
ISO/IEC JTC 1/SC 29/WG 11**Coding of moving pictures and audio****Convenorship: UNI (Italy)**

Document type: Approved WG 11 document

Title: G-PCC TMC13v8 performance evaluation and anchor results

Status: Final

Date of document: 2019-12-06

Source: 3DG

Expected action: None

No. of pages: 11

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 N18885
Geneva, CH – October 2019

Source: 3DG

Title: G-PCC TMC13v8 performance evaluation and anchor results

Abstract

This document provides the reference anchor results for experiments on point cloud compression for dynamically acquired content (category three) and high density content (category one) using the N18883 common test conditions [1].

Summary

This report contains the following:

report_*.txt	verification report of all data points
pcc-\$B__vs__\$A.xlsm	results reporting \$B against \$A

Bitstreams and results were generated on a heterogeneous 64bit linux cluster using revision release-v8.0-rc1 of TMC13 built with gcc-5.3.1:

```
CMAKE_BUILD_TYPE:STRING=Release
CMAKE_CXX_FLAGS:STRING=-g -O3
CMAKE_CXX_FLAGS_RELEASE:STRING=-O3 -DNDEBUG
```

All distortion are measured using pc_error version release-0.13.4. Due to the nature of the cluster environment, reported run time changes are approximate only.

Subsequent to verification, the tag “release-v8.0” is available from <http://mpegx.int-evry.fr/software/MPEG/PCC/TM/mpeg-pcc-tmc13>. Further software documentation and usage description is available [2, 3].

Anchor results according to common test conditions

Anchor results using the following common test conditions of N18883 are reported in the enclosed reporting sheets^{1,2}:

- C1: (near) lossless geometry, lossy attributes [all intra],
- C2: lossy geometry, lossy attributes [all intra],
- CW: (near) lossless geometry, lossless attributes [all intra],
- CX: (near) lossless geometry, near lossless attributes [all intra],

NOTE — TMC13 is currently an intra only codec supporting random access.

¹[pcc-tmc13-tmc13v8.0_octree_raht_vs__tmc13v8.0_octree_predlift.xlsm](#)

²[pcc-tmc13-tmc13v8.0_trisoup_raht_vs__tmc13v8.0_trisoup_predlift.xlsm](#)

Summary analysis of v8.0-rc1 against v7.0 results

Compression results comparing v8.0-rc1 against v7.0 on test sequences from categories one and three using both the lod-based lifting/predicting transforms and RAHT are provided with this report³⁴⁵⁶ and summarised in tables 1 to 4.

Table 1 – Summary performance of octree geometry and predlift attribute coding using release v8.0-rc1 relative to v7.0 results

Condition	Class	BPP Ratio [%]			Refl	BD-Rate [Δ%]						Avg. of ratio maxrssk [%]		Ratio of avg. runtime [%]	
		Geometry	Colour			D1	D2	Y	Cb	Cr	R	Encoder	Decoder	Encoder	Decoder
C1_ai	cat1-A							-3.1	-1.8	-2.2		81	93	110	115
C1_ai	cat3-fused							-0.9	-1.4	-1.2	-1.8	95	95	184	189
C1_ai	cat3-frame										-10.2	137	156	121	126
C1_ai	overall							-2.9	-1.7	-2.1	-7.7	95	107	118	123
C2_ai	cat1-A					-0.2	-0.2	-0.2	0.1	2.0		91	137	14	116
C2_ai	cat1-B					-2.5	-2.5					91	106	86	
C2_ai	cat3-fused					-2.0	-2.0	-0.4	-1.9	-2.1	-1.2	86	86	26	96
C2_ai	cat3-frame					-8.5	-8.5				-8.2	159	221	102	132
C2_ai	overall					-2.3	-2.3	-0.2!	-0.1!	1.5!	-6.1	99	131	40	
CW_ai	cat1-A	98.1	81.7									80	96	113	124
CW_ai	cat1-B	91.2										130	179	285	364
CW_ai	cat3-fused	98.0	92.4	99.6								95	95	172	187
CW_ai	cat3-frame	94.2		97.5								137	156	107	112
CW_ai	overall	93.0	83.4!	98.3								109	139	171	198
CY_ai	cat1-A							-10.5	-10.5	-10.5		80	96	112	118
CY_ai	cat3-fused							-9.0	-9.0	-9.0	-0.8	95	95	197	199
CY_ai	cat3-frame										-3.4	137	156	121	126
CY_ai	overall							-10.3	-10.3	-10.3	-2.6	94	109	120	126

NOTE — Condition CY metrics reported using Hausdorff PSNR.

