|  |  |
| --- | --- |
| **Joint Collaborative Team on Video Coding (JCT-VC)**  **of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11**  6th Meeting: Torino, 14-22 July, 2011 | Document: JCTVC-F620 |

|  |  |  |  |
| --- | --- | --- | --- |
| *Title:* | Crosscheck of JCTVC-F319 Toshiba Adaptive Scaling with Offset RFC | | |
| *Status:* | Input Document to JCT-VC | | |
| *Purpose:* | Information | | |
| *Author(s) or Contact(s):* | Dzung Hoang | Tel: Email: | +1-408-532-5694 dzung.hoang@zenverge.com |
| *Source:* | Zenverge, Inc. | | |

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

# Abstract

This document reviews Toshiba’s JCTVC-F319 proposal for Adaptive Scaling with Offset (TASO) for compressing reference frame buffers. Several discrepancies between Toshiba’s software and the text document are identified. Toshiba’s BD-rate results for comparison between HM 3.0 and TASO are confirmed. However, differences are found with the comparison between HM 3.0 and HM 3.0 without IBDI. These differences are likely due to how HM is configured to operate without IBDI.

# Introduction

In JCTVC-F319 [1], Toshiba proposes a reference frame compression (RFC) scheme for N-to-8 compression. The proposed scheme is a combination of adaptive scaling proposed in [2] and adaptive offset proposed in [3].

# Analysis of Algorithm

We refer to Toshiba’s algorithm as Toshiba Adaptive Scaling with Offset (TASO). TASO operates upon 4×4 blocks of each YCrCb channel separately. Based upon the difference between the maximum pixel and minimum pixel values within a block, TASO determines a scaling factor S so that the residual after prediction by the minimum pixel can be represented in 7 bits in a process that we refer to as adaptive scaling. With adaptive scaling, the minimum pixel value is quantized by S and then used to predict each pixel in the block. The residual is then coded using 7 bits. If the scaling factor S is above a threshold, fixed rounding to 8 bits is used instead. When adaptive scaling is enabled, an offset value is computed from the sum of the quantization errors. This offset value is added to the reconstructed pixel after prediction and inverse quantization.

Toshiba advocates specifying the RFC algorithm from the point of view of an in-loop filter in that they specify only how each output block is computed based on the input block. There is no attempt to define a storage format to show that the compressed block can be stored in 128 bits. This is a serious weakness in their approach in that there is no proof of a minimum compression ratio. If TASO is to be generalized to N-to-8 compression, such a proof of minimum compression ratio would be highly desirable for all the values of N to be supported. Furthermore, Toshiba does not specify what values of internal bit depth (N) are supported.

In a past contribution [2], Toshiba did describe a storage format for the case of 10-to-8 compression. Therefore using [2] and [3] we can construct a storage format for 10-to-8 compression that fits in exactly 128 bits.

Listing : A possible storage format for TASO compression for 10-to-8 case

|  |
| --- |
| u(8) /\* P0\_flag \*/ |
| if (P0\_flag != 0) { |
| for (i=1; i<16; i++) u(8) /\* pixel value P[i] \*/ |
| } else { |
| u(1) /\* S: [0..1] \*/ |
| u(10-S) /\* M >> S \*/ |
| u(S) /\* offset \*/ |
| u(4) /\* M\_index \*/ |
| for (i=0; i<15; i++) u(7) /\* residual excluding M \*/ |
| } |

# Discrepancies

Comparing JCTVC-F319 and the software that Toshiba sent us, we found the following discrepancies.

1. In Section 3, Toshiba indicates that HM 2.0 plus the Memory AHG modifications was used. Clearly, they mean HM 3.0 plus the Memory AHG modifications.
2. In Annex A, the variable BitDepth is defined as the “input sample bit-depth.” We believe that the intention is for BitDepth to represent the actual sample bit-depth of the input video not including the internal bit depth increase. However, it was decided in a past meeting that the internal bit depth increase will not be signaled separately at the bitstream level and that the signaled bit depth would represent the video bit depth plus any internal increment.
3. Inspection of the software confirms usage of g\_uiBASE\_MAX, g\_uiIBDI\_MAX, and g\_uiBitIncrement. These variables depend upon knowledge of both the input bit depth and the internal bit depth. However, the Toshiba software has FULL\_NBIT set to 0 in TypeDef.h, which defaults to treating input video as 8-bit even for the 10-bit sequences. Therefore g\_uiBASE\_MAX is always set to 255. The variable MaxRange of Annex A is mirrored in the software as iMaxRange, and is always set to 128. Annex A says that MaxRange should be set to 64 when BitDepth >= 9. Therefore there is a mismatch between the software and Annex A.
4. The above mis-handling of 10-bit video Nebuta and SteamLocomotive shows up in Toshiba’s spreadsheet. For the HM 3.0 versus IBDI-off comparison, Nebuta incurred a 1.1% coding loss and SteamLocomotive incurred 9.9% coding loss. Given that these are 10-bit sequences, there should be no loss when turning IBDI off.
5. It was initially suspected that items 3 and 4 are problems with the HM 3.0 default of setting FULL\_NBIT to 0. However, we investigated setting FULL\_NBIT to 1 and found that this breaks HM 3.0.

