

**ISO/IEC JTC 1/SC 29/WG 11**

**Coding of moving pictures and audio**

**Convenorship: UNI (Italy)**

**ISO/IEC JTC 1/SC 29/WG 11 N19535**

**Document type: Approved WG 11 document**

**Title: V-PCC EE4FE 2.7 on multiple video codec integration in V-PCC software**

**Status: Approved**

**Date of document: 2020-07-17**

**Source: 3DG**

**Expected action:**

**No. of pages: 12**

**Email of convenor: leonardo@chiariglione.org**

**Committee URL: mpeg.chiariglione.org**

**INTERNATIONAL ORGANISATION FOR STANDARDISATION**

**ORGANISATION INTERNATIONALE DE NORMALISATION**

**ISO/IEC JTC 1/SC 29/WG 11**

**CODING OF MOVING PICTURES AND AUDIO**

**ISO/IEC JTC 1/SC 29/WG 11 N19535**

**June 2020, teleconference**

|  |  |
| --- | --- |
| **Source:** | **3DG** |
| **Status** | **Draft** |
| **Title** | **V-PCC EE4FE 2.7 on multiple video codec integration in V-PCC software** |
| **Author** | **Yiting Tsai (ITRI.), Alexandros Tourapis (Apple)** |

# Abstract

This document provides a description of the V-PCC Exploration Experiment 2.7 on the integration of multiple video coding specifications and their support in the V-PCC software.

# Introduction

The goal of the V-PCC Exploration Experiment 2.7 is the integration of multiple different video coding specifications in the V-PCC software. This will enable the 3DG group to validate the agnostic video coding nature and characteristics of the V-PCC specification and evaluate the behavior of these specifications for coding the different data components, i.e. occupancy, geometry, and attribute data, of V-PCC. Such could provide us with considerable insights in terms of the characteristics of such data as well as on the relationships of such data and the coding tools supported by such video specifications.

For this work we considered several implementations of different video coding standards the HEVC and AVC implementations supported in the ffmpeg distribution package [4], the VTM reference software of the Versatile Video Coding (VVC) specification, and the JM reference software for the Advance Video Coding (AVC/H.264) specification [5]. More details on such support are included in m52889 [3] , m53410 [6], m53510[7], m54574[8], m54575[9] and m54665[10]

These different coding specifications are compared versus the coding performance and coding speed of the HM reference software of HEVC. Our evaluations used the V-PCC TMC2 release-v9.0 [1] and all experiments were conducted using the official V-PCC CTC conditions [2].

# Mandates

The mandates for this EE are as follows:

1. To study the coding performance of separate components and their combinations with the reference HM video coding implementation and the video coding specifications available in the ffmpeg distribution package.
2. To provide recommendations on best practices for the coding of the various v-pcc video components, including geometry, attribute, and occupancy map data when using a combination of different video coding specifications and SW packages.
3. To verify the codec-agnostic characteristics of the V-PCC, and consequently V3C, architecture
4. To implement carriage of the video components in a sample stream format
5. To propose appropriate configuration files and evaluate the coding performance for the various video sub-components of V-PCC when using different coding specifications and their corresponding SW encoders
6. To integrate support for the different coding specifications within TMC2 using a library form instead of using system commands and external SW calls.

# Participants

|  |  |  |  |
| --- | --- | --- | --- |
| ***Participant*** | ***Contact*** | ***Email*** | ***Type*** |
| Futurewei technologies | Vlad Zakharchenko | Vladyslav.zakharchenko@futurewei.com | P |
| ITRI | YiTing Tsai | yitingtsai@itri.com | P |
| Apple | Jungsun Kim  Alexandors Tourapis | jungsun\_kim@apple.com  alexismt@apple.com | P |
| RWTH | Dominik Mehlem | mehlem@ient.rwth-aachen.de | P |

(P=proponent, C=crosss checker)

# Methods to be evaluated

## M52889 [V-PCC][SW] on ffmpeg integration in TMC2

The ffmpeg open source project [4] consists of a vast software suite of programs and libraries for handling multimedia data, including support of several video and image compression specifications, such as AVC, HEVC, JPEG etc. Although such support may be limited in terms of certain profiles, the implementations of common profiles, such as HEVC Main 10, have been heavily optimized in terms of speed, especially compared to equivalent reference SW implementations developed by MPEG and the ITU-T. In particular, integrating the ffmpeg library for HEVC decoding[11] resulted in a significant speedup of the V-PCC decoding process compared to the use of the HM decoder. It has been suggested to conduct an exploration experiment that would confirm the possibility of re-using existing video coding specification capabilities and come up with the best practices for dynamic point cloud compression using the V-PCC specification.

