

**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 w19085**

**Document type: Approved WG 11 document**

**Title:**

**Status: Approved**

**Date of document: 2020-06-01**

**Source: 3DG**

**Expected action:**

**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 JTC1/SC29/WG11**

**CODING OF MOVING PICTURES AND AUDIO**

**ISO/IEC JTC1/SC29/WG11 W19325**

**April 2020, Alpbach**

|  |  |
| --- | --- |
| **Source** | **3DG** |
| **Title** | **[V-PCC] V-PCC Test Model v10** |

**[V-PCC] V-PCC Test Model v10**

# Introduction

Based on the MPEG-3DG-V-PCC decisions, a new release of the V-PCC test model 10.0 has been delivered.

# New integrations

The next table presents the integrations that have been made in this new release.

# Software manual

## Obtaining the software

The authoritative location of the software is the following git repository: <http://mpegx.int-evry.fr/software/MPEG/PCC/TM/mpeg-pcc-tmc2.git>.

Each released version may be identified by a version control system tag in the form release-v9.0.

An example:

$ git clone http://mpegx.int-evry.fr/software/MPEG/PCC/mpeg-pcc-tmc2.git

$ cd mpeg-pcc-tmc2

It is strongly advised to obtain the software using the version control system rather than to download a zip (or other archive) of a release. The build system uses the version control system to accurately identify the version being built.

Bugs should be reported on the issue tracker set up at <http://mpegx.int-evry.fr/software/MPEG/PCC/TM/mpeg-pcc-tmc2/issues/new>.

The codec is supported on Linux, OSX and Windows platforms. The build configuration is managed using CMake. It is strongly advised to build the software in a separate build directory.

Bash scripts can be use to build mpeg-pcc-dmetric project: build.sh to build solutions and clear.sh to clean.

**Linux**

mkdir build

cd build

cmake .. make

../bin/PccAppEncoder --help

../bin/PccAppDecoder --help

../bin/PccAppMetrics –help

**OSX**

mkdir build

cd build

cmake .. -G Xcode

xcodebuild

../bin/PccAppEncoder --help

../bin/PccAppDecoder --help

../bin/PccAppMetrics --help

As an alternative, the generated XCode project may be opened and built from XCode itself.

**Windows**

md build

cd build

cmake .. -G "Visual Studio 15 2017 Win64"

Open the generated visual studio solution to build it.

The common test conditions use HM reference software to encode the created videos. To respect the CTC, we must use the HM: HM-16.20+SCM-8.8 and apply a patch to this version to activate the 3D motion estimation. The patch can be found in the subfolder: mpeg-pcc-tmc2/dependencies/hm-modification/pcc\_me-ext\_for\_HM-16.20+SCM-8.8.patch.

The next command lines could be used to download HM reference software and apply patch:

svn checkout https://hevc.hhi.fraunhofer.de/svn/svn\_HEVCSoftware/tags/HM-16.20+SCM-8.8/external/HM-16.20+SCM-8.8+3DMC;

cd external/HM-16.20+SCM-8.8+3DMC

svn patch ../../mpeg-pcc-tmc2/dependencies/hm-modification/pcc\_me-ext\_for\_HM-16.20+SCM-8.8.patch

## Using the codec

../bin/PccAppEncoder [--help] [-c config.cfg] [--parameter=value]

../bin/PccAppDecoder [--help] [--parameter=value]

../bin/PccAppMetrics [--help] [--parameter=value]

### Principle

The encoder takes as input a PLY file describing a point cloud with integer positions and, optionally, per-point integer colour attributes.

The output of the encoder is a binary bitstream encapsulated using the V-PCC annex-B format.

Conversely, the decoder takes as input a compressed bitstream file in V-PCC annex-B format and produces a reconstructed PLY file with position and any present attribute values.

The software may be configured using either command line arguments or from a configuration file specified using the -c|--config= option.

Sample configuration files are provided in the cfg/ directory.

Parameters are set by the last value encountered on the command line. Therefore if a setting is set via a configuration file, and then a subsequent command line parameter changes that same setting, the command line parameter value will be used.

### Common test condition configurations

The configuration files stored in the ./cfg/ sub-folder could be used to perform the V-PCC common test conditions (CTC) experiments. An example of the usage of this file could be found in ./test/runme\_linux.sh.

The reference software configuration not defined the CTC, please validated your experimentes based on the w18665 - V-PCC common test conditions document.

