|  |  |
| --- | --- |
| **Joint Collaborative Team on 3D Video Coding Extension Development**  **of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11**  7th Meeting: San Jose, USA, 9 Jan –17 Jan. 2014 | Document: JCT3V- G 0143 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *Title:* | **3D-CE5 related: On** **neighbouring reference pixel selection for depth intra coding** | | | |
| *Status:* | Input Document | | | |
| *Purpose:* | Proposal | | | |
| *Author(s) or Contact(s):* | Zhouye Gu1  Jianhua Zheng2  Nam Ling1  Philipp Zhang2 | Tel: +1-206-816-2367 Email: zgu@scu.edu  zhengjianhua@hisilicon.com | |  |
| *Source:* Santa Clara University1 and HiSilicon Technologies2 | | |  | |

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# Abstract

This contribution proposes an improvement of reference pixel selection for depth intra coding methods of both conventional intra prediction modes and new intra prediction modes in 3D-HEVC. It is reported that 0.1% BD-rate saving is achieved for synthesized views under Intra-only test condition and 0.1% BD-rate saving is achieved for total bitrate and synthesized views under Common Test Condition.

# Introduction

In 3D-HEVC, partition-based methods are applied for depth map intra coding. With partition-based depth map intra coding methods, a depth block is partitioned by two parts, i.e., P0 and P1 as illustrated in Figure 1, and each part is represented by a single DC value. Furthermore, the DC value (DC0 and DC1) for each partition (P0 and P1) can be further compensated by a delta DC/residual value which is explicitly signalled in the bitstream.

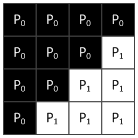
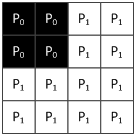
 

Figure 1: Examples of depth block partitioning[2]

## Partition DC prediction in current 3D-HEVC [3]

The DC value of each partition is predicted using one or two reconstructed neighboring reference samples depending on the partition pattern as shown in Figure 2[3]. The reconstructed neighboring reference samples are denoted by pi,j, where i = −1, j = −1..7 and i = 0..7, j = −1 in this example.

Assume the partition value of the left-top sample (c0,0) is X, where X= 0 or 1, given the partition pattern *bPattern*x,y, where x = 0..*N* − 1, y = 0..*N*−1, the predicted DC values, i.e., DC0 and DC1, are derived by follows:

Set bT = (*bPattern*0,0 != *bPattern*N-1,0)? 1 : 0; set bL = (*bPattern*0,0 != *bPattern*0,N-1)? 1 : 0

If bT equals bL

* DCX = (p-1,0 + p0,-1)>>1
* DC1-X = bL ? (p-1,N-1 + pN-1,-1)>>1 : 2B-1

Otherwise

* DCX = bL ? p(N-1)>>1,-1 : p-1, (N-1)>>1
* DC1-X = bL ? p-1,N-1 : pN-1,-1

(a) Case 1 (b) Case 2

(c) Case 3 (d) Case 4

Figure 2: Selection of reference samples for difference partition pattern cases. [3]

However, if the reference pixels in the top-right and left-bottom blocks are available, they can also be used for DC prediction.

On the other hand, for conventional HEVC intra prediction modes, neighbouring reference pixels are filtered before intra prediction as depicted in “8.4.4.2.3 Filtering process of neighbouring samples” of the specification [4]. However, for the depth map intra coding, it is desirable to keep the sharp slope of neighbouring samples for better view synthesis.

Therefore, in this proposal, we propose a new neighbouring reference pixel selection algorithm to improve depth map intra coding.

# Proposed solution

For Case 1 in Figure 2 (a), given the block size equal to NxN, based on the availability of neighboring reference pixels, the proposed method predicts the bottom-right part of Case 1 as shown in Figure 3. Given the block size equal to NxN, we check if the value of reference pixels (p-1,2\*N-1) and (p2\*N-1,-1) are available. If they are both available, we will calculate the absolute value of difference between (p-1,2\*N-1) and (p-1,0) as abs(p-1,2\*N-1 - p-1,0) and the absolute value of difference between (p2\*N-1,-1) and (p0,-1) as abs(p2\*N-1,-1 - p0,-1). Then if abs(p-1,2\*N-1 - p-1,0) is larger than abs(p2\*N-1,-1 - p0,-1), the value of (p-1,2\*N-1) is used as the DC predictor of the right-bottom partition, while if abs(p-1,2\*N-1 - p-1,0) is smaller than or equal to abs(p2\*N-1,-1 - p0,-1), the value of (p2\*N-1,-1) is used as the DC predictor of the right-bottom partition.