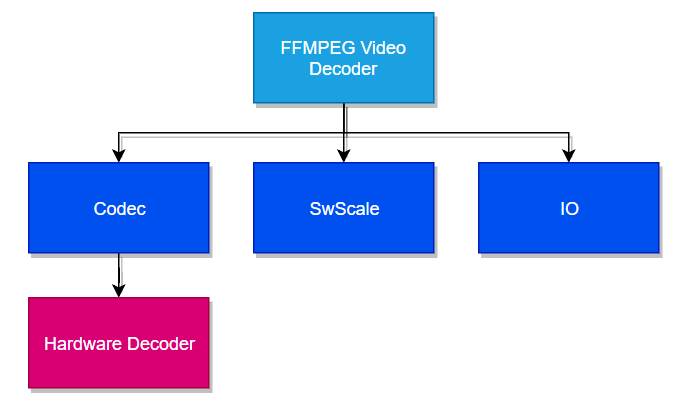


figure. 1. ffmpeg decoder architecture

## [VPCC] [EE4FE 2.7 related] AVC codec integration in V-PCC test model

Today, AVC is the dominant video coding technology used world-wide. A lot of video services use AVC, such as terrestrial broadcasting, direct broadcast satellite TV services, IPTV services…etc. In addition, AVC can provide for a more hardware friendly implementation compared to HEVC.

In this contribution we propose to also include support for the MPEG-4 AVC/H.264 specification into the V3C/V-PCC test model. For this purpose, we integrated the AVC reference software, JM, version 19.0 into the V-PCC TMC2 V9 model. One limitation of this SW, however, is that it only supports the byte stream file format of AVC. Unfortunately, it is currently required that AVC, as well as HEVC, video sub-bitstreams in V3C/V-PCC are in the sample stream format. A modification of this SW to support the sample stream format is therefore needed. Furthermore, because of the nature of the V-PCC video data, such as the presence of maps, additional changes were necessary in the default configuration files provided with this reference SW. These modifications are shown in detail in tables 1 through 3. It should be highlighted that these modifications are still suboptimal and further investigation on how to best configure the JM software, or even SW modifications so as to allow us to match the current HM reference configurations, are needed. For example, we have used a conservating intra biased coding structure, without making use of any Hierarchical coding or B slices.

*Table 1.The modifications of occupancy map configuration*

|  |  |
| --- | --- |
| Before (encoder.cfg in JM19.0) | After(modifications for PCC) |
| ProfileIDC = 100 | ProfileIDC = 244(High 4:4:4) |
| LevelIDC = 40 | LevelIDC = 51 |
| IntraPeriod = 0 | IntraPeriod = 1 |
| QPISlice = 28 | QPISlice = 0 |
| QPPSlice = 28 | QPPSlice = 0 |
| LosslessCoding = 0 | LosslessCoding = 1 |

*Table 2.The modifications of geometry map configuration*

|  |  |
| --- | --- |
| Before (encoder.cfg in JM19.0) | After(modifications for PCC) |
| LevelIDC = 40 | LevelIDC = 51 |
| IDRPeriod = 0 | IDRPeriod = 2 |
| EnableIDRGOP = 0 | EnableIDRGOP = 1 |
| NumberBFrames = 7 | NumberBFrames = 0 |

*Table 3.The modifications of attribute map configuration*

|  |  |
| --- | --- |
| Before (encoder.cfg in JM19.0) | After(modifications for PCC) |
| LevelIDC = 40 | LevelIDC = 51 |
| IDRPeriod = 0 | IDRPeriod = 2 |
| EnableIDRGOP = 0 | EnableIDRGOP = 1 |
| NumberBFrames = 7 | NumberBFrames = 0 |

## Versatile Video Coding for V-PCC

The current V-PCC design is video codec agnostic. However, for the current testing procedure in the TMC2 V‑PCC test model, the development effort of the V-PCC specifications was almost on its entirety was done using the HM reference software of HEVC. This was done for a variety of reasons, including the fact that at the point of starting this work the HEVC was considered as the current state in the art in terms of video coding and since it was well acknowledged that other video specifications could be used within the same. This also helped considerably the development of the V-PCC specification since it allowed researchers to focus on the High Level Syntax aspects and the pre and post-processing stages needed for the conversion of the point cloud data to projected video images and vice versa.