### Examples

This new release is available in MPEG 3DG V-PCC git repository: <http://mpegx.int-evry.fr/software/MPEG/PCC/TM/mpeg-pcc-tmc2.git>

Tag: <http://mpegx.int-evry.fr/software/MPEG/PCC/TM/mpeg-pcc-tmc2/tags/release-v10.0>

#### Encoder

The next command line encodes one streams:

./bin/PccAppEncoder \

--config=./cfg/common/ctc-common.cfg \

--config=./cfg/condition/ctc-all-intra.cfg \

--config=./cfg/sequence/queen.cfg \

--config=./cfg/rate/ctc-r1.cfg \

--colorTransform=0 \

--configurationFolder=./cfg/ \

--uncompressedDataFolder=./People/ \

--colorSpaceConversionPath=./external/HDRTools/bin/HDRConvert \

--videoEncoderPath=./external/HM-16.20+SCM-8.8+3DMC/bin/TAppEncoderHighBitDepthStatic\

--videoEncoderOccupancyMapPath=./external/HM-16.20+SCM-8.8+3DMC/bin/TAppEncoderHighBitDepthStati --compressedStreamPath=./S22C2AI\_queen/S22C2AIR01\_queen.bin \

--frameCount=32

To compute the metrics in the encode, the normal of the source point cloud must be given to the encoder. The next parameter must be added to the previous command:

--normalDataPath=./People/Technicolor/queen\_n/frame\_%04d\_n.ply

#### Decoder

The next command line decodes one streams:

./bin/PccAppDecoder \

--startFrameNumber=0000 \

--compressedStreamPath=./S22C2AI\_queen/S22C2AIR01\_queen.bin \

--reconstructedDataPath=./S22C2AI\_queen/S22C2AIR01\_queen\_dec\_%04d.ply \ --videoDecoderPath=

./external/HM-16.20+SCM-8.8+3DMC/bin/TAppDecoderHighBitDepthStatic \

--videoDecoderOccupancyMapPath=\

./external/HM-16.20+SCM-8.8+3DMC/bin/TAppDecoderHighBitDepthStatic \

--colorSpaceConversionPath=./external/HDRTools/bin/HDRConvert \

--inverseColorSpaceConversionConfig=./cfg/hdrconvert/yuv420torgb444.cfg \

--nbThread=1 \

--colorTransform=0

To compute the metrics in the decoder, the normal of the source point cloud and the source PLY must be given to the decoder. The next parameter must be added to the previous command:

--config=./cfg/sequence/queen.cfg \

--uncompressedDataFolder=./People/ \

--normalDataPath=./People/Technicolor/queen\_n/frame\_%04d\_n.ply

#### Metrics

PccAppMetrics could be used to test the PccLibMetrics. For CTC experiments, it’s sugested to used mpeg-pcc-dmetrics: http://mpegx.int-evry.fr/software/MPEG/PCC/mpeg-pcc-dmetric.git.

For example, mpeg-pcc-dmetric and PccAppMetric could be used with the next command line:

../bin/PccAppMetrics \

--uncompressedDataPath=

./People/8i/8iVFBv2/longdress/Ply/longdress\_vox10\_1051.ply\

--reconstructedDataPath=./S26C2AIR01\_longdress\_dec\_1051.ply \

--normalDataPath=./People/8i/longdress\_n/longdress\_vox10\_1051\_n.ply \

--resolution=1023 \

--frameCount=1

./mpeg-pcc-demetric/test/pc\_error \

--fileA=./People/8i/8iVFBv2/longdress/Ply/longdress\_vox10\_1051.ply \

--fileB=S26C2AIR01\_longdress\_dec\_1051.ply \

--inputNorm=./People/8i/longdress\_n/longdress\_vox10\_1051\_n.ply \ --color=1 \

--resolution= 1023

More examples of running could be found in ./test/runme\_linux.sh. These examples can be start based on your system with the following scripts:

./test/runme\_linux.sh

./test/runme\_windows.bat

./test/runme\_osx.sh

## General options

The next table shows the parameters of the encoder, decoder and metrics programs.

### Encoder parameters

|  |  |
| --- | --- |
| **Parameter=Value** | **Usage** |
| –help=0 | This help text |
| **Global** |  |
| -c,–config=... | Configuration file name |

–configurationFolder=“” Folder where the configuration files

are stored, use for cfg relative

paths.

|  |  |
| --- | --- |
| –uncompressedDataFolder=“” | Folder where the uncompress input data are stored, use for cfg relative paths. |
| –uncompressedDataPath=“” | Input pointcloud to encode. Multi-frame sequences may be represented by %04i |
| –compressedStreamPath=“” | Output  compressed bitstream |
| –reconstructedDataPath=“” | Output decoded pointcloud. Multi-frame sequences may be represented by %04i |
| –startFrameNumber=0 | First frame number in sequence to encode/decode |
| –frameCount=300 | Number of frames to encode |
| –groupOfFramesSize=32 | Random access period |
| –colorTransform=1 | The colour transform to be applied:  0: none  1: RGB to YCbCr (Rec.709) |
| –colorSpaceConversionPath=“” | Path to the HDRConvert. If unset, an internal color space conversion is used |
| –colorSpaceConversionConfig=“” | HDRConvert configuration file used for  RGB444 to YUV420 conversion |
| –inverseColorSpaceConversionConfig=“” | HDRConvert configuration file used for  YUV420 to RGB444 conversion |
| –videoEncoderPath=“” | HM video encoder executable |
| –videoEncoderOccupancyMapPath=“” | HM lossless video encoder executable for occupancy map |
| –nbThread=1 | Number of thread used for parallel processing |
| –keepIntermediateFiles=0 | Keep intermediate files: RGB, YUV and bin |
| **Encoder** |  |
| –nnNormalEstimation=16 | Number of points used for normal estimation |
| –maxNNCountRefineSegmentation=256 | Number of nearest neighbors used during segmentation refinement |
| –iterationCountRefineSegmentation=100 | Number of iterations performed during segmentation refinement |
| –occupancyResolution=16 | Resolution T of the occupancy map |

