefIdxLX ] is equal to the picture order count of the picture RefPicListLX[ i ], the flag availableFlagLXInterView is set equal to 1 and the following applies.

mvLXInterView[ 0 ] = refMvLX[ 0 ] (G‑)  
mvLXInterView[ 1 ] = refMvLX[ 1 ] (G‑)  
refIdxLX = i (G‑)  
IvpMvFlagLX[ xP, yP ] = 1 (G‑)  
IvpMvDispLX[ xP, yP ] = mvDisp[ 0 ] (G‑)

1. When refPredFlagLY is equal to 1 (with Y equal to 1-X), the following apply for each i from 0 to num\_ref\_idx\_lX\_active\_minus1, inclusive,
   * When availableFlagLXInterView is 0, and the picture order count of the picture refRefPicListLY[ refRefIdxLY ] is equal to the picture order count of the picture RefPicListLX[ i ], the flag availableFlagLXInterView is set equal to 1 and the following applies.

mvLXInterView[ 0 ] = refMvLY[ 0 ] (G‑)  
mvLXInterView[ 1 ] = refMvLY[ 1 ] (G‑)  
refIdxLX = i (G‑)  
IvpMvFlagLX[ xP, yP ] = 1 (G‑)  
IvpMvDispLX[ xP, yP ] = mvDisp[ 0 ] (G‑)