However, since different video coding specifications may include different coding tools and can have very different coding characteristics, it can be quite interesting and useful for future development to evaluate their behavior and coding performance characteristics in the context of V-PCC. This is especially true for VVC, which is the latest video coding specification developed jointly by MPEG and the ITU-T. It would be quite interesting to see how the various new coding tools introduced in this specification behave in the context of V-PCC.

Tables 4 through 7 provide a summary of the coding performance impact of the VTM reference software of VVC compared to that of the HM of HEVC. It can be seen that considerable coding gain can be achieved, which seems comparable to the coding gains also reported for other 2D video applications targetted by the VVC specification.

*Table 4 - Performance impact of using the VTM (VVC) instead of the HM (HEVC) (AI, 32 frames)*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Class | D1 | D2 | Luma | Cb | Cr |
| loot | -17.1% | -14.9% | -17.8% | -24.0% | -17.3% |
| redandblack | -23.4% | -23.2% | -23.1% | -20.8% | -6.2% |
| soldier | -12.6% | -12.3% | -16.5% | -28.4% | -28.6% |
| queen | -27.9% | -27.8% | -19.3% | -24.7% | -18.4% |
| longdress | -20.1% | -21.1% | -21.3% | -19.4% | -11.8% |
| basketball | -30.6% | -28.5% | -21.1% | -20.1% | -28.5% |
| dancer | -28.4% | -26.4% | -21.2% | -21.0% | -22.8% |
| AVG | -22.9% | -22.0% | -20.0% | -22.6% | -19.1% |

*Table 5 - Performance impact of using VTM (VVC) instead of the HM (HEVC) (AI, full)*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Class | D1 | D2 | Luma | Cb | Cr |
| loot | -16.5% | -14.5% | -17.8% | -24.4% | -18.0% |
| redandblack | -22.8% | -22.5% | -22.9% | -20.5% | -5.9% |
| soldier | -12.5% | -12.5% | -16.6% | -28.3% | -29.8% |
| queen | -26.8% | -26.9% | -19.6% | -27.8% | -22.4% |
| longdress | -21.1% | -21.1% | -21.5% | -21.6% | -12.8% |
| basketball | -30.2% | -28.0% | -21.1% | -19.7% | -27.5% |
| dancer | -28.5% | -26.3% | -20.9% | -20.0% | -22.9% |
| AVG | -22.6 | -21.7% | -20.1% | -23.2% | -19.9% |

*Table 6 - Performance impact of using VTM (VVC) instead of the HM (HEVC) (RA, 32 frames)*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Class | D1 | D2 | Luma | Cb | Cr |
| loot | -25.0% | -24.7% | -24.1% | -37.6% | -29.3% |
| redandblack | -26.7% | -27.1% | -24.5% | -29.6% | -19.5% |
| soldier | -25.5% | -25.7% | -26.5% | -40.6% | -41.9% |
| queen | -33.1% | -33.4% | -23.7% | -35.4% | -31.2% |
| longdress | -25.8% | -26.6% | -25.7% | -28.9% | -23.4% |
| basketball | -33.4% | -32.5% | -25.1% | -29.5% | -31.8% |
| dancer | -30.9% | -29.7% | -24.1% | -21.9% | -20.9% |
| AVG | -28.6% | -28.5% | -24.8% | -31.9% | -28.3% |

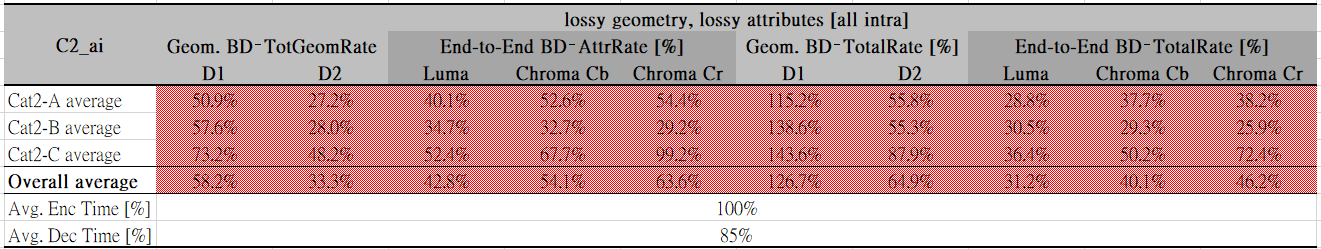
*Table 7 - Performance impact of using VVC instead of HEVC (RA, full)*

