|  |  |
| --- | --- |
| **Joint Collaborative Team on Video Coding (JCT-VC)**  **of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11**  5th Meeting: Geneva, 16-23 March, 2011 | Document: JCTVC-E283  WG11 Number: m19810 |

|  |  |  |  |
| --- | --- | --- | --- |
| *Title:* | **CE4 Subset2: Report of Intra Coding Improvements for Slice Boundary Blocks** | | |
| *Status:* | Input Document to JCT-VC | | |
| *Purpose:* | Proposal | | |
| *Author(s) or Contact(s):* | Yongbing Lin,  Changcai Lai,  Jianhua Zheng,  Lingzhi Liu | Tel: Email: | +86-10-82836279 yblin@huawei.com laichangcai@huawei.com |
| Source: | HiSilicon Technologies Co., Ltd. | | |

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

# Abstract

This contribution presents the experimental results of Intra Coding Improvements for Slice Boundary Blocks in Core Experiment 4 (CE4), which was firstly proposed in the contribution of JCTVC-D302 at the last JCT-VC meeting.

In this document, two improvements on reference pixel processing for slice boundary blocks are proposed. Firstly, “unavailable” reference pixels for boundary blocks are padded with the value of the closest pixel to the upper-left corner from an available neighbor block, instead of being padded with a fixed DC value. Secondly, for those positive directions in UDI, “unavailable” reference pixels are generated by projecting the reference pixels of an available neighbor block. Then the generated reference pixels are further used for UDI intra prediction.

The experimental results show the proposed two improvements achieve an average BD-rate reduction of 0.6% and 0.7% with approximately the same encoding and decoding time for Intra HE and LC configurations in case of 1500bytes slice. It also shows the improved performance with 0.1%-0.3% bit-rate reduction for random access and low delay configurations.

# Introduction

Multiple slice partition has been supported by HEVC to meet the requirement of video transmission over error-prone channel. The future of multiple slice leads to large gain drop, *e.g.,* about 5% bit-rate increase over HM2.0 for all Intra coding in case of 1500bytes per slice, as shown in Table 1. The gain drop is caused by the restriction on the reference relationship between different slices.

This proposal presents reference pixel processing for slice boundary block to improve intra prediction accuracy. For slice boundary block where only one of its neighbor above and left blocks is available, the reference pixels located in its unavailable neighbor reference block are generated with the pixels from its available neighbor reference block, instead of being padded with a fixed DC value.

Table 1. Performance of Huawei’s slice implementation based on HM2.0 (1500bytes per slice)



# Reference pixel processing for slice boundary block

Slice boundary blocks lead to the gain drop due to its unavailable reference pixels. This proposal deals with the issue on how to effectively generate the unavailable reference pixels.

In current implementation of HM2.0, those unavailable reference pixels of boundary blocks are padded with DC value, *e.g.*, 128 for LC case. However the fixed value doesn’t adapt to the different content of the boundary blocks. Generating these unavailable reference pixels from its available reference block seems reasonable to improve Intra prediction accuracy.

## Reference pixel padding with the pixel closest to upper-left corner

The first improvement for reference pixel processing is to use the value of the available reference pixel closest to upper-left corner of current block. This approach has been proposed in JCTVC-D302 in last meeting [1].

Figure 1a shows the proposed padding process for the block at top boundary of a slice, where its left neighbor block is available while its upper and upper-left neighbor blocks are not available. The value of the reference pixel closest to the upper-left corner, which is located in the left neighbor block, is copied to form the upper reference pixels for Intra prediction of current block. It is noticed that the upper-left corner reference pixel is also padded. As shown in Fig. 1b, similar padding process can be applied for the block at left boundary of a slice, where the left and upper-left neighbor blocks are not available while the upper neighbor block is available.



1. Padding process for top boundary block b) Padding process for left boundary block

Fig. 1. Reference pixel padding for slice boundary blocks

For better understanding the padding process, the following paragraphs further summary the details of reference pixel padding for different cases.

*If both left and upper neighbor blocks are not available, both left and upper reference pixels of current block are padded with DC value.*

*If left neighbor block is not available while upper neighbor block is available, the left reference pixels of current block are padded with the pixel closest to upper-left corner from the upper block.*

*If left neighbor block is available while upper neighbor block is not available, the upper reference pixels of current block are padded with the pixel closest to upper-left corner from the left block.*

*If both left and upper neighbor blocks are available, no padding process is required.*

## Reference pixel generation by projecting available reference side

The second improvement for reference pixel processing is to use the reference pixels generated by projecting the available reference side in case of Intra prediction with positive directions.

As shown in Figure 2, Intra prediction direction for current block is positive direction from upper-right to below-left, and its upper neighbor block is unavailable while its left reference side is available. In this case, the left reference side can be projected to generate the upper reference pixels according to the inversed Intra prediction direction. If applicable, the last projected pixel can be extended to generate remain reference pixels required for Intra prediction. Finally, all the generated upper reference pixels are used for UDI Intra prediction with the specified Intra prediction direction.



Fig. 2. Reference pixel generation by projecting left reference side to upper side

# Experimental results

The following running environment and compiler have been used in our simulation.

|  |  |
| --- | --- |
| **Running environment** | **Compiler** |
| Intel Xeon X7560 @2.27GHz, 256G RAM  Windows Server 2003, 64 bits | VS2005 |

The proposed two improvements have been integrated into Huawei’s slice implementation which has very similar performance to HM2.0-dev-slice. The two improvements and their combination have been tested under the conditions defined by CE4, *i.e.*, LCU=64x64, LCU-aligned, 1500 bytes per slice [2][3]. The test results are summarized in the following tables. See the attached documents for details.

Table 2 and Table 3 show the coding performances of the proposed improvement 1 and improvement 2, respectively. The two improvements have reached 0.4% and 0.5% bit-rate reduction for all intra HE and LC cases with negligible increased encoding time.

Table 4 shows the performance for the combination of the proposed two improvements. The experimental results show the two proposed methods achieve an average BD-rate reduction of 0.6% and 0.7% for both all Intra HE and LC configurations. It also shows the improved performance of 0.1%-0.3% bit-rate reduction for random access and low delay configurations with almost the same encoding and decoding time.

Table 2. Performance of improvement 1 (reference pixel padding with the closest pixel)



Table 3. Performance of improvement 2 (reference pixel generation based on projecting)



Table 4. Performance for combination of improvement 1 and improvement 2



# Conclusion

The document reports the performance of two improvements on reference pixel processing for slice boundary block. The test results show that the proposed methods improve coding performance by 0.6% and 0.7% bit-rate reduction for Intra HE and LC cases with negligible increased encoding and decoding time. It’s suggested that the two improvements be introduced in the HEVC test model (HM).

# References

1. Yongbing Lin, Changcai Lai, Jianhua Zheng, etc. "Intra coding improvements for slice boundary blocks", Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, JCTVC-D302, 4th Meeting, Daegu, KR, Jan., 2011.
2. Yu-Wen Huang, Il-Koo Kim, “Description of Core Experiment 4: Slice Boundary Processing and Fine Granularity”, Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, JCTVC-D604, 4th Meeting, Daegu, KR, Jan., 2011.
3. F. Bossen., “Common conditions and software reference configurations”, Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, JCTVC-D600, 4th Meeting, Daegu, KR, Jan., 2011.

# Patent rights declaration(s)

**Huawei Technologies Co., Ltd and HiSilicon Technologies Co., Ltd. may have IPR 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).**