We describe this DC estimation algorithm under Case 1 (in Figure 2 (a)) as follows:

If p-1,2\*N-1 and p2\*N-1,-1 are both available

DC1-X = abs(p-1,2\*N-1 - p-1,0) > abs(p2\*N-1,-1 - p0,-1) ? p-1,2\*N-1 : p2\*N-1,-1

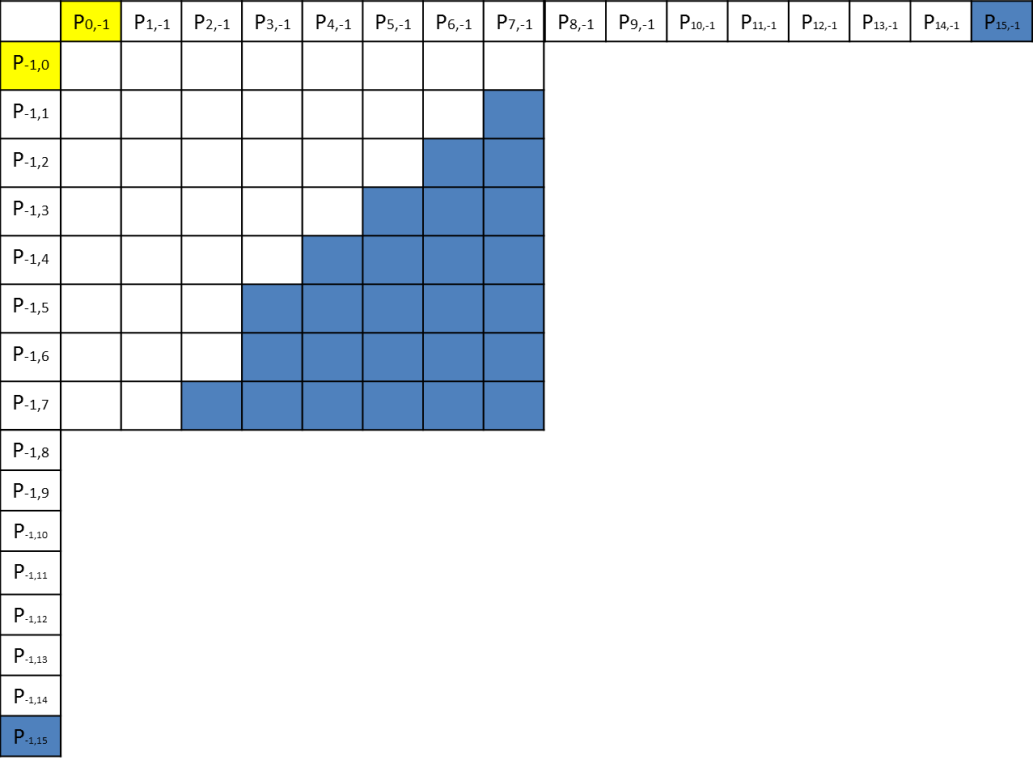


Figure 3. Selection of reference samples from neighboring

Moreover, if the conventional intra prediction modes (mode 0 to mode 34) are used for depth map intra prediction, when the PU size is smaller than 16, the filtering process of neighbouring samples described in 8.4.4.2.3 of the specification [4] will be skipped.

# Experimental results

The proposed method is implemented on top of HTM-9.0r1, and simulations were performed under both “common test condition” [1] and “Intra-only” [2] configurations.

The results are summarized in Table 1 and Table 2. As it is reported, the proposed method achieves -0.1% BD-rate coding gain on total bitrate and synthesized views under CTC and -0.1% BD-rate coding gain on synthesized views under “Intra-only”.