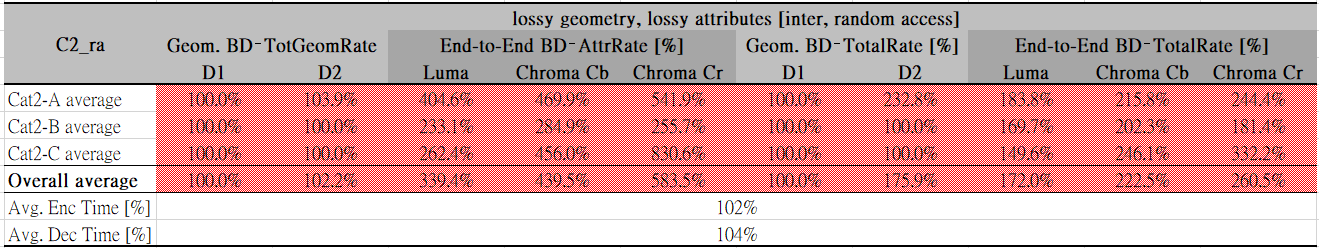
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Class | D1 | D2 | Luma | Cb | Cr |
| loot | -24.4% | -23.8% | -33.4% | -33.4% | -27.5% |
| redandblack | -26.0% | -26.3% | -24.2% | -27.1% | -18.7% |
| soldier | -23.6% | -23.9% | -25.2% | -37.3% | -36.8% |
| queen | -32.1% | -32.3% | -23.8% | -37.6% | -32.9% |
| longdress | -27.4% | -27.6% | -26.7% | -31.9% | -25.6% |
| basketball | -32.9% | -31.7% | -24.9% | -26.4% | -29.0% |
| dancer | -30.8% | -29.6% | -24.1% | -21.9% | -20.3% |
| AVG | -28.2% | -27.9% | -24.7% | -30.8% | -27.3% |

## [VPCC] [EE4FE 2.7 Report] Evaluation of the results of AVC integration in V-PCC software

This contribution provided information on several experiments conducted in the context of EE4FE 2.7. In particular, it results of two different variants for the integration of the AVC specification in the PCC software. In the first, AVC coding was used for all video components of V-PCC including the occupancy map, and the geometry and attribute data. In the second experiment, AVC was used only for coding the occupancy map data, while all other video components were coded using the HEVC specification.

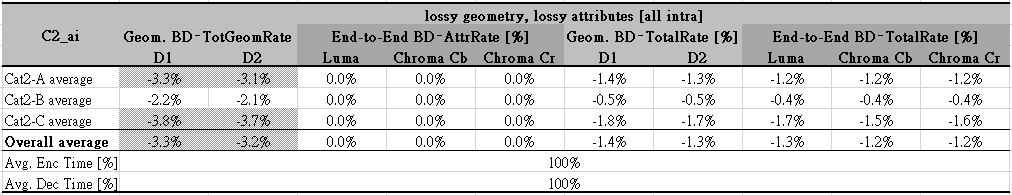
In the first experiment, and as expected, a considerable coding loss was shown compared to the use of the HEVC specification. It should be highlighted that a lot of the loss was due to the use of a very different coding configuration, including coding structures, than those used for the HEVC encodings. Unfortunately, because of limitations in the JM configuration but also other reasons, we did not fully attempt to match the coding configurations between the two SW. Test1: AVC (Occupancy map + Geometry map + Attribute map) v.s. TMC2 V9