Table 2 – Summary performance of octree geometry and RAHT attribute coding using release v8.0-rc1 relative to v7.0 results

Condition	Class	BPP Ratio [%]			Refl	BD-Rate [Δ %]					Avg. of ratio maxrssk [%]		Ratio of avg. runtime [%]	
		Geometry	Colour	D1		D2	Y	Cb	Cr	R	Encoder	Decoder	Encoder	Decoder
C1_ai	cat1-A						-0.1	-0.1	-0.1		81	87	106	109
C1_ai	cat1-B						-0.1	-0.1	-0.1		118	142	148	151
C1_ai	cat3-fused						-0.1	-0.1	-0.1	0.0	94	93	142	137
C1_ai	cat3-frame									0.0	136	156	109	110
C1_ai	overall						-0.1	-0.1	-0.1	0.0	105	119	125	127
C2_ai	cat1-A				-0.2	-0.2	-0.0	2.7	2.4		90	132	13	104
C2_ai	cat1-B				-2.5	-2.5	0.2	1.1	1.2		92	101	25	117
C2_ai	cat3-fused				-2.0	-2.0	1.5	0.5	0.4	1.5	85	79	32	110
C2_ai	cat3-frame				-8.5	-8.5				0.4	159	221	100	112
C2_ai	overall				-2.3	-2.3	0.2	1.8	1.7	0.8	99	127	24	111

Table 3 – Summary performance of trisoup geometry and lifting based attribute coding using release v8.0-rc1 relative to v7.0 results

Condition	Class	BPP Ratio [%]			Refl	BD-Rate [Δ%]					Avg. of ratio maxrssk [%]		Ratio of avg. runtime [%]		
		Geometry	Colour			D1	D2	Y	Cb	Cr	R	Encoder	Decoder	Encoder	Decoder
C2_ai	cat1-A					−0.1	−0.0	0.5	0.9	0.7		99	93	24	111

Table 4 – Summary performance of trisoup geometry and RAHT attribute coding using release v8.0-rc1 relative to v7.0 results

Condition	Class	BPP Ratio [%]			Refl	BD-Rate [Δ%]					Avg. of ratio maxrssk [%]		Ratio of avg. runtime [%]		
		Geometry	Colour			D1	D2	Y	Cb	Cr	R	Encoder	Decoder	Encoder	Decoder
C2_ai	cat1-A					−0.1	−0.0	1.0	2.1	2.2		99	90	24	105
C2_ai	cat1-B					0.1	0.1	2.9	3.3	3.7		82	82	23	99
C2_ai	overall					0.0	0.0	2.0	2.8	2.9		90	86	23	102

³[pcc-tmc13-tmc13v8.0-rc1_octree_predlift_vs__tmc13v7.0_octree_predlift.xlsm](#)

⁴[pcc-tmc13-tmc13v8.0-rc1_octree_raht_vs__tmc13v7.0_octree_raht.xlsm](#)

⁵[pcc-tmc13-tmc13v8.0-rc1_trisoup_lift_vs__tmc13v7.0_trisoup_lift.xlsm](#)

⁶[pcc-tmc13-tmc13v8.0-rc1_trisoup_raht_vs__tmc13v7.0_trisoup_raht.xlsm](#)

Cross checking

A cross-check of release-v8.0-rc1 was kindly performed by BlackBerry over all CTC configurations (octree, trisoup, RAHT, predlift) and conditions (C1, C2, CW, CX). All cross-checks⁷⁸⁹¹⁰ completed successfully and any deviation in exact reported results due to average calculation methods is negligible.

Tool verification

Following the integration of each tool, tests are made to verify the integration with differential results provided with the report. Only a single frame of the multi-frame sequences are used to reduce simulation time.

The general progression of coding performance with successive integrations is shown in tables 5 to 8.

Table 5 – Octree & lifting transform progression – C1_ai,overall

Condition	Class	BPP Ratio [%]			Ref	BD-Rate [Δ %]					R	Avg. of ratio maxrssk [%]		Ratio of avg. runtime [%]	
		Geometry	Colour			D1	D2	Y	Cb	Cr		Encoder	Decoder	Encoder	Decoder
C1_ai	00=7dot1 fixes						0.0	-0.0	-0.0	0.0	100	100	112	113	
C1_ai	01=hls+refactors						0.0	-0.0	-0.0	0.0	100	100	109	114	
C1_ai	02=attrectx						-0.1	-0.1	-0.2	0.0	100	100	111	115	
C1_ai	03=lobias						-0.2	-0.3	-0.3	-7.7	100	100	112	117	
C1_ai	04=gbr						-0.2	-0.3	-0.3	-7.7	100	100	114	116	
C1_ai	05=icp						-0.2	-0.3	-0.3	-7.7	100	100	111	115	
C1_ai	06=ycgeo						-0.2	-0.3	-0.3	-7.7	105	103	112	117	
C1_ai	07=lodroi						-0.2	-0.3	-0.3	-7.7	105	103	113	117	
C1_ai	08=geomquant						-0.2	-0.3	-0.3	-7.7	105	103	115	118	
C1_ai	09=qtbt						-0.2	-0.3	-0.3	-7.7	116	117	116	118	
C1_ai	10=planar						-0.2	-0.3	-0.3	-7.7	117	118	128	133	
C1_ai	11=idcm						-0.2	-0.3	-0.3	-7.7	117	118	128	132	
C1_ai	12=cfg						-2.9	-1.7	-2.1	-7.7	117	118	128	129	
C1_ai	13=lodshare						-2.9	-1.7	-2.1	-7.7	121	133	126	128	
C1_ai	14=intcloud						-2.9	-1.7	-2.1	-7.7	95	107	125	129	