# Simulation Results

We ran the Toshiba software using RA-HE and LB-HE configurations. BD-rate results match for the comparison between HM 3.0 and TASO. Our decoder run times are higher for TASO, but our run time results are not reliable. We have rerun the decoder several times for HM 3.0 and get wildly varying run times. The results are shown in Table 1 below.

Table : BD-Rate results comparing HM 3.0 against TASO

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | |  | Low delay B HE | | | Random Access HE | | | | Y | U | V | Y | U | V | | Class A |  |  |  | 0.07 | -0.07 | 0.20 | | Class B | 0.38 | 0.11 | 0.23 | 0.11 | -0.03 | 0.01 | | Class C | 0.22 | 0.21 | 0.14 | 0.10 | 0.15 | 0.11 | | Class D | 0.09 | 0.19 | 0.52 | 0.08 | -0.02 | 0.03 | | Class E | 3.00 | 0.58 | 0.76 |  |  |  | | **Overall** | **0.76** | **0.24** | **0.38** | **0.09** | **0.01** | **0.09** | | Enc Time[%] | 102% | | | 100% | | | | Dec Time[%] | 115% | | | 114% | | | |

For the comparison between HM 3.0 with and without IBDI, we get differences as shown in Table 2 and Table 3. The two results differ mostly with the Class A sequences. The differences are likely to be due to how HM 3.0 is configured to run without IBDI. We disabled IBDI by setting the “--InternalBitDepth” command-line switch to reflect the input bit depth of each video sequence. For the 10-bit sequences Nebuta and SteamLocomotive, we used “--InternalBitDepth=10” and for the other 8-bit sequences we used “--InternalBitDepth=8.” We suspect that Toshiba is using “--InternalBitDepth=8” even for the 10-bit sequences. The other minor differences can be attributed to the use of different revisions of HM 3.0.

Table : Toshiba BD-Rate results comparing HM 3.0 HM 3.0 w/o IBDI

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | |  | Low delay B HE | | | Random Access HE | | | | Y | U | V | Y | U | V | | Class A |  |  |  | 3.52 | 19.76 | 21.80 | | Class B | 2.95 | 11.45 | 12.63 | 2.63 | 8.18 | 7.47 | | Class C | 1.52 | 5.25 | 6.12 | 1.36 | 3.95 | 4.05 | | Class D | 0.87 | 8.56 | 9.42 | 0.92 | 3.14 | 2.98 | | Class E | 6.90 | 17.45 | 13.21 |  |  |  | | **Overall** | **2.81** | **10.30** | **10.31** | **2.14** | **8.72** | **8.98** | | Enc Time[%] | 100% | | | 99% | | | | Dec Time[%] | 95% | | | 97% | | | |

Table : Zenverge BD-Rate results comparing HM 3.0 HM 3.0 w/o IBDI

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | |  | Low delay B HE | | | Random Access HE | | | | Y | U | V | Y | U | V | | Class A |  |  |  | 0.75 | 2.41 | 2.66 | | Class B | 3.00 | 11.29 | 12.39 | 2.61 | 8.16 | 7.48 | | Class C | 1.53 | 5.32 | 6.07 | 1.38 | 3.97 | 4.07 | | Class D | 0.88 | 8.92 | 9.13 | 0.93 | 3.07 | 3.05 | | Class E | 6.76 | 17.79 | 13.20 |  |  |  | | **Overall** | **2.81** | **10.42** | **10.15** | **1.49** | **4.62** | **4.50** | | Enc Time[%] | 100% | | | 99% | | | | Dec Time[%] | 97% | | | 97% | | | |

# References

1. T. Chujoh and T. Yamakage, “Adaptive scaling with offset for reference pictures memory compression,” JCT-VC Document JCTVC-F319, Torino, July 2011.
2. T. Chujoh and T. Yamakage, “Adaptive scaling for reference pictures memory compression,” JCT-VC Document JCTVC-E133, Geneva, March 2011.
3. D. Hoang, “Unified scaling with adaptive offset for reference frame compression with IBDI,” JCT-VC Document JCTVC-D035, Daegu, January 2011.

# Patent rights declaration(s)

**Zenverge, Inc. 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).**