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

……

……

6. The merging candidate list, extMergeCandList, is constructed as follows:

i = 0

if( availableFlagT )

extMergeCandList[ i++ ] = T

if( availableFlagD )

extMergeCandList[ i++ ] = D

if( availableFlagIvMC && ( !availableFlagT | | differentMotion( T, IvMC ) ) )

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 (I 109)

if( availableFlagIvDC && ( !availableFlagA1 | | differentMotion( A1, IvDC ) ) &&

( !availableFlagB1 | | differentMotion( B1, IvDC ) ) && ( i < ( 5 + NumExtraMergeCand ) ) )

extMergeCandList[ i++ ] = IvDC

~~if( availableFlagVSP && !ic\_flag && iv\_res\_pred\_weight\_idx = = 0 &&~~

~~i < ( 5 + NumExtraMergeCand ) )~~

~~extMergeCandList[ i++ ] = VSP~~

if( availableFlagA0 && i < ( 5 + NumExtraMergeCand ) )

extMergeCandList[ i++ ] = A0

if( availableFlagB2 && i < ( 5 + NumExtraMergeCand ) )

extMergeCandList[ i++ ] = B2

if( availableFlagIvMCShift && i < ( 5 + NumExtraMergeCand ) &&

( !availableFlagIvMC | | differentMotion( IvMC, IvMCShift ) ) )

extMergeCandList[ i++ ] = IvMCShift

if( availableFlagVSP && !ic\_flag && iv\_res\_pred\_weight\_idx = = 0){

j = 4

while ( j < MaxNumMergeCand ){

extMergeCandList[j] = extMergeCandList[j-1]

}

extMergeCandList[3] = VSP

}

……

……

~~10. The derivation process for a view synthesis prediction flag as specified in subclause I.8.5.3.2.17 is invoked with the luma location ( xCb, yCb ), the luma location ( xPb, yPb ), the variables nPbW and nPbH, the merge candidate indicator N as the inputs, and the output is the mergeCandIsVspFlag.~~

11. The variable vspModeFlag is derived as specified in the following:

vspModeFlag = ( view\_synthesis\_pred\_flag[ layerId ] ) && (merge\_idx[xPb][yPb] = = 3) &&

(!ic\_flag && iv\_res\_pred\_weight\_idx = = 0)

~~vspModeFlag = mergeCandIsVspFlag && !ic\_flag &&~~

~~( iv\_res\_pred\_weight\_idx = = 0 ) && availableFlagVSP~~ (I 111)

**I.8.5.3.2.15 Derivation process for the shifted disparity merging candidate**

……

……

The variable availableFlagIvDCShift is set equal to 0 and for i in the range of 0 to numMergeCand 1, inclusive, the following applies:

– The variable N is set equal to mergeCandList[ i ].

~~– The derivation process for a view synthesis prediction flag as specified in subclause I.8.5.3.2.17 is invoked with the luma location ( xPb, yPb ), the variables nPbW and nPbH, the merge candidate indicator N as inputs, and the output is the mergeCandIsVspFlag.~~

– When availableFlagIvDCShift is equal to 0 and availableFlagN is equal to 1, the candidate N is not equal to IvMC or IvDC, ~~and mergeCandIsVspFlag is not equal to 0,~~ predFlagL0N is equal to 1 and ViewIdx( RefPicList0[ refIdxL0N ] ) is not equal to ViewIdx, the following applies:

– availableFlagIvDCShift is set equal to 1

– predFlagLXIvDCShift is set equal to predFlagLXN, ( with X being replaced by 0 and 1 )

– refIdxLXIvDCShift is set equal to refIdxLXN, ( with X being replaced by 0 and 1 )

– mvL0IvDCShift[ 0 ] is set equal to mvL0N[ 0 ] + 4

– mvL0IvDCShift[ 1 ] is set equal to ( view\_synthesis\_pred\_flag[ nuh\_layer\_id ] ? 0 : mvL0N[ 1 ] )

– mvL1IvDCShift = mvL1N

……

……

**I.8.5.3.2.17 Derivation process for a view synthesis prediction flag**

~~Inputs to this process are:~~

* ~~a luma location ( xCb, yCb ) of the top-left sample of the current luma coding block relative to the top-left luma sample of the current picture,~~
* ~~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 merge candidate indicator N, specifying the merge candidate.~~

~~Outputs of this process are:~~

* ~~a variable mergeCandIsVspFlag specifying, whether the merge candidate is a view synthesis prediction merge candidate.~~

1. ~~The variable mergeCandIsVspFlag is derived as specified in the following:~~

* ~~If N is equal to VSP, mergeCandIsVspFlag is set equal to 1,~~
* ~~Otherwise, if N is equal to A~~~~1~~~~, B~~~~1~~~~,~~~~B~~~~0~~~~,~~~~A~~~~0~~~~,~~~~or B~~~~2~~~~,~~~~the following applies:~~ 
  + ~~The luma position ( xN, yN ) is specified in Table I‑9 depending on N.~~
  + ~~If one of the following conditions is true, the variable mergeCandIsVspFlag is set equal to VspModeFlag[ xN ][ yN ].~~
    - ~~N is equal to A~~~~1~~ ~~or A~~~~0~~
    - ~~N is equal to B~~~~0~~~~,~~~~B~~~~1~~~~,~~~~or B~~~~2~~ ~~and ( yN  >>  Log2CtbSizeY ) is equal to ( yCb >> Log2CtbSizeY )~~
  + ~~Otherwise, ,mergeCandIsVspFlag is set equal to 0.~~

~~Otherwise, mergeCandIsVspFlag is set equal to 0.~~

~~Table I‑9 – Specification of xN and yN depending on N~~

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **~~N~~** | ~~A~~~~1~~ | ~~B~~~~1~~ | ~~B~~~~0~~ | ~~A~~~~0~~ | ~~B~~~~2~~ |
| **~~xN~~** | ~~xPb − 1~~ | ~~xPb + nPbW − 1~~ | ~~xPb + nPbW~~ | ~~xPb − 1~~ | ~~xPb − 1~~ |
| **~~yN~~** | ~~yPb + nPbH − 1~~ | ~~yPb − 1~~ | ~~yPb − 1~~ | ~~yPb + nPbH~~ | ~~yPb − 1~~ |