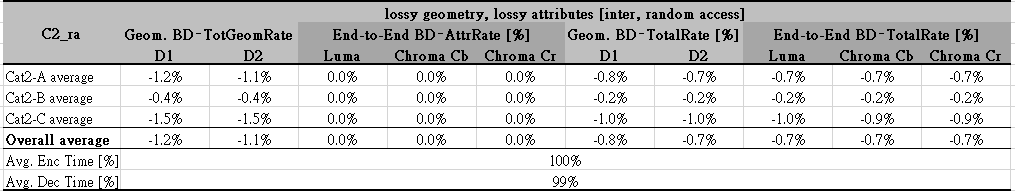




In the second case, we observed that using the JM reference SW for the encoding of the occupancy, some gains could be achieved compared to the exclusive use of the HM reference SW of HEVC. We have not yet explored on how much of this gain stems from the differences in the coding configuration structure of the JM vs that of the HM. The results show on average about 1.3% and 1.2% in coding gain in luma and in chroma, respectively, for the all intra condition. Furthermore, coding gains of 0.8% and 0.7% in terms of BD-rate bitrate savings for luma and chroma in the random access case is also shown.

Test2: AVC (Occupancy map) + HEVC(Geometry map + Attribute map) v.s. TMC2 V9

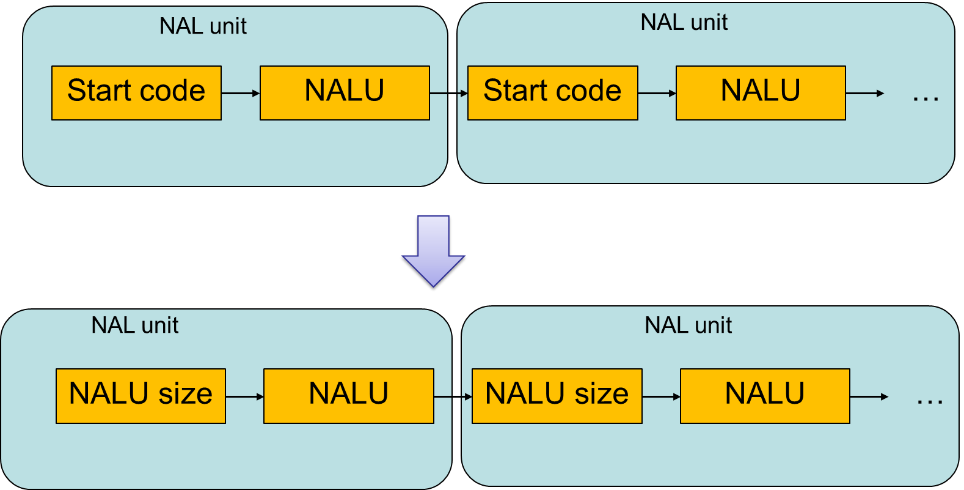


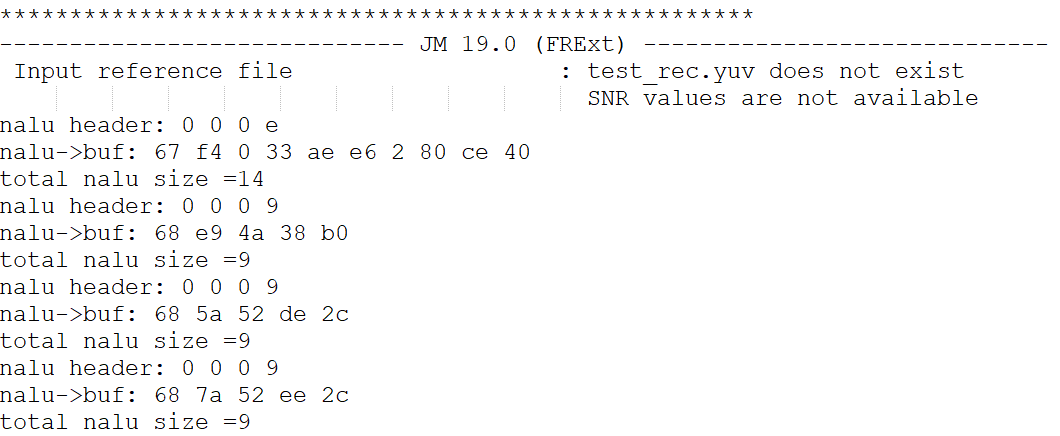


## [VPCC] [EE4FE 2.7 Report] Evaluation of the sample stream format for AVC

It has been recommended that the encapsulation of video data in the context of V-PCC is only done using the sample stream format instead of the byte stream format. However, for a variety of reasons, the AVC and HEVC reference SW only support the byte stream format, and the development of V-PCC in its entirety utilized this format. It was highly recommended that sample stream format support was introduced into both the AVC and HEVC (as well as the VVC VTM reference SW) and it was suggested that this EE also studies such implementations. This document provides some preliminary results on the support of the sample stream format using the JM reference SW of the Advanced Video Coding (AVC) specification and its consideration in the context of V-PCC.