–minPointCountPerCCPatchSegmentation=16 Minimum number of points for a connected component to be retained as a patch

–maxNNCountPatchSegmentation=16 Number of nearest neighbors used during connected components

Maximum depth per patch

extraction

–surfaceThickness=4

Surfacethickness

–maxAllowedDepth=255

–maxAllowedDist2MissedPointsDetection=9 Maximum distance for a point to be ignored during missed point detection

–maxAllowedDist2MissedPointsSelection=1 Maximum distance for a point to be ignored during missed points

selection

|  |  |
| --- | --- |
| –lambdaRefineSegmentation=3 | Controls the smoothness of the patch boundaries during segmentation refinement |
| –minimumImageWidth=1280 | Minimum width of packed patch frame |
| –minimumImageHeight=1280 | Minimum height of packed patch frame |
| –maxCandidateCount=4 | Maximum nuber of candidates in list L |
| –occupancyPrecision=4 | Occupancy map B0 precision |
| –occupancyMapVideoEncoderConfig=“” | Occupancy map encoder config file |
| –occupancyMapQP=8 | QP for compression of occupancy map video |
| –useOccupancyMapVideo=1 | compress occupancy map with video codec |
| –neighborCountSmoothing=64 | todo(kmammou) |
| –radius2Smoothing=64 | todo(kmammou) |
| –radius2BoundaryDetection=64 | todo(kmammou) |
| –thresholdSmoothing=64 | todo(kmammou) |
| –gridSmoothing=1 | Enable grid smoothing |
| –thresholdColorSmoothing=10 | Threshold of color smoothing |
| –thresholdLocalEntropy=4.5 | Threshold of local entropy |
| –radius2ColorSmoothing=64 | Redius of color smoothing |
| –neighborCountColorSmoothing=64 | Neighbor count for color smoothing |
| –flagColorSmoothing=0 | Enable color smoothing |
| –thresholdColorPreSmoothing=10 | Threshold of color pre-smoothing |
| –thresholdColorPreSmoothing | Threshold of color pre-smoothing local |
| LocalEntropy=4.5 | entropy |
| –radius2ColorPreSmoothing=64 | Redius of color pre-smoothing |
| –neighborCountColorPreSmoothing=64 | Neighbor count for color pre-smoothing |
| –flagColorPreSmoothing=1 | Enable color pre-smoothing |
| –bestColorSearchRange=0 | todo(kmammou) |
| –geometryQP=28 | QP for compression of geometry video |
| –textureQP=43 | QP for compression of texture video |
| –geometryConfig=“” | HM configuration file for geometry compression |
| –geometryD0Config=“” | HM configuration file for geometry D0 compression |
| –geometryD1Config=“” | HM configuration file for geometry D1 compression |
| –textureConfig=“” | HM configuration file for texture compression |
| –losslessGeo=0 | Enable lossless encoding of geometry |
| –losslessTexture=0 | Enable lossless encoding of texture |
| –noAttributes=0 | Disable encoding of attributes |
| –losslessGeo444=0 | Use 4444 format for lossless geometry |
| –useMissedPointsSeparateVideo=0 | compress missed point with video codec |
| –geometryMPConfig=“” | HM configuration file for missed points geometry compression |
| –textureMPConfig=“” | HM configuration file for missed points texture compression |
| –absoluteD1=1 | Absolute D1 |
| –constrainedPack=1 | Temporally consistent patch packing |
| –binArithCoding=1 | Binary arithmetic coding |
| –testLevelOfDetail=0 | Force non-zero level of detail for testing |
| –testLevelOfDetailSignaling=0 | Test the patch resolution signaling with pseudo-random values |
| –groupDilation=1 | Group Dilation |
| –textureDilationOffLossless=1 | Group Dilation |
| –enhancedDeltaDepthCode=0 | Use enhanced-delta-depth code |
| –patchColorSubsampling=0 | Enable per patch color sub-sampling |
| –deltaCoding=1 | Delta meta-data coding |
| –projectionMode=0 | projectionMode 0:min, 1:max, 2:adaptive frame and patch, 3:adaptive patch (all frames) |
| –oneLayerMode=0 | Use one layer mode |
| –singleLayerPixelInterleaving=0 | Use single layer pixel interleaving |
| –sixDirectionMode=1 | Use Six Direction Projection mode |
| –surfaceSeparation=0 | surface separation |
| –packingStrategy=1 | Patches packing strategy(0: anchor packing, 1/2