Table 6 – Octree & lifting transform progression – C2_ai,overall

BPP Ratio [%]					BD-Rate [$\Delta\%$]						Avg. of ratio maxrssk [%]		Ratio of avg. runtime [%]	
Condition	Class	Geometry	Colour	Refl	D1	D2	Y	Cb	Cr	R	Encoder	Decoder	Encoder	Decoder
C2_ai	00=7dot1 fixes				-0.0!	-0.0!	0.0!	-0.2!	0.0!	0.0				
C2_ai	01=hls+refactors				-0.0!	-0.0!	0.0!	-0.2!	0.0!	0.0				
C2_ai	02=attrectx				-0.0!	-0.0!	-0.1!	-0.3!	-0.1!	0.0				
C2_ai	03=lobias				-0.0!	-0.0!	-0.3!	-0.6!	-0.4!	-6.9				
C2_ai	04=gbr				-0.0!	-0.0!	-0.3!	-0.6!	-0.4!	-6.9				
C2_ai	05=icp				-0.0!	-0.0!	-0.3!	-0.6!	-0.4!	-6.9				
C2_ai	06=ycgeo				-0.0!	-0.0!	-0.3!	-0.6!	-0.4!	-6.9				
C2_ai	07=lodroi				-0.0!	-0.0!	-0.3!	-0.6!	-0.4!	-6.9				
C2_ai	08=geomquant				-0.0!	-0.0!	-0.3!	-0.6!	-0.4!	-6.9				
C2_ai	09=qtbt				-1.2!	-1.2!	-0.3!	-0.6!	-0.4!	-7.0				
C2_ai	10=planar				-2.3!	-2.3!	-0.3!	-0.6!	-0.4!	-6.9				
C2_ai	11=idcm				-2.3!	-2.3!	-0.3!	-0.6!	-0.4!	-6.9				
C2_ai	12=cfg				-2.3!	-2.3!	-0.2!	-0.1!	1.5!	-6.1				
C2_ai	13=lodshare				-2.3!	-2.3!	-0.2!	-0.1!	1.5!	-6.1				
C2_ai	14=intcloud				-2.3!	-2.3!	-0.2!	-0.1!	1.5!	-6.1				

Table 7 – Octree & predicting transform progression – CW_ai,overall

Condition	Class	BPP Ratio [%]			Ref1	BD-Rate [Δ%]					R	Avg. of ratio maxrssk [%]		Ratio of avg. runtime [%]	
		Geometry	Colour	Refl		D1	D2	Y	Cb	Cr		Encoder	Decoder	Encoder	Decoder
CW_ai	00=7dot1 fixes	100.0	100.0!	100.0							100	100	106	106	
CW_ai	01=hls+refactors	100.0	100.0!	100.0							100	100	107	107	
CW_ai	02=attrectx	100.0	98.8!	100.0							100	100	106	109	
CW_ai	03=lobias	100.0	98.8!	98.3							100	100	107	107	
CW_ai	04=gbr	100.0	99.0!	98.3							100	100	107	105	
CW_ai	05=icp	100.0	83.0!	98.3							100	100	107	107	
CW_ai	06=ycgeo	100.0	83.4!	98.3							104	102	110	111	
CW_ai	07=lodroi	100.0	83.4!	98.3							104	102	106	108	
CW_ai	08=geomquant	100.0	83.4!	98.3							104	102	111	108	
CW_ai	09=qtbt	99.1	83.4!	98.3							112	112	111	111	
CW_ai	10=planar	93.2	83.4!	98.3							139	158	185	198	
CW_ai	11=idcm	93.0	83.4!	98.3							139	158	186	195	
CW_ai	12=cfg	93.0	83.4!	98.3							139	158	185	200	
CW_ai	13=lodshare	93.0	83.4!	98.3							141	167	187	199	
CW_ai	14=intcloud	93.0	83.4!	98.3							109	139	177	205	

⁷[report_tmc13v8.0-rc1_octree_predlift_apple_vs_bb.txt](#)

⁸[report_tmc13v8.0-rc1_trisoup_predlift_apple_vs_bb.txt](#)

⁹[report_tmc13v8.0-rc1_octree_raht_apple_vs_bb.txt](#)