Table 1: BD rate results for 3-view case under CTC



Table 2: BD rate results for 3-view case under Intra-only



# References

1. D. Rusanovskyy, K. Müller, A. Vetro, “Common Test Conditions of 3DV Core Experiments,” JCT3V-F1100, Geneva, Switzerland, 25 Oct –1 Nov. 2013
2. F. Jäger, “Description of Core Experiment 5 (CE5) on Depth Intra Coding,” JCT3V-E1105, Vienna, AT, 27 July – 2 Aug. 2013.
3. X. Zhao, L. Zhang, Y. Chen and M. Karczewicz “CE6.h related: Simplified DC predictor for depth intra modes,” JCT3V-D0183, Incheon, KR, 20–26 Apr. 2013
4. Benjamin Bross, W.-J. Han, Jens-Rainer Ohm, G. J. Sullivan, Ye-Kui Wang, and Thomas Wiegand: “High Efficiency Video Coding (HEVC) text specification draft 10 (For FDIS & Final Call)”, JCT-VC, Doc. JCTVC-L1003, Geneva, Switzerland, January 2013.

# Change in WD

**H.8.4.4.2.11 Depth partition value derivation and assignment process**

– the neighbouring samples p[ x ][ y ], with x = −1, y = −1..nTbS \* 2 − 1 and x = 0..nTbS \* 2 − 1, y = −1,

– a binary array partitionPattern[ x ][ y ], with x, y =0..nTbS − 1, specifying a partitioning of the prediction block in a partition 0 and a partition 1.

– a variable nTbS specifying the transform block size,

– a flag dcOffsetAvailFlag, specifying whether DC Offset values are available,

– a flag intraChainFlag, specifying whether the current intra prediction mode is equal to INTRA\_CHAIN

– the variables dcOffsetP0 and dcOffsetP1, specifying the DC offsets for the block partitions

Output of this process is:

– the predicted samples predSamples[ x ][ y ], with x, y = 0..nTbS − 1.

The variables vertEdgeFlag and horEdgeFlag are derived as specified in the following:

vertEdgeFlag = ( partitionPattern[ 0 ][ 0 ] != partitionPattern[ nTbS − 1 ][ 0 ] ) ? 1 : 0 (H 45)

horEdgeFlag = ( partitionPattern[ 0 ][ 0 ] != partitionPattern[ 0 ][ nTbS − 1 ] ) ? 1 : 0 (H 46)

The variables dcVal0 and dcVal1 are derived as specified in the following:

If vertEdgeFlag is equal to horEdgeFlag, the following applies:

dcValBR = horEdgeFlag ?

( ( p[ −1 ][ nTbS − 1 ] + p[ nTbS − 1 ][ −1 ] ) >> 1 ) : ( 1 << ( BitDepthY − 1 ) ) (H 47)

dcValLT = ( p[ −1 ][ 0 ] + p[ 0 ][ −1 ] ) >> 1 (H‑48)

* + If horEdgeFlag is equal to 0

if p [ nTbS\*2 − 1 ][ -1 ] is available and p [ -1 ][ nTbS\*2 − 1 ] is available

dcValBR = abs(p [ 0 ][ -1 ] - p [ nTbS\*2 − 1 ][ -1 ]) > abs(p [ -1 ][ 0 ] - p [ -1 ][ nTbS\*2 − 1 ]) ?

p [ nTbS\*2 − 1 ][ -1 ] : p [ -1 ][ nTbS\*2 − 1 ]

Otherwise ( horEdgeFlag is not equal to vertEdgeFlag), the following applies:

* + 1. dcValBR = horEdgeFlag ? p[ −1 ][ nTbS − 1 ] : p[ nTbS − 1 ][ −1 ] (H‑49)
    2. dcValLT = horEdgeFlag ? p[ ( nTbS – 1 ) >> 1 ][ −1 ] : p[ −1 ][ ( nTbS – 1 ) >> 1 ] (‑50)

**H.8.4.4.2.1 General intra sample prediction**

Inputs to this process are:

* a sample location ( xTbCmp, yTbCmp ) specifying the top-left sample of the current transform block relative to the top left sample of the current picture,
* a variable predModeIntra specifying the intra prediction mode,
* a variable nTbS specifying the transform block size,
* a variable cIdx specifying the colour component of the current block.

Output of this process is the predicted samples predSamples[ x ][ y ], with x, y = 0..nTbS − 1.

