|  |  |
| --- | --- |
| **Joint Collaborative Team on Video Coding (JCT-VC)**  **of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11**  12th Meeting: Geneva, CH, 14–23 Jan. 2013 | Document: JCTVC-L0156 |

|  |  |  |  |
| --- | --- | --- | --- |
| *Title:* | **Simplification of TE5.1.5 on inter-layer intra mode prediction** | | |
| *Status:* | Input Document to JCT-VC | | |
| *Purpose:* | Proposal, Report | | |
| *Author(s) or Contact(s):* | Zhijie Zhao, Joern Ostermann Appelstr. 9a Hannover, Germany | Tel: Email: | 0049-511 7625333 [zhao@tnt.uni-hannover.de](mailto:zhao@tnt.uni-hannover.de)  ostermann@tnt.uni-hannover.de |
| *Source:* | [Leibniz Universitaet Hannover] | | |

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

# Abstract

This document presents a simplification of the inter-layer intra prediction mode coding (ILIPM) proposed in JCTVC-K0238. In the original proposal of JCTVC-K0238, inter-layer intra prediction mode coding is evaluated only if the PU size of the current block and the PU size of the co-located block in the base layer match a defined constraint. In this proposal, inter-layer intra prediction mode coding is enabled if the co-located block in the base layer is intra-encoded. The implementation was integrated into the Smuc 0.1.1 software. Compared to enhancement layer only of anchor in TE 5.1, it reports that the simplified ILIPM achieves average luma BD-rate gains of 0.6% for All Intra HEVC 2x spatial scalability, 0.4% for All Intra HEVC 1.5x spatial scalability. Moreover, it reportedly shows average luma BD-rate gains of 0.4% for All Intra HEVC 2x spatial scalability, 0.3% for All Intra HEVC 1.5x spatial scalability when Mode Dependent Coefficient Scanning (MDCS) is disabled.

# Introduction

In HEVC, there are totally 35 intra prediction modes. In the scalable extension of HEVC, since there exists high correlation between base and enhancement layers, intra prediction mode of base layer can be used to improve the coding efficiency of enhancement layer. In JCTVC-K0238 [1], a flag, called intra\_bl\_mode\_flag, was introduced before prev\_intra\_luma\_pred\_flag in the coding unit syntax to indicate whether the intra prediction mode of the current PU in enhancement layer is equal to that of the co-located PU in base layer. If intra\_bl\_mode\_flag is true, prev\_intra\_luma\_pred\_flag and rem\_intra\_luma\_pred\_mode do not need to be coded any more. Otherwise, MPM coding as in the single layer HEVE coding is performed.

# Proposed method

In the original proposal of inter-layer intra prediction mode coding (ILIPM) in JCTVC-K0238, the introduced intra\_bl\_mode\_flag is coded if the following condition is true,

( ( 1 << log2CbSize\_enh ) >> trafoDepth = = (( 1 << log2CbSize\_base ) >> trafoDepth ) << 1) (1)

This original design was based on the idea that other inter-layer prediction tools may be coupled with other inter-layer coding tools. Therefore, the PU size relationship of the base and enhancement layer PU has been considered. However, no matter what the size ratio between the enhancement and base layer PU, the intra prediction mode of the co-located PU in base layer can always be used to improve the coding efficiency of enhancement layer if the co-located PU is intra-coded. Therefore, the constraint in Eq. (1) is removed in the current proposal. Intra\_bl\_mode\_flag is encoded in CU syntax of enhancement layer. The intra prediction modes of the current PU in enhancement layer and the co-located PU in base layer will be compared during encoding. If they are equal, intra\_bl\_mode\_flag is set to true, and prev\_intra\_luma\_pred\_flag as well as rem\_intra\_luma\_pred\_mode do not need to be coded any more. Otherwise, intra\_bl\_mode\_flag is set to false and MPM coding as in the single layer HEVE coding is performed.

The proposed syntax is shown in Table 1.

