J.8.3.1.4 Derivation process for the disparity vector and the inter-view reference

Inputs to this process are depth reference view component depthPic, the location of a top-left sample ( dbx1, dby1 ) of a partition and the listSuffixFlag.

Outputs of this process are a picture InterViewPic, an offset vector dv and a variable InterViewAvailable

Set InterViewAvailable equal to 0.

The following applies to derive an inter-view reference picture or inter-view only reference picture, InterViewPic, with X set to 1 when listFuffixFlag is 1 or 0 otherwise:

for( cIdx = 0;cIdx<num\_ref\_idx\_l0\_active\_minus1 + 1 && !InterViewAvailable; cIdx ++)  
 if ( view order index of RefPicList0[ cIdx ] is equal to 0) {  
 InterViewPic = RefPicList0[ cIdx ]  
 InterViewAvailable = 1  
 }

When InterViewAvailable is equal to 1, the following steps apply in order.

– The process specified in subclause  is invoked with mbPartIdx set equal to 0, subMbPartIdx set equal to 0, currSubMbType set equal to "na", and listSuffixFlag set equal to 0 as input and with reference indices refIdxCandL0[ i ] and the motion vectors mvCandL0[ i ] as outputs with i equal to 0, 1, and 2 corresponding to neighbouring partition A, B, and C, respectively.

– The process specified in subclause  is invoked with mbPartIdx set equal to 0, subMbPartIdx set equal to 0, currSubMbType set equal to "na", and listSuffixFlag set equal to 1 as input and with reference indices refIdxCandL1[ i ] and the motion vectors mvCandL1[ i ] as outputs with i equal to 0, 1, and 2 corresponding to neighbouring partition A, B, and C, respectively.

– Set the dv and DvAvailable as follows

for( cIdx = 0;cIdx<3 && !DvAvailable; cIdx ++) {  
 if (view order index of RefPicList0[ refIdxCandLX[ cIdx ] ] is equal to 0 ) {  
 dv[ 0 ] = mvCandLX[ cIdx ][ 0 ]  
 dv[ 1 ] = mvCandLX[ cIdx ][ 1 ]  
 DvAvailable = 1  
 }

– When DvAvailable is equal to 0, the following apples.

~~depth = depthPic[dbx1 +(partWidth << 1) -1, dby1+ (partHeight << 1) -1 ]~~

depth = INT\_MIN

for(j=0; j<partHeight; j+=(partHeight-1))

for(i=0; i<partWidth; i+=(partWidth-1)) {

if( depthPic[ dbx1+ i, dby1+j ] > depth )

depth = depthPic[ dbx1+ i, dby1+j ]

}

index = ViewIdTo3DVAcquisitionParamIndex( view\_id of the current view )  
refIndex = ViewIdTo3DVAcquisitionParamIndex( view\_id of the InterViewPic )

dv[ 0 ] = 0  
dv[ 1 ] = Disparity( NdrInverse[depth], ZNear[ dps\_id, index ], ZFar[dps\_id, index ],  
 FocalLengthX[dps\_id, index ], AbsTX[ index ] – AbsTX[ refIndex ] )