The nTbS \* 4 + 1 neighbouring samples p[ x ][ y ] that are constructed samples prior to the deblocking filter process, with x = −1, y = −1..nTbS \* 2 − 1 and x = 0..nTbS \* 2 − 1, y = −1, are derived as follows:

* The neighbouring location ( xNbCmp, yNbCmp ) is specified by:
  + - ( xNbCmp, yNbCmp ) = ( xTbCmp + x, yTbCmp + y ) (8 27)
* The current luma location ( xTbY, yTbY ) and the neighbouring luma location ( xNbY, yNbY ) are derived as follows:

( xTbY, yTbY ) = ( cIdx  = =  0 ) ? ( xTbCmp, yTbCmp ) : ( xTbCmp  <<  1, yTbCmp  <<  1 ) (8 28)

( xNbY, yNbY ) = ( cIdx  = =  0 ) ? ( xNbCmp, yNbCmp ) : ( xNbCmp  <<  1, yNbCmp  <<  1 ) (8 29)

* The availability derivation process for a block in z-scan order as specified in subclause 6.4.1 is invoked with the current luma location ( xCurr, yCurr ) set equal to ( xTbY, yTbY ) and the neighbouring luma location ( xNbY, yNbY ) as inputs, and the output is assigned to availableN.
* Each sample p[ x ][ y ] is derived as follows:

If one or more of the following conditions are true, the sample p[ x ][ y ] is marked as "not available for intra prediction":

The variable availableN is equal to FALSE.

CuPredMode[ xNbY ][ yNbY ] is not equal to MODE\_INTRA and constrained\_intra\_pred\_flag is equal to 1.

Otherwise, the sample p[ x ][ y ] is marked as "available for intra prediction" and the sample at the location ( xNbCmp, yNbCmp ) is assigned to p[ x ][ y ].

When at least one sample p[ x ][ y ] with x = −1, y = −1..nTbS \* 2 − 1 and x = 0..nTbS \* 2 − 1, y = −1 is marked as "not available for intra prediction", the reference sample substitution process for intra sample prediction in subclause 8.4.4.2.2 is invoked with the samples p[ x ][ y ] with x = −1, y = −1..nTbS \* 2 − 1 and x = 0..nTbS \* 2 − 1, y = −1, nTbS, and cIdx as inputs, and the modified samples p[ x ][ y ] with x = −1, y = −1..nTbS \* 2 − 1 and x = 0..nTbS \* 2 − 1, y = −1 as output.

Depending on the value of predModeIntra, the following ordered steps apply:

1. When cIdx is equal to 0 and predModeIntra is in the range of 0 to 34

– If DepthFlag is equal to 0 or nTbS is greater than 8

the filtering process of neighbouring samples specified in subclause 8.4.4.2.3 is invoked with the sample array p and the transform block size nTbS as inputs, and the output is reassigned to the sample array p.

1. The intra sample prediction process according to predModeIntra applies as follows:

– If predModeIntra is equal to INTRA\_PLANAR, the corresponding intra prediction mode specified in subclause 8.4.4.2.4 is invoked with the sample array p and the transform block size nTbS as inputs, and the output is the predicted sample array predSamples.

– Otherwise, if predModeIntra is equal to INTRA\_DC, the corresponding intra prediction mode specified in subclause 8.4.4.2.5 is invoked with the sample array p, the transform block size nTbS, and the colour component index cIdx as inputs, and the output is the predicted sample array predSamples.

– Otherwise, if predModeIntra is in the range of INTRA\_ANGULAR2..INTRA\_ANGULAR34, the corresponding intra prediction mode specified in subclause 8.4.4.2.6 is invoked with the intra prediction mode predModeIntra, the sample array p, the transform block size nTbS, and the colour component index cIdx as inputs, and the output is the predicted sample array predSamples.

– Otherwise, if predModeIntra is equal to INTRA\_DMM\_WFULL, the corresponding intra prediction mode specified in subclause  is invoked with the location ( xTbY, yTbY ), the sample array p and the transform block size nTbS as the inputs and the output are the predicted sample array predSamples.

– Otherwise, if predModeIntra is equal to INTRA\_DMM\_CPREDTEX, the corresponding intra prediction mode specified in subclause  is invoked with the location ( xTbY, yTbY ), with the sample array p and the transform block size nTbS as the inputs and the output are the predicted sample array predSamples.

# Patent rights declaration(s)

**HiSilicon Technologies, Santa Clara University and Huawei Technologies may have current or pending patent rights relating to the technology described in this contribution and, conditioned on reciprocity, is prepared to grant licenses under reasonable and non-discriminatory terms as necessary for implementation of the resulting ITU-T Recommendation | ISO/IEC International Standard (per box 2 of the ITU-T/ITU-R/ISO/IEC patent statement and licensing declaration form).**