¹⁰[report_tmc13v8.0-rc1_trisoup_raht_apple_vs_bb.txt](#)

Table 8 – Octree & predicting transform progression – CW_ai,cat1-B

Condition	Class	BPP Ratio [%]			D1	D2	BD-Rate [Δ %]			R	Avg. of ratio maxrssk [%]		Ratio of avg. runtime [%]	
		Geometry	Colour	Refl			Y	Cb	Cr		Encoder	Decoder	Encoder	Decoder
CW_ai	00=7dot1fixes	100.0									100	100	98	99
CW_ai	01=hls+refactors	100.0									100	100	106	108
CW_ai	02=attrctx	100.0									100	100	102	105
CW_ai	03=lobias	100.0									100	100	98	101
CW_ai	04=gbr	100.0									100	100	102	96
CW_ai	05=icp	100.0									100	100	100	98
CW_ai	06=ycgco	100.0									102	100	102	105
CW_ai	07=lodroi	100.0									102	100	97	101
CW_ai	08=geomquant	100.0									102	100	105	100
CW_ai	09=qtbt	100.0									106	104	103	107
CW_ai	10=planar	91.2									165	210	303	370
CW_ai	11=idcm	91.2									165	210	308	352
CW_ai	12=cfg	91.2									165	210	300	359
CW_ai	13=lodshare	91.2									165	210	303	371
CW_ai	14=intcloud	91.2									130	179	274	367

Integration 0 — v7.1 release

Includes the previous adoption of scalable decoding and bug fixes affecting v7.0. The adoption of an lod generation bug fix [4] was found to conflict with an earlier adoption. A correct (trivial) fix was implemented instead.

indent: use pragma once to avoid excessive auto indentation
attr/m50745: fix invalid array access in LoD neighbour search
attr/m51010: fix nearest neighbour search tiebreaker
attr/m51481: fix insufficient precision in lifting quantization
attr/m47352: permit partial geometry and attribute reconstruction
attr: pass aps reference rather than use many arguments in calls

Results compared to release v7.0:

Octree-LoD: [pcc-tmc13-tmc13v7.0+integration00=7dot1fixes_octree_predlift.xlsm](#)

Integration 1 — HLS & refactoring

None of the following aspects affect the coding conditions:

hls/m50745: avoid signalling variables in meaningless cases
attr: don't specialise k=0 case in (de/en)codeColorsPred
hls/m51025: support multi-frame sequences
cli: add support for encoding pointcloud sequences
ply: move ply io methods to separate compilation unit
hls/m51027: signal geometry axis order

Results compared to integration 0:

Octree-LoD: [pcc-tmc13-tmc13v7.0+integration01=hls+refactors_octree_predlift.xlsm](#)

Integration 2 — attribute zero-run contextualisation

attr/m50669: adjust context derivation for zerorun

Results compared to integration 1:

Octree-LoD: [pcc-tmc13-tmc13v7.0+integration02=attrctx_octree_predlift.xlsm](#)

Integration 3 — attribute lod neighbour bias

attr/m48894: introduce neighbour bias LoD predictor construction

Results compared to integration 2:

Octree-LoD: [pcc-tmc13-tmc13v7.0+integration03=lobias_octree_predlift.xlsm](#)

Integration 4 — attribute colour (GBR) coding order

This fixes an issue where colour components are coded in an order contrary to the description in the codec independent code points. This ordering is required by inter-component prediction.

attr: change RGB coding order to GBR

Results compared to integration 3:

Octree-LoD: [pcc-tmc13-tmc13v7.0+integration04=gbr_octree_predlift.xlsm](#)

Integration 5 — attribute inter-component prediction

attr/m49605: add inter component prediction for predicting coder

Results compared to integration 4:

Octree-LoD: [pcc-tmc13-tmc13v7.0+integration05=icp_octree_predlift.xlsm](#)

Integration 6 — attribute YCgCoR colourspace conversion for lossless

The integrated version has been updated to use the YCgCoR formula described in the codec independent code points which fixes an incorrect shift in the CE code. In order to correctly signal YCgCoR, a secondary bitdepth field has been added to indicate the bitdepth of the non-primary attribute components. Recolouring methods are updated to correctly handle non 8-bit data.

refactor: move pointset processing functions to compilation unit

refactor: move colour conversion functions to colourspace.h

attr/m49601: use attr_t (uint16_t) for all attributes

attr: refactor colourspace conversion to support other formats

attr/m49601: add support to signal chroma (secondary) bitdepth

attr/m49601: add YCgCoR colour transform support

Results compared to integration 5:

Octree-LoD: [pcc-tmc13-tmc13v7.0+integration06=ycgco_octree_predlift.xlsm](#)

Integration 7 — attribute region-wise quantization for LoD coding

The feature itself is not enabled in the common test conditions, and no configuration parameter is able to control it.