* AVC with sample stream header

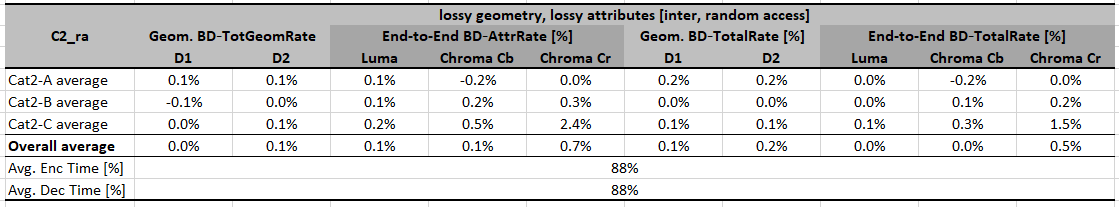




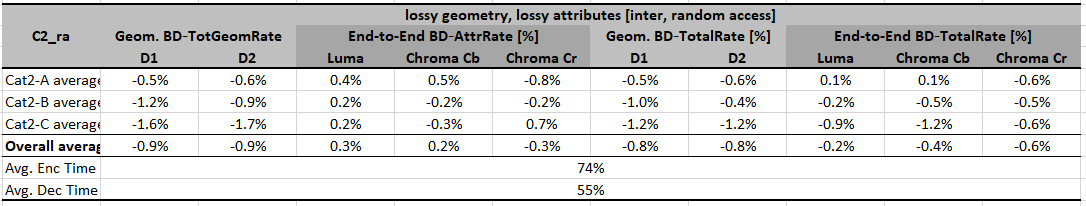
## [VPCC] [EE4FE 2.7 Report] VVC tool analysis for V-PCC content

RWTH provided several results pertaining to the use of the VVC specification and it's VTM reference SW in the context of V-PCC. The tests included the following configuration settings: GEO off, IBC on, ISP off, MIP off, MRL off, SAO off, SBT off, Deblocking filter off. The tests were conducted using VTM version 8.2 and were compared to the VTM CTC for the random access case. The VPCC software is in random access configuration in compliance with the CTC, but with 3D motion compensation disabled.

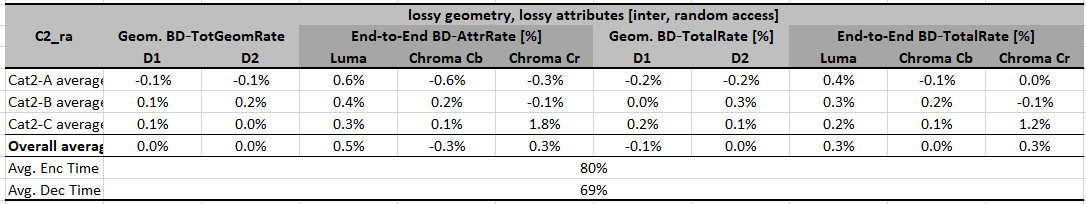
* Test 1: VVC(GEO off) v.s. VVC CTC (ra, 32 frames)



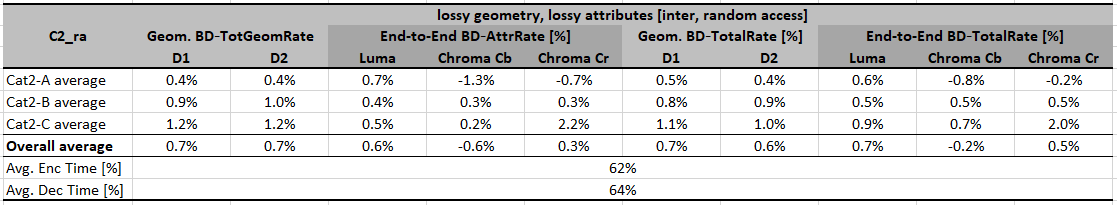
* Test 2: VVC(IBC on) v.s. VVC CTC (ra, 32 frames)



