|  |  |
| --- | --- |
| **Joint Collaborative Team on Video Coding (JCT-VC)**  **of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11**  13th Meeting: Incheon, KR, 18–26 Apr. 2013 | Document: JCTVC-M0298 |

|  |  |  |  |
| --- | --- | --- | --- |
| *Title:* | **Bandwidth reduction for range extension** | | |
| *Status:* | Input Document to JCT-VC | | |
| *Purpose:* | Proposal | | |
| *Author(s) or Contact(s):* | Vadim Seregin Xianglin Wang  Joel Sole Marta Karczewicz  5775 Morehouse Drive San Diego, CA 92121-1714 USA | Email: | [vseregin@qti.qualcomm.com](mailto:vseregin@qti.qualcomm.com) |
| *Source:* | Qualcomm | | |

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

# Abstract

HEVC Range Extensions 4:4:4 potential applications include consumer applications, so the complexity increase due to the 4:4:4 processing has to be taken into account. The scope of this contribution is the bandwidth increase for 4:4:4 motion estimation and compensation, which is around 37% more than the 4:2:0 case for 4×2 memory block. Specifically, in this contribution, restriction of bi-directional prediction for 8×8 chroma components is studied, while luma component still can have bi-directional prediction. This modification increases bandwidth by 9%, as opposed to 37%. The impact on coding performance is about 0.2% and 0.4% luma BD rate loss for Main-tier RA and LB, and High-tier RA and LB respectively in for both YUV422 and YUV444.

# Problem Statement

This contribution addresses bandwidth reduction issue for HEVC range extension. In real decoder implementation it is desired to reduce bandwidth requirement due to accessing reference pixels, while keeping a good tradeoff between bandwidth reduction and coding performance.

In the following analysis, bandwidth is calculated as number of reference pixels fetched from the memory and necessary for interpolation based on 4x2 and 8x2 memory blocks as described in [1]. It assumes that memory can be accessed and read based on the memory blocks, and only integer number of blocks can be fetched. Let’s denote a block size to be inter predicted using 2D interpolation with filter length *L* as *M×N*, memory block as *m×n*, then for each pixel within *M×N* block the number of pixels per fetched from the memory can be calculated as follows.

The numbers corresponding to the bandwidth worst case are summarized in the Table 1, and can be found in more details in the accompanied excel table. In the excel table, luma column corresponds to the total number of pixels calculated by above formula per *M×N* block, similarly chroma columns includes the total number of pixels for both chroma components, where interleaved chroma storage was assumed. And finally, YUV column represents the total number of fetched pixels for luma and chroma components normalized by the block size.

In current HEVC, after adoption of bi-directional restriction for 8×4 and 4×8 PU, the worst case is 8×8 bi-predicted blocks with fractional pixel motion vectors (MV) from both directions.

Table 1. Number of pixels for 2D interpolation

|  |  |  |  |
| --- | --- | --- | --- |
| Memory configuration | 4:2:0 | 4:4:4 | Ratio |
| 4×2 | 15.0 | 20.5 | 137% |
| 8×2 | 18.0 | 24.0 | 133% |

# Proposed bandwidth reduction

In this contribution, two methods are studied to reduce the bandwidth:

1. Following HEVC bandwidth reduction for 8×4 and 4×8 PUs, where bi-directional motion vector is converted to uni-directional of the RefPicList0, the same conversion is done for 8×8 PUs.
2. Bi-directional MV is converted to uni-directional of the RefPicList0 at motion compensation stage for both chroma components only, while luma component is still bi-predicted.

Bandwidth assessment data is summarised in the Table 2.

Table 2. Proposed worst case bandwidth reduction for 4:4:4

|  |  |  |  |
| --- | --- | --- | --- |
| Memory configuration | 4:4:4 | Method 1/2 | Ratio |
| 4×2 | 20.5 | 16.3 | 79% |
| 8×2 | 24.0 | 21.0 | 88% |

According to the Table 2, both methods reduce the worst case of 8×8 bi-predicted PU from 20.5 to 16.3 and from 24.0 to 21.0 for 4×2 and 8×2 memory blocks respectively, and the new worst case become either 16×8 bi-predicted PU or 8×4 uni-predicted PU depending on the memory configuration.

# Experimental results

Proposed bandwidth reduction was implemented on top of HM10.0\_RExt2.0 software and experimental results under common test conditions [2] are summarised in the next tables.

**Table 3. Experimental results for method 1.**



**Table 4. Experimental results for method 2.**



# Conclusion

In this contribution, bi-directional prediction restriction for 8×8 blocks of chroma components is studied for worst case bandwidth reduction. Addressing the worst case of 8×8 bi-prediction blocks, 21% and 12% of worst case bandwidth reduction is achieved for 4×2 and 8×2 memory blocks with luma BD-rate loss from 0.2% to 0.4% among RA and LB test cases for both YUV422 and YUV444.

# References

[1] E. François, A. Tabatabai, and E. Alshina “BoG report: Methodology for evaluating complexity of combined and residual prediction methods in SHVC”, JCTVC-L0440, Geneva, CH, Jan. 2013.

[2] D. Flynn, and K. Sharman “Common test conditions and software reference configurations for HEVC range extensions”, JCTVC-L1006, Geneva, CH, Jan. 2013.

# Draft text

## Method 1

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

…

10. When predFlagL0 is equal to 1 and predFlagL1 is equal to 1, and ( nOrigPbW + nOrigPbH ) is equal to 12 or equal to 16 and ChromaArrayType !=0, the following applies:

refIdxL1 = −1 (8‑96)

predFlagL1 = 0 (8‑97)

##### Table 7‑11 – Name association to inter prediction mode

|  |  |  |
| --- | --- | --- |
| **inter\_pred\_idc** | **Name of inter\_pred\_idc** | |
| ( nPbW + nPbH ) != 12 || ( ( nPbW + nPbH ) > 16 && ChromaArrayType !=0 ) | ( nPbW + nPbH )  == 12 || (( nPbW + nPbH )  == 16 && ChromaArrayType !=0 ) |
| 0 | PRED\_L0 | PRED\_L0 |
| 1 | PRED\_L1 | PRED\_L1 |
| 2 | PRED\_BI | na |

## Method 2

**8.5.3.3.1 General**

…

For X being each of 0 and 1, when predFlagLX is equal to 1, the following applies:

– The reference picture consisting of an ordered two-dimensional array refPicLXL of luma samples and when ChromaArrayType is not equal to 0, two ordered two-dimensional arrays refPicLXCb and refPicLXCr of chroma samples is derived by invoking the process specified in subclause 8.5.3.3.2 with refIdxLX as input.

– For X being equal to 1, and ChromaArrayType !=0, when predFlagL0 is equal to 1 and predFlagL1 is equal to 1, and ( nPbW + nPbH ) is equal to 16, the following applies:

refPicL1Cb = refPicL0Cb

refPicL1Cr = refPicL0Cr

mvCL1 = mvCL0

– The array predSamplesLXL, and when ChromaArrayType is not equal to 0, the arrays predSamplesLXCb, and predSamplesLXCr are derived by invoking the fractional sample interpolation process specified in subclause 8.5.3.3.3 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 and, when ChromaArrayType is not equal to 0, mvCLX, and the reference arrays refPicLXL, refPicLXCb, and refPicLXCr as inputs.

…

# Patent rights declaration(s)

**Qualcomm Incorporated 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).**