attr/m49626: add region-wise quantization support for LoD coding

Results compared to integration 6:

Octree-LoD: [pcc-tmc13-tmc13v7.0+integration07=lodroi_octree_predlift.xlsm](#)

Integration 8 — geometry quantisation

In order to reduce the memory used to represent the octree, and to reduce confusion caused by extra position state, the non-normative geometry octree position representation has been refactored.

geom/m49232: octree node geometry quantisation

geom/m50927: slice-based geometry quantization

geom: use single level based tree position representation

geom: remove reconstructed position data from PCCPointSet3

Results compared to integration 7:

Octree-LoD: [pcc-tmc13-tmc13v7.0+integration08=geomquant_octree_predlift.xlsm](#)

Integration 9 — geometry qtbt

Please see the release notes for comments on qtbt

geom/m49231: add support for non-cubic nodes with implicit qtbt
geom: fix interaction of qtbt and octree geometry quantisation

Results compared to integration 8:

Octree-LoD: [pcc-tmc13-tmc13v7.0+integration09=qtbt_octree_predlift.xlsm](#)

Integration 10 — geometry planar

geom: unify encoder and decoder internal point representation
geom/m48906: planar mode coding of octree occupancy

Results compared to integration 9:

Octree-LoD: [pcc-tmc13-tmc13v7.0+integration10=planar_octree_predlift.xlsm](#)

Integration 11 — geometry idcm duplicate points

geom/m48905: permit more than one duplicate point in IDCM nodes

Results compared to integration 10:

Octree-LoD: [pcc-tmc13-tmc13v7.0+integration11=idcm_octree_predlift.xlsm](#)

Integration 12 — CTC configuration changes

The following represent significant baseline CTC changes. They are performed last so as to better compare integrations to v7.0.

cfg/m51008: update dist2 parameter for cat1 sequences
recolour/m51510: reduce default search range to 1

Results compared to integration 11:

Octree-LoD: [pcc-tmc13-tmc13v7.0+integration12=cfg_octree_predlift.xlsm](#)

Integration 13 — non-normative attribute lod sharing

remove inappropriate copy constructors
attr: use interface to hide attribute codec internals
attr: refactor lod representation to be member of attribute codec
attr/m51092: reuse lods between attributes of same slice

Results compared to integration 12:

Octree-LoD: [pcc-tmc13-tmc13v7.0+integration13=lodshare_octree_predlift.xlsm](#)

Integration 14 — non-normative integer point positions

maths: use operator[] in Vec3<> arithmetic operators
maths: allow the usual numeric conversions between Vec3<T> and Vec3<U>
maths: be explicit in the computation type of Vec3::getNorm2 + friends
ply: perform inverse scaling during ply writing
geom: convert octree internal positions to int32_t
trisoup: convert internal positions to int32_t
pointcloud/m42611: change geometry representation from double to int32_t

Results compared to integration 13:

Octree-LoD: [pcc-tmc13-tmc13v7.0+integration14=intcloud_octree_predlift.xlsm](#)

Release v7.1

This release contains the integration of, or aspects relating to: [5, 4, 6, 7]

Release v8.0

This release contains the integration of, or aspects relating to: [8, 9, 10, 11, 12, 13, 14, 15, 16, 4, 17, 18, 19, 20, 21, 22]

The release of version 8.0 includes the following notable fixes following the release candidate, neither of which affect the CTC anchor result:

- Reverting an aspect of m50745, wherein the adopted change to make the APS search_range syntax element conditional conflicted with the requirement to use the search_range in the neighbour search, not just LoD generation.
- An integration error involving geometry scaling in the encoder.

General comments

- Colourspace conversion has been updated to handle RGB, YCgCoR, and YCbCr (Rec.709). The configuration options that control conversion for the encoder and decoder have changed:
 - A per-attribute colourMatrix option has been added. This indicates the colourspace of the coded attribute data.
 - The colorTransform option that both specified a global transform type and caused conversion to/from RGB to be performed has been removed.
 - A new global option convertPlyColourspace=1 will, after reading the ply data convert RGB attributes from RGB according to the value of colourMatrix. Before writing ply data, the use of this option will convert colour attributes back to RGB according to the bitstream value of cicp_colour_matrix_idx.
- The software now supports writing multi-frame sequences using a printf-like %d directive to act as a placeholder for the current frame number. When encoding, parameter sets are repeated for each frame, with the usual caveat that they must not change.

NB: multi-frame coding is not used to generate the CTC anchors. This is due to a requirement to send a single sequence-level bounding box, where as in the single-frame coding case, the sequence-level bounding box is calculated for each frame. Further study of is encouraged.