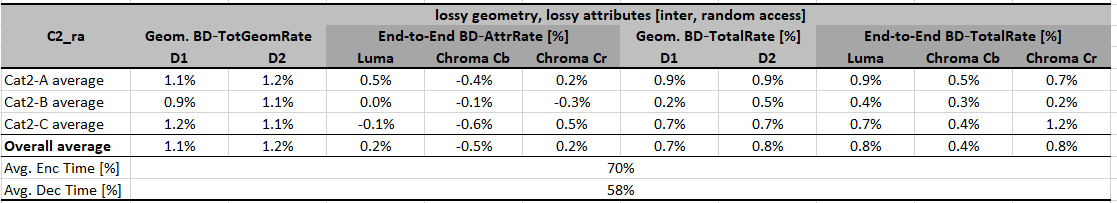
* Test 3: VVC(ISP off) v.s. VVC CTC (ra, 32 frames)



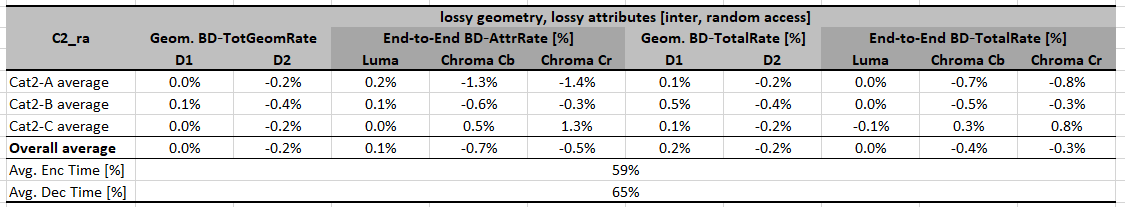
* Test 4: VVC(MIP off) v.s. VVC CTC (ra, 32 frames)



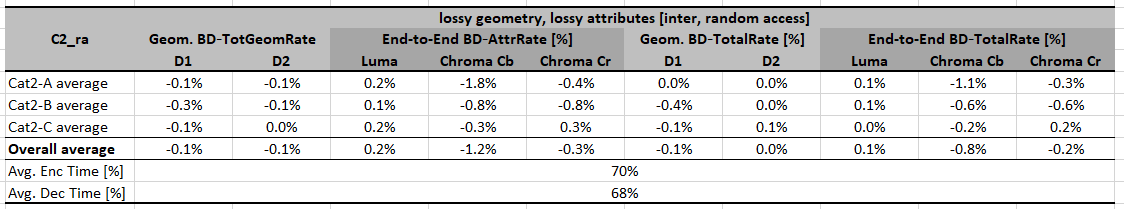
* Test 5: VVC(MRL off) v.s. VVC CTC (ra, 32 frames)



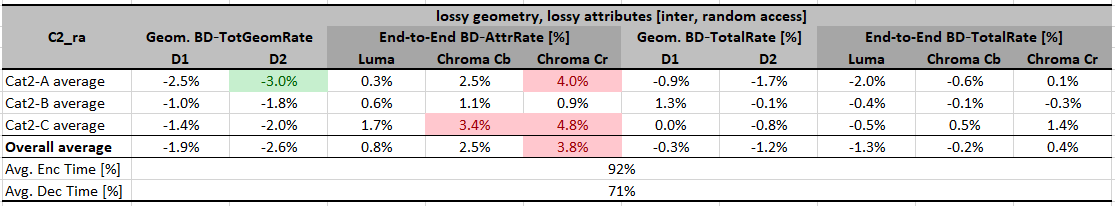
* Test 6: VVC(SAO off) v.s. VVC CTC (ra, 32 frames)



* Test 7: VVC(SBT off) v.s. VVC CTC (ra, 32 frames)



* Test 8: VVC(Deblock off) v.s. VVC CTC (ra, 32 frames)



# Evaluation

## Item

1. CTC anchor .
2. CTC anchor with 3d motion compensation disabled.
3. CTC anchor with vvc encoder for occupancy, the geometry and attribute coding in sample stream format.
4. CTC anchor with avc encoder for occupancy, the geometry and attribute coding in sample stream format.
5. CTC anchor with avc encoder for occupancy map and hevc encoder for the geometry and attribute coding.
6. CTC anchor with hevc sample stream format NALU.
7. Additional combinations suggested at the evaluation process

