All the changes are highlighted in green

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

This process is only invoked when MergeFlag[ xPb ][ yPb ] is equal to 1, where ( xPb, yPb ) specify the top-left sample of the current luma prediction block relative to the top-left luma sample of the current picture.

* ….

The motion vectors mvL0 and mvL1, the reference indices refIdxL0 and refIdxL1, and the prediction utilization flags predFlagL0 and predFlagL1 are derived by the following ordered steps:

* 1. ……
  2. ……
  3. ……
  4. Depending on view\_synthesis\_pred\_flag[ nuh\_layer\_id ], PartMode and DispAvailabilityIdc[ xPb ][ yPb ], the following applies:
     + If view\_synthesis\_pred\_flag[ nuh\_layer\_id ] is equal to 0 or PartMode is not equal to PART\_2Nx2N or DispAvailabilityIdc[ xPb ][ yPb ] is equal to DISP\_NONE, the flag availableFlagVSP is set equal to 0.
     + Otherwise (view\_synthesis\_pred\_flag[ nuh\_layer\_id ] is equal to 1), the derivation process for a view synthesis prediction merge candidate as specified in subclause I.8.5.3.2.13 is invoked with the luma locations ( xPb, yPb ) and the variables nPbW and nPbH as inputs, and the outputs are the availability flag availableFlagVSP, the reference indices refIdxL0VSP and refIdxL1VSP, the prediction list utilization flags predFlagL0VSP and predFlagL1VSP, and the motion vectors mvL0VSP and mvL1VSP.

I.8.5.3.2.13 Derivation process for a view synthesis prediction merge candidate

This process is only invoked when PartMode is equal to PART\_2Nx2N.

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

I.8.5.5.2 Derivation process for a disparity sample array

Inputs to this process are:

……

* Otherwise (partIdc is equal to 2), the following applies:
  + ~~The variable minSubBlkSizeFlag is derived as specified in the following:~~ 
    - 1. ~~minSubBlkSizeFlag = ( nPSW % 8 != 0) | | ( nPSH % 8  != 0 ) (‑287)~~
  + ~~Depending on the value of minSubBlkSizeFlag, the following applies:~~
    - ~~If minSubBlkSizeFlag is equal to 1, the following applies:~~ 
      * 1. ~~horSplitFlag = ( nPSH % 8 != 0 ) (‑288)~~
    - ~~Otherwise (minSubBlkSizeFlag is equal to 0), the following applies:~~ 
      * 1. xP0 = Clip3( 0, pic\_width\_in\_luma\_samples − 1, xTL ) (I‑289)
        2. yP0 = Clip3( 0, pic\_height\_in\_luma\_samples − 1, yTL ) (I‑290)
        3. xP1 = Clip3( 0, pic\_width\_in\_luma\_samples − 1, xTL + nPSW − 1 ) (I‑291)
        4. yP1 = Clip3( 0, pic\_height\_in\_luma\_samples − 1, yTL + nPSH  − 1 ) (I‑292)

horSplitFlag = ( refDepPels[ xP0 ][ yP0 ] < refDepPels[ xP1 ][ yP1 ] )  
 = = ( refDepPels[ xP1 ][ yP0 ] < refDepPels[ xP0 ][ yP1] ) ) (I‑293)

* + The variables nSubBlkW and nSubBlkH are modified as specified in the following:
    - 1. nSubBlkW = horSplitFlag ? 8 : 4 (I‑294)

nSubBlkH = horSplitFlag ? 4 : 8