- In order to correctly signal YCgCoR, a second bitdepth field is added to the SPS attribute data. This should be confirmed for adoption.
- The use of in-tree geometry quantisation causes a number of issues relating to other integrations. In particular:
 - non-cubic geometry coding (qtb): parameters must be recomputed for each octree node in order to correctly mask implied occupancy bits. In order to complete the integration a fix has been applied and should be reviewed further.
 - planar mode: there may be an issue with the way planar mode inspects multiple levels of the tree if geometry quantisation changes the representation. Further study is encouraged.
 - Partial decoding / reconstruction as used by trisoup and LoD scalability: internal node positions are not necessarily correctly inverse quantised before being used by subsequent processes.
- The internal representation of geometry octree positions (node.pos) has changed. Partially coded positions are now LSB aligned, indicating the position of the node within the octree level.

- The internal component ordering for directly coding ‘RGB’ data is now Green-Blue-Red. Any code that made assumptions about the ordering must be updated.
- The software now has a notion of internal and external geometry axes. Variables with names indicating a particular axis may not relate to the external axis with the same name depending upon the configuration.
- The point cloud position representation has been changed to use `int32_t` (previously `double`). Non-normative inverse scaling is handled during ply writing. Operations on point positions that used square distances may need to be reviewed to check or specify the computation type (eg, `Vec3<>::getDist2()`).
- CTC configurations are provided for the following test conditions:
 - octree + pred/lift transforms [C1, C2, CW, CY]
 - octree + RAHT [C1, C2]
 - trisoup + pred/lift transforms [C2]
 - trisoup + RAHT [C2]
- A review of the CTC conditions is still required for the next meeting, since several test points cause issues in calculating reportable results. In particular:
 - some sequences have so few points that decoding is instantaneous (causes issues for geometric mean).
 - some trisoup test points are lossless.
 - some trisoup geometry configuration results are identical over multiple test points causing the failure of BD-rate calculations.
 - the current sequence categorisation does not facilitate identifying the type of content providing compression gains or losses.
- The software may be configured to output either ASCII or binary ply files using the `outputBinaryPly` option. Be aware that under certain test conditions this will affect the re-scaled geometry values due to the difference in precision of the two representations. Anchor results have been generated using the ASCII option.

Changes between v7.0 and v8.0-rc1

attr: remove spurious logging of `zero_cnt`
 hls/m50745: `aps.search_range` has meaning when scalable lifting enabled
 hls: refactor LoD presence test into `aps.lodParametersPresent()`
 attr: make LoD sharing determination more robust
 geom: fix encoder geometry scaling
 release: update version to 8.0 (tag: `release-v8.0`)

Changes between v7.0 and v8.0-rc1

indent: use `pragma once` to avoid excessive auto indentation
 attr/m50745: fix invalid array access in LoD neighbour search
 attr/m51010: fix nearest neighbour search tiebreaker
 attr/m51481: fix insufficient precision in lifting quantization
 attr/m47352: permit partial geometry and attribute reconstruction
 attr: pass `aps` reference rather than use many arguments in calls
 release: update version to 7.1 (tag: `release-v7.1`)
 hls/m50745: avoid signalling variables in meaningless cases
 attr: don't specialise `k=0` case in `(de/en)codeColorsPred`

hls/m51025: support multi-frame sequences
 cli: add support for encoding pointcloud sequences
 ply: move ply io methods to separate compilation unit
 hls/m51027: signal geometry axis order
 attr/m50669: adjust context derivation for zerorun
 attr/m48894: introduce neighbour bias LoD predictor construction
 attr: change RGB coding order to GBR
 attr/m49605: add inter component prediction for predicting coder
 refactor: move pointset processing functions to compilation unit
 refactor: move colour conversion functions to colourspace.h
 attr/m49601: use attr_t (uint16_t) for all attributes
 attr: refactor colourspace conversion to support other formats
 attr/m49601: add support to signal chroma (secondary) bitdepth
 attr/m49601: add YCgCoR colour transform support
 attr/m49626: add region-wise quantization support for LoD coding
 geom/m49232: octree node geometry quantisation
 geom/m50927: slice-based geometry quantization
 geom: use single level based tree position representation
 geom: remove reconstructed position data from PCCPointSet3
 geom/m49231: add support for non-cubic nodes with implicit qtbt
 geom: fix interaction of qtbt and octree geometry quantisation
 geom: unify encoder and decoder internal point representation
 geom/m48906: planar mode coding of octree occupancy
 geom/m48905: permit more than one duplicate point in IDCM nodes
 cfg/m51008: update dist2 parameter for cat1 sequences
 recolour/m51510: reduce default search range to 1
 remove inappropriate copy constructors
 attr: use interface to hide attribute codec internals
 attr: refactor lod representation to be member of attribute codec
 attr/m51092: reuse lods between attributes of same slice
 maths: use operator[] in Vec3<T> arithmetic operators
 maths: allow the usual numeric conversions between Vec3<T> and Vec3<U>
 maths: be explicit in the computation type of Vec3::getNorm2 + friends
 ply: perform inverse scaling during ply writing
 geom: convert octree internal positions to int32_t
 trisoup: convert internal positions to int32_t
 pointcloud/m42611: change geometry representation from double to int32_t
 maths: don't emit newline in formatted output operators
 release: update version to 8.0-rc1 (tag: release-v8.0-rc1)