## Test

A mobile device or a head-mounted display may have multiple decoders integrated within it and it may be more efficient to use those independent decoders instead of using a single decoder of one specification. Furthermore, even though HEVC is commonly more efficient than AVC, it is however much more complicated. This is also true for VVC. Using a lower complexity codec specification for certain components where maybe the more complex specification give fewer benefits may be advisable.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test no | Component | Occupancy map component | Geometry component | Attribute component | Condition |
| 1 | Reference, CTC | HM16.20 | HM16.20+ 3DME | HM16.20 + 3DME | CTC |
| 2 | Reference, constrained | HM16.20 | HM16.20 | HM16.20 | CTC |
| 3 | VVC | VTM | VTM | VTM | Lossy |
| 4 | AVC | JM | JM | JM | Lossy |
| 5 | OM-avc | JM | HM16.20 + 3DME | HM16.20 + 3DME | Lossy/lossless |
| 6 | Additional combinations | TBD | TBD | TBD | TBD |

## BDBR performance and Time complecity

The coding efficiency and run-time of the various conducted tests were evaluated between 2 items. For the full decoded point cloud sequence, the BDBR in the CTC spreadsheet is used for evaluation. It was encouraged to isolate these results. For example, one could isolate and report the performance of lossless coding of the occupancy, as well as the coding performance of lossy coding of the geometry for all three coding specifications. For attribute coding, one could settle on one particular coding method for the geometry signal (e.g. use of the HM software of HEVC), so that the attribute signals would be identical and could be used as inputs for all cases. These could then be coded using all other specifications of interest and these coding results could be more fairly evaluated.

## Additional works

In the V-PCC specification, determining the video coding specifications that are used for the different video coding components of V-PCC can be done through either looking at the codec profile component of V-PCC or by looking at the information in the codec component SEI message that was introduced in this specification. It was highly encouraged to integrate this SEI into the TMC2 reference SW since that could enable seamless detection of the video coding specifications used by each bitstream and allow for an automated process of decoding any V-PCC bitstream without the need for user interaction. That is, the V-PCC decoder could on it's own determination and automatically call the appropriate video decoders that should be used to decode heterogeneously coded video sub-components of V-PCC.

# Timeline

* 2020-07-17: V-PCC Expected date for release of finalized CE description CTC
* 2020-07-31: V-PCCv10 release software
* 2020-08-07: Expected date for the release of cross-verified TMC2v10 software and anchors
* 2020-10-07: MPEG document upload deadline
* 2020-10-12: MPEG #132(Rennes) meeting start

# References

1. “V-PCC Test Model v9”, ISO/IEC JTC1/SC29/WG11 MPEG2020 Doc. w19085, Brussels, BE, January 2020
2. “Common Test Conditions for PCC” ISO/IEC JTC1/SC29 WG11 MPEG2020 Doc, w19083, Brussels, BE, January 2020
3. “[V-PCC][SW] on ffmpeg integration in TMC2” , ISO/IEC JTC1/SC29 WG11 (MPEG) input document m52889, Brussels, BE, January 2020
4. ffmpeg software implementation - <https://www.ffmpeg.org/download.html#get-sources>
5. Advanced Video Coding- <https://en.wikipedia.org/wiki/Advanced_Video_Coding>
6. “[VPCC] [EE4FE 2.7 related] AVC codec integration in V-PCC test model” , ISO/IEC JTC1/SC29 WG11 (MPEG) input document m53410, Alpbach, AT, April 2020
7. “[VPCC] Versatile Video Coding for VPCC” , ISO/IEC JTC1/SC29 WG11 (MPEG) input document m53510, Brussels, BE, January 2020
8. “[VPCC] [EE4FE 2.7 Report] Evaluation of the results of AVC integration in V-PCC software” , ISO/IEC JTC1/SC29 WG11 (MPEG) input document m54574, teleconference, June 2020
9. “[VPCC] [EE4FE 2.7 Report] Evaluation of the sample stream format for AVC” , ISO/IEC JTC1/SC29 WG11 (MPEG) input document m54575, teleconference, June 2020
10. “[VPCC] [EE4FE 2.7 Report] VVC tool analysis for VPCC content” , ISO/IEC JTC1/SC29 WG11 (MPEG) input document m54665, teleconference, June 2020
11. OpenHEVC-<https://github.com/OpenHEVC/openHEVC>