|  |  |
| --- | --- |
| **Joint Collaborative Team on 3D Video Coding Extensions**  **of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11**  4th Meeting: Incheon, KR, 20–26 Apr. 2013 | Document: JCT3V-D0179 |

|  |  |  |  |
| --- | --- | --- | --- |
| *Title:* | **CE1.h related: Low-complexity BVSP for 3D-HEVC** | | |
| *Status:* | Input Document | | |
| *Purpose:* | Proposal | | |
| *Author(s) or Contact(s):* | Liu He Li Zhang Ying Chen  5775 Morehouse Drive San Diego, CA 92121 USA | Tel: Email: | 1-858-651-6660 [lizhang@qti.qualcomm.com](mailto:lizhang@qti.qualcomm.com)  1-858-845-6589 [cheny@qti.qualcomm.com](mailto:cheny@qti.qualcomm.com) |
| *Source:* | Qualcomm Incorporated | | |

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

# Abstract

In the current BVSP method of 3D-HEVC, motion compensation is performed always in a 4x4 block level regardless the size of prediction unit. To reduce the memory bandwidth and computational complexity, the size of motion compensation unit is set based on the size of the CU so that when the size of the CU becomes larger, less blocks in an inter-view reference picture need to be accessed. The proposed method can significantly reduce the memory bandwidth and computational complexity in the situation when CU is large. The contribution reports an average bitrate loss of 0.1% for video PSNR vs. video bitrate, and no loss for coded & synthesis PSNR vs. total bitrate.

# Introduction

In current 3D-HEVC, when BVSP is used, the motion compensation operation is performed for each 4x4 block in a prediction unit (PU). Before motion compensation of each 4x4 block, a maximum depth value is picked in every 4x4 block of the corresponding depth picture in the base view and converted to be the disparity motion vector of this 4x4 block [1].

# Problem

In other words, the size of the motion compensation unit is always 4x4 regardless the size of the current PU. However, when the coding PU with large size, e.g. 64x64, 256 (64x64/4x4) disparity motion vectors should be derived and 256 different blocks should be accessed. To reduce the memory bandwidth and computational complexity, adaptive size of motion compensation (MC) unit is proposed to reduce the complexity.

# Proposal

It is proposed that the motion compensation unit size is set to max(N/2, 4) x max(N/2, 4), wherein 2Nx2N is the CU size covering current PU.

For example, in Fig. 1, the size of current PU is 16x32, and the size of CU is 32x32. So the size of MC unit is set to be 8x8. In this case, the number of MC units will be reduced by 75% compared with current BVSP method.

This means significantly less access of different regions of the inter-view reference frame. If the size of current PU is 64x64, the number of motion compensation units will be reduced by 93.75% for the current PU.

32

16

8

32x32 CU contains two 16x32 PUs

8x8 MC unit

Figure 1. Example on larger MC unit size.

# Simulations results

The proposed method was implemented into HTM6.0. Simulation results of the proposal are shown in Table 1. Simulations were conducted following common test conditions [2].

As shown in Table 1, the proposed algorithm causes 0.1% coding loss for texture views and no loss if evaluated by synthesized views. The encoding and rendering measurement time may be not accurate because of hybrid CPUs used for the simulations. For the decoding time, it is measured with one single machine in windows operation.

Table 1: Proposed method vs HTM 6.0 (CTC)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | video 1 | video 2 | video PSNR / video bitrate | video PSNR / total bitrate | synth PSNR / total bitrate | enc time | dec time | ren time |
| Balloons | 0.0% | 0.0% | 0.0% | 0.0% | -0.1% | 88% | 93% | 87% |
| Kendo | 0.1% | 0.0% | 0.0% | 0.0% | 0.1% | 87% | 99% | 81% |
| Newspaper\_CC | -0.3% | 0.2% | 0.0% | 0.0% | -0.1% | 93% | 100% | 85% |
| GT\_Fly | 1.2% | 1.3% | 0.3% | 0.3% | 0.2% | 95% | 101% | 95% |
| Poznan\_Hall2 | -0.5% | 0.0% | -0.1% | -0.1% | 0.0% | 89% | 100% | 84% |
| Poznan\_Street | 0.8% | 0.3% | 0.2% | 0.1% | 0.1% | 88% | 100% | 87% |
| Undo\_Dancer | 2.1% | 2.1% | 0.5% | 0.5% | 0.0% | 86% | 100% | 79% |
| 1024x768 | -0.1% | 0.0% | 0.0% | 0.0% | 0.0% | 89% | 97% | 84% |
| 1920x1088 | 0.9% | 0.9% | 0.2% | 0.2% | 0.1% | 89% | 100% | 86% |
| average | 0.5% | 0.6% | 0.1% | 0.1% | 0.0% | 89% | 99% | 85% |

# References

1. D. Tian, F. Zou, A. Vetro (MERL), "CE1.h: Backward View Synthesis Prediction using Neighboring Blocks", JCT3V-C0152, Geneva, CN, Jan. 2013
2. D. Rusanovskyy, K. Mueller, A. Vetro, "Common Test Conditions of 3DV Core Experiments", JCT3V-B1100, Shanghai, CN, October 2012

# 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).**