Location of changes

README.md	2 +-
cfg/octree-lifft-ctc-lossless-geom-lossy-attrs.yaml	16 +-
cfg/octree-lifft-ctc-lossy-geom-lossy-attrs.yaml	16 +-
cfg/octree-predt-ctc-lossless-geom-lossless-attrs.yaml	21 +-
cfg/octree-predt-ctc-lossless-geom-nearlossless-attrs.yaml	20 +-
cfg/octree-raht-ctc-lossless-geom-lossy-attrs.yaml	15 +-
cfg/octree-raht-ctc-lossy-geom-lossy-attrs.yaml	15 +-
cfg/sequences-cat1.yaml	44 +-

cfg/sequences-cat3.yaml	10 +
cfg/trisoup-lifft-ctc-lossy-geom-lossy-attrs.yaml	7 +-
cfg/trisoup-raht-ctc-lossy-geom-lossy-attrs.yaml	6 +-
doc/README.options.md	160 +++++-
doc/README.usage.md	12 +-
doc/mpeg-pcc-tmc13-sw-manual.tex	6 +-
tmc3/Attribute.h	93 ++++
tmc3/AttributeCommon.cpp	113 ++++
tmc3/AttributeCommon.h	71 +++
tmc3/AttributeDecoder.cpp	309 +++++-----
tmc3/AttributeDecoder.h	29 +-
tmc3/AttributeEncoder.cpp	344 ++++++-----
tmc3/AttributeEncoder.h	27 +-
tmc3/CMakeLists.txt	12 +-
tmc3/OctreeNeighMap.cpp	103 +---
tmc3/OctreeNeighMap.h	57 +-
tmc3/PCCMath.h	256 ++++++---
tmc3/PCCMisc.h	7 +-
tmc3/PCCPointSet.h	516 +------
tmc3/PCCTMC3Common.h	318 ++++++---
tmc3/PCCTMC3Decoder.h	48 +-
tmc3/PCCTMC3Encoder.h	36 +-
tmc3/TMC3.cpp	625 ++++++-----
tmc3/TMC3.h	6 -
tmc3/colourspace.h	128 +++++
tmc3/decoder.cpp	114 +---
tmc3/encoder.cpp	145 +----
tmc3/entropydirac.h	11 +
tmc3/geometry.h	7 +
tmc3/geometry_intra_pred.cpp	79 +-+
tmc3/geometry_intra_pred.h	4 +-
tmc3/geometry_octree.cpp	241 ++++++-
tmc3/geometry_octree.h	142 ++++
tmc3/geometry_octree_decoder.cpp	933 ++++++-----
tmc3/geometry_octree_encoder.cpp	992 ++++++-----
tmc3/geometry_trisoup.h	8 +-
tmc3/geometry_trisoup_decoder.cpp	44 +-
tmc3/geometry_trisoup_encoder.cpp	22 +-
tmc3/hls.h	116 +++-
tmc3/io_hls.cpp	205 +++++-
tmc3/io_hls.h	12 +-
tmc3/partitioning.cpp	6 +-
tmc3/ply.cpp	494 ++++++-----
tmc3/ply.h	85 +++
tmc3/{PCCPointSetProcessing.h => pointset_processing.cpp}	188 +++++-
tmc3/pointset_processing.h	188 ++++++
tmc3/quantization.cpp	80 +-+
tmc3/quantization.h	100 +++-
tmc3/tables.cpp	31 +-
tmc3/tables.h	2 +
tools/ply-merge.cpp	23 +-

59 files changed, 6016 insertions(+), 1704 deletions(-)

References

- [1] 3DG, “Common Test Conditions for PCC,” ISO/IEC JTC1/SC29/WG11, 128th meeting, Geneva, Tech. Rep. w18883, Oct. 2019.
- [2] —, “G-PCC Test Model v7,” ISO/IEC JTC1/SC29/WG11, 127th meeting, Gothenburg, Tech. Rep. w18664,

Jul. 2019.