Table Proposed syntax changes

|  |
| --- |
| coding\_unit( x0, y0, log2CbSize ) { |
| CurrCbAddrTS = MinCbAddrZS[ x0 >> Log2MinCbSize ][ y0 >> Log2MinCbSize ] |
| if( transquant\_bypass\_enable\_flag ) { |
| **cu\_transquant\_bypass\_flag** |
| } |
| if( slice\_type != I ) |
| **cu**\_**skip\_flag[** x0 **][** y0 **]** |
| if( cu\_skip\_flag[ x0 ][ y0 ] ) |
| prediction\_unit( x0, y0 , log2CbSize ) |
| else { |
| if( slice\_type != I ) |
| **pred\_mode\_flag** |
| if( PredMode != MODE\_INTRA | | log2CbSize = = Log2MinCbSize ) |
| **part\_mode** |
| x1 = x0 + ( ( 1 << log2CbSize ) >> 1 ) |
| y1 = y0 + ( ( 1 << log2CbSize ) >> 1 ) |
| x2 = x1 − ( ( 1 << log2CbSize ) >> 2 ) |
| y2 = y1 − ( ( 1 << log2CbSize ) >> 2 ) |
| x3 = x1 + ( ( 1 << log2CbSize ) >> 2 ) |
| y3 = y1 + ( ( 1 << log2CbSize ) >> 2 ) |
| if( PredMode = = MODE\_INTRA ) { |
| if( PartMode = = PART\_2Nx2N && pcm\_enabled\_flag &&  log2CbSize >= Log2MinIPCMCUSize &&  log2CbSize <= Log2MaxIPCMCUSize ) |
| **pcm\_flag** |
| if( pcm\_flag ) { |
| **num\_subsequent\_pcm** |
| NumPCMBlock = num\_subsequent\_pcm + 1 |
| while( !byte\_aligned( ) ) |
| **pcm\_alignment\_zero\_bit** |
| for( i = 0; i < 1 << ( log2CbSize << 1 ); i++ ) |
| **pcm\_sample\_luma**[ i ] |
| for( i = 0; i < ( 1 << ( log2CbSize << 1 ) ) >> 1; i++ ) |
| **pcm\_sample\_chroma**[ i ] |
| NumPCMBlock− − |
| } else { |
| pbOffset = ( PartMode = = PART\_NxN ) ? ( ( 1 << log2CbSize ) >> 2 ) : 0 |
| for( j = 0; j <= pbOffset; j = j + pbOffset ) |
| for( i = 0; i <= pbOffset; i = i + pbOffset ) { |
| **intra\_bl\_mode\_flag**[ x0 + i ][ y0+ j ] |
| if(!intra\_bl\_mode\_flag[ x0 + i ][ y0+ j ]) |
| **prev\_intra\_luma\_pred\_flag**[ x0 + i ][ y0+ j ] |
| } |
| for( j = 0; j <= pbOffset; j = j + pbOffset ) |
| for( i = 0; i <= pbOffset; i = i + pbOffset ) { |
| if(!intra\_bl\_mode\_flag[ x0 + i ][ y0+ j ] ){ |
| if( prev\_intra\_luma\_pred\_flag[ x0 + i ][ y0+ j ] ) |
| **mpm\_idx**[ x0 + i ][ y0+ j ] |
| Else |
| **rem\_intra\_luma\_pred\_mode**[ x0 + i ][ y0+ j ] |
| } |
| } |
| **intra\_chroma\_pred\_mode**[ x0 ][ y0 ] |
| } |
| } else { |
| if( PartMode = = PART\_2Nx2N ) |
| prediction\_unit( x0, y0 , log2CbSize ) |
| else if( PartMode = = PART\_2NxN ) { |
| prediction\_unit( x0, y0 , log2CbSize ) |
| prediction\_unit( x0, y1 , log2CbSize ) |
| } else if( PartMode = = PART\_Nx2N ) { |
| prediction\_unit( x0, y0 , log2CbSize ) |
| prediction\_unit( x1, y0 , log2CbSize ) |
| } else if( PartMode = = PART\_2NxnU ) { |
| prediction\_unit( x0, y0 , log2CbSize ) |
| prediction\_unit( x0, y2 , log2CbSize ) |
| } else if( PartMode = = PART\_2NxnD ) { |
| prediction\_unit( x0, y0 , log2CbSize ) |
| prediction\_unit( x0, y3 , log2CbSize ) |
| } else if( PartMode = = PART\_nLx2N ) { |
| prediction\_unit( x0, y0 , log2CbSize ) |
| prediction\_unit( x2, y0 , log2CbSize ) |
| } else if( PartMode = = PART\_nRx2N ) { |
| prediction\_unit( x0, y0 , log2CbSize ) |
| prediction\_unit( x3, y0 , log2CbSize ) |
| } else { /\* PART\_NxN \*/ |
| prediction\_unit( x0, y0 , log2CbSize ) |
| prediction\_unit( x1, y0 , log2CbSize ) |
| prediction\_unit( x0, y1 , log2CbSize ) |
| prediction\_unit( x1, y1 , log2CbSize ) |
| } |
| } |
| if( !pcm\_flag ) { |
| if( PredMode != MODE\_INTRA &&   !(PartMode = = PART\_2Nx2N && merge\_flag[x0][y0]) ) |
| **rqt\_root\_cbf** |
| if( !rqt\_root\_cbf ) { |
| MaxTrafoDepth = ( PredMode = = MODE\_INTRA ?   max\_transform\_hierarchy\_depth\_intra + IntraSplitFlag :   max\_transform\_hierarchy\_depth\_inter ) |
| transform\_tree( x0, y0, x0, y0, x0, y0, log2CbSize, log2CbSize, log2CbSize, 0, 0 ) |
| } |
| } |
| } |
| } |

# Experimental Results

Test sequences and coding condition as shown in [2] were used to evaluate the performance of the proposed approaches. Smuc 0.1.1 was used as the anchor. The encoding and decoding test were conducted on a Linux cluster with Intel Xeon CPU X5670, 2.93GHz. The encoder and decoder binaries were compiled with gcc 4.1.2 64-bit. Both the proposed encoder/decoder and anchor encoder/decoder were tested in the same cluster. The average performances of the proposed approaches are shown in the following tables. The detailed results can be found in the enclosed excel sheets.

Table 2 presents the performance of the simplified ILIPM in case Mode Dependent Coefficient Scanning (MDCS) is enabled. The results without MDCS are shown in Table 3.

Table Performance of the simplified ILIPM compared to anchor of TE 5.1, with MDCS

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **AI HEVC 2x** | | | **AI HEVC 1.5x** | | |
|  | Y | U | V | Y | U | V |
| Class A | -0.3% | -0.1% | 0.0% |  |  |  |
| Class B | -0.4% | -0.2% | -0.2% | -0.1% | 0.1% | 0.1% |
| **Overall (EL+BL)** | -0.3% | -0.2% | -0.2% | -0.1% | 0.1% | 0.1% |
| **Overall (EL)** | -0.6% | -0.3% | -0.2% | -0.4% | 0.2% | 0.2% |
| Enc Time[%] | 101.1% | | | 100.8% | | |
| Dec Time[%] | 99.1% | | | 98.9% | | |
| Enc Mem[%] | #DIV/0! | | | #DIV/0! | | |
| BL Match | Matched | | | Matched | | |

Table Performance of the simplified ILIPM compared to anchor of TE 5.1, without MDCS

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **AI HEVC 2x** | | | **AI HEVC 1.5x** | | |
|  | Y | U | V | Y | U | V |
| Class A | -0.2% | 0.0% | 0.1% |  |  |  |
| Class B | -0.2% | 0.0% | 0.1% | -0.1% | 0.1% | 0.1% |
| **Overall (EL+BL)** | -0.2% | 0.0% | 0.1% | -0.1% | 0.1% | 0.1% |
| **Overall (EL)** | -0.4% | 0.0% | 0.2% | -0.3% | 0.2% | 0.3% |
| Enc Time[%] | 101.0% | | | 100.2% | | |
| Dec Time[%] | 99.0% | | | 98.7% | | |
| Enc Mem[%] | #DIV/0! | | | #DIV/0! | | |
| BL Match | Matched | | | Matched | | |

# Conclusion

This contribution presents a method of inter-layer intra prediction mode coding for scalable extension of HEVC. MPM process is not performed if the intra prediction mode of current PU in enhancement layer is equal to that of the co-located PU in base layer. The proposed method not only improves the coding efficiency of enhancement layer, but also reduces the decoding time at the decoder.

# References

|  |  |
| --- | --- |
| [1] | Z. Zhao, J. Si and J. Ostermann, "Inter-layer intra prediction mode coding for the scalable extension of HEVC," JCTVC-K0238, Shanghai, 2012. |
| [2] | V. Seregin, P. Onno, S. Liu, T. Lee, C. Kim and H. Yang, "JCTVC-K1105, Description of Tool Experiment C5: Inter-layer syntax prediction using HEVC base layer," 2012. |

# Patent rights declaration(s)

**Leibniz Universitaet Hannover 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).**