The modifications related to clipping alignment are highlighted in Green

The modifications related to reference picture selection are highlighted in yellow

I.8.5.3.3.7 Bilinear sample interpolation and residual prediction process

The process is only invoked if res\_pred\_flag is equal to 1.

Inputs to this process are:

* a luma location ( xCb, yCb ) specifying the top-left sample of the current luma coding block relative to the top left luma sample of the current picture,
* a luma location ( xBl, yBl ) specifying the top-left sample of the current luma prediction block relative to the top-left sample of the current luma coding block,
* a variable nCbS specifying the size of the current luma coding block,
* two variables nPbW and nPbH specifying the width and the height of the current luma prediction block,
* the prediction list utilization flags, predFlagL0 and predFlagL1,
* the prediction list indication X,
* the prediction list utilization flag predFlagLX,
* the reference index refIdxLX,
* the motion vectors mvLX, mvCLX .

Outputs of this process are:

* the (nPbW)x(nPbH) array predSamplesLXL,
* the (nPbW / 2)x(nPbH / 2) arrays predSamplesLXCb and predSamplesLXCr.

If ( DiffPicOrderCnt( RefPicListX[ refIdxLX ] , RefPicList1[ RpRefIdxL1 ] ) < DiffPicOrderCnt( RefPicListX[ refIdxLX ] , RefPicList0[ RpRefIdxL0 ] ) )

Z =1

Otherwise,

Z=0;

The location ( xP, yP ) is derived by:

* 1. xP = xCb + xBl (‑237)
  2. yP = yCb + yBl (‑238)

The variable ivRefFlag is set equal to ( DiffPicOrderCnt( currPic, RefPicListX[ refIdxLX ] ) = =  0 ), and the variable availFlag is set equal to 0.

Depending on ivRefFlag and RpRefIdxLX, the following applies:

* If ivRefFlag is equal to 0 and RpRefIdxL~~X~~ Z is not equal to −1, the variable availFlag is set equal to 1, the variable refIdxLX is set equal to RpRefIdxLX and the residual prediction motion vector scaling process as specified in subclause is invoked with the prediction list utilization variable equal to ~~X~~ Z, the motion vector mvLX, and the RefPicListX[ refIdxLX ] and as inputs and modified mvLX as output.
* Otherwise, when ivRefFlag is equal to 1, the following applies:
  + The derivation process for a motion vector from a reference block for residual prediction as specified in subclause is invoked with ( xP, yP ), nPbW and nPbH, RefPicListX[ refIdxLX ], and mvLX as inputs, and availFlag, motion vector mvT and prediction list utilization variable Y as outputs.
  + When availFlag is equal to 0 and RpRefIdxLX is not equal to −1, availFlag is set equal to 1, mvT is set equal to (0, 0), Y is set equal to ~~X~~ 0.

The motion vector mvCLX is set equal to mvLX.

The arrays predSamplesLXL, predSamplesLXCb, and predSamplesLXCr  are derived as specified in the following:

* The reference picture consisting of an ordered two-dimensional array refPicLXL of luma samples and two ordered two-dimensional arrays refPicLXCb and refPicLXCr of chroma samples is derived by invoking the process specified in subclause 8.5.2.2.1 with currRefIdx as input.
* The arrays predSamplesLXL, predSamplesLXCb, and predSamplesLXCr are derived by invoking the bilinear sample interpolation process specified in subclause I.8.5.3.3.7.1 with the luma locations ( xCb, yCb ), ( xBl, yBl ), , the luma prediction block width nPbW, the luma prediction block height nPbH,, the motion vectors mvLX, mvCLX, and the reference arrays with refPicLXL, refPicLXCb and refPicLXCr. as inputs.

When availFlag is equal to 1 and iv\_res\_pred\_weight\_idx is not equal to 0, the following applies:

* Depending on ivRefFlag, the variables rpPic, rpRefPic, mvRp and curRefIdx are derived as specified in the following:
  + If ivRefFlag is equal to 0, the following applies:
    - Let rpPic be the picture with PicOrderCnt( rpPic ) equal to PicOrderCntVal and ViewIdx equal to RefViewIdx[ xP ][ yP ].
    - Let rpRefPic be the picture with PicOrderCnt( rpRefPic ) equal to RefPicList~~X~~Z [ RpRefIdxL~~X~~Z ] ) and ViewIdx equal to RefViewIdx[ xP ][ yP ],
    - The variable mvRp is set equal to MvDisp[ xP ][ yP ].
    - The variable curRefIdx is set equal to RpRefIdxL~~X~~Z.
  + Otherwise (ivRefFlag is equal to 1), the following applies:
    - Let rpPic be the picture RefPicListY[ RpRefIdxLY ]. [Ed. (CY): here the interaction with F0105 needs to be further studied.]
    - Let rpRefPic be the picture with PicOrderCnt( rpRefPic ) equal to PicOrderCnt( rpPic ) and ViewIdx equal to RefViewIdx[ xP ][ yP ]
    - The variable mvRp is set equal to mvT.
    - The variable currRefIdx is set equal to RpRefIdxLY.
* The arrays rpSamplesLXL, rpSamplesLXCb, and rpSamplesLXCr are derived as specified in the following:
  + Let the reference picture sample arrays rpPicLXL, rpPicLXCb, and rpPicLXCr corresponding to decoded sample arrays SL, SCb, SCr derived in subclause 8.7 for the previously-decoded picture rpPic.
  + The arrays rpSamplesLXL, rpSamplesLXCb, and rpSamplesLXCr are derived by invoking the bilinear sample interpolation process specified in subclause I.8.5.3.3.7.1 with the luma locations ( xCb, yCb ) and ( xBl, yBl ), the luma prediction block width nPbW, the luma prediction block height nPbH, the motion vectors mvLX equal to mvRp and mvCLX equal to mvRp, and the reference arrays with rpPicLXL, rpPicLXCb and rpPicLXCr as inputs.
* The arrays rpRefSamplesLXL, rpRefSamplesLXCb, and rpRefSamplesLXCr are derived as specified in the following:
  + Let the reference picture sample arrays rpRefPicLXL, rpRefPicLXCb, and rpRefPicLXCr corresponding to decoded sample arrays SL, SCb, SCr derived in subclause 8.7 for the previously-decoded picture rpRefPic.
  + The arrays rpRefSamplesLXL, rpRefSamplesLXCb, and rpRefSamplesLXCr are derived by invoking the bilinear sample interpolation process specified in subclause with the luma locations ( xCb, yCb ), ( xBl, yBl ), the luma prediction block width nPbW, the luma prediction block height nPbH,, the motion vector mvLX equal to ( mvLX + mvRp ) and the motion vector mvCLX equal to ( mvCLX + mvRp ), and the reference arrays with rpRefPicLXL, rpRefPicLXCb and rpRefPicLXCr as inputs.
* The variable shiftVal is set equal to ( iv\_res\_pred\_weight\_idx − 1 ).
* The modified prediction samples predSamplesLXL[ x ][ y ] with x = 0..( nPbW ) − 1 and y = 0..( nPbH ) − 1 are derived as specified in the following:
  1. predSamplesLXL[ x ][ y ] = ( Clip3( 0, ( 1 << bitDepth ) − 1,predSamplesLXL[ x ][ y ] +   
      ( ( rpSamplesLXL[ x ][ y ] − rpRefSamplesLXL[ x ][ y ] )  >>  shiftVal ) ) ) (I‑239)
* The modified prediction samples predSamplesLXCb[ x ][ y ] with x = 0..( nPbW /2 ) − 1 and y = 0..( nPbH /2 )−1 are derived as specified in the following:
  1. predSamplesLXCb[ x ][ y ] = ( Clip3( 0, ( 1 << bitDepth ) − 1,predSamplesLXCb[ x ][ y ] +   
      ( ( rpSamplesLXCb[ x ][ y ] − rpRefSamplesLXCb[ x ][ y ] )  >>  shiftVal ) ) ) (I‑240)
* The modified prediction samples predSamplesLXCr[ x ][ y ] with x = 0..( nPbW /2 ) − 1 and y = 0..( nPbH /2 ) − 1 are derived as specified in the following:
  1. predSamplesLXCr[ x ][ y ] = ( Clip3( 0, ( 1 << bitDepth ) − 1,predSamplesLXCr[ x ][ y ] +   
      ( ( rpSamplesLXCr[ x ][ y ] − rpRefamplesLXCr[ x ][ y ] )  >>  shiftVal ) ) ) (I‑241)