- [3] —, “G-PCC codec description,” ISO/IEC JTC1/SC29/WG11, 127th meeting, Gothenburg, Tech. Rep. w18673, Jul. 2019.
- [4] S. Kuma and O. Nakagami, “[G-PCC] syntax modification for lifting attribute coding,” ISO/IEC JTC1/SC29/WG11, 128th meeting, Geneva, Tech. Rep. m50745, Oct. 2019.
- [5] O. Nakagami and S. Kuma, “[G-PCC] (New proposal) Spatial scalability support for G-PCC,” ISO/IEC JTC1/SC29/WG11, 126th meeting, Geneva, Tech. Rep. m47352, Mar. 2019.
- [6] Z. Gao, D. Flynn, A. Tourapis, and K. Mammou, “[G-PCC][New proposal] Improved implementation of the Prediction and Lifting schemes,” ISO/IEC JTC1/SC29/WG11, 128th meeting, Geneva, Tech. Rep. m51010, Oct. 2019.
- [7] A. Zaghetto, S. Kuma, O. Nakagami, D. B. Graziosi, and A. Tabatabai, “[G-PCC] Lifting quantization bug-fix,” ISO/IEC JTC1/SC29/WG11, 128th meeting, Geneva, Tech. Rep. m51481, Oct. 2019.
- [8] D. Flynn and S. Lasserre, “PCC TMC3 Software optimisation and practical considerations,” ISO/IEC JTC1/SC29/WG11, 122nd meeting, San Diego, Tech. Rep. m42611, Apr. 2018.
- [9] Q. Zhang, Y. Shao, J. Wang, G. Li, and S. Liu, “[G-PCC][New proposal] An enhanced prediction scheme for Cat3 point clouds,” ISO/IEC JTC1/SC29/WG11, 127th meeting, Gothenburg, Tech. Rep. m48894, Jul. 2019.
- [10] S. Lasserre and S. Flynn, “[GPCC][new input] Improved geometry coding for duplicated points,” ISO/IEC JTC1/SC29/WG11, 127th meeting, Gothenburg, Tech. Rep. m48905, Jul. 2019.
- [11] —, “[GPCC][new input] Planar mode in octree-based geometry coding,” ISO/IEC JTC1/SC29/WG11, 127th meeting, Gothenburg, Tech. Rep. m48906, Jul. 2019.
- [12] X. Zhang, W. Gao, S. Yea, and S. Liu, “[G-PCC][New proposal] Implicit geometry partition for point cloud coding,” ISO/IEC JTC1/SC29/WG11, 127th meeting, Gothenburg, Tech. Rep. m49231, Jul. 2019.
- [13] S. Yea, A. Vosoughi, and S. Liu, “[G-PCC][New Proposal] Lossless Color Transform for Attribute Coding,” ISO/IEC JTC1/SC29/WG11, 127th meeting, Gothenburg, Tech. Rep. m49601, Jul. 2019.
- [14] —, “[G-PCC][New Proposal] Inter-Channel Prediction for Attribute Coding,” ISO/IEC JTC1/SC29/WG11, 127th meeting, Gothenburg, Tech. Rep. m49605, Jul. 2019.
- [15] C. D. Han and T. Sugio, “[G-PCC] Region-wise Quantization Control in Attribute Coding,” ISO/IEC JTC1/SC29/WG11, 127th meeting, Gothenburg, Tech. Rep. m49626, Jul. 2019.
- [16] J. Chen, B. Wang, W. Wang, L. Yu, P. Li, and X. Zheng, “[G-PCC][New proposal] An improved context selection of a flag in attribute residuals coding,” ISO/IEC JTC1/SC29/WG11, 128th meeting, Geneva, Tech. Rep. m50669, Oct. 2019.
- [17] X. Zhang, W. Gao, and S. Liu, “[G-PCC][EE13.6 related] slice based geometry quantization,” ISO/IEC JTC1/SC29/WG11, 128th meeting, Geneva, Tech. Rep. m50927, Oct. 2019.
- [18] Z. Gao, D. Flynn, and K. Mammou, “[G-PCC] [New proposal] Improved encode parameters for Lifting attribute coding,” ISO/IEC JTC1/SC29/WG11, 128th meeting, Geneva, Tech. Rep. m51008, Oct. 2019.
- [19] D. Flynn and K. Mammou, “[G-PCC][New proposal] G-PCC multi-frame sequence coding,” ISO/IEC JTC1/SC29/WG11, 128th meeting, Geneva, Tech. Rep. m51025, Oct. 2019.
- [20] —, “[G-PCC][New proposal] G-PCC geometry swizzling,” ISO/IEC JTC1/SC29/WG11, 128th meeting, Geneva, Tech. Rep. m51027, Oct. 2019.
- [21] H. Hur and S. Oh, “[G-PCC][New Proposal] on possible reduction of attribute coding time for cat3-fused dataset,” ISO/IEC JTC1/SC29/WG11, 128th meeting, Geneva, Tech. Rep. m51092, Oct. 2019.
- [22] A. Vosoughi, S. Yea, S. Wenger, and S. Liu, “[G-PCC][VE13.1 Report] Improved color transfer,” ISO/IEC JTC1/SC29/WG11, 128th meeting, Geneva, Tech. Rep. m51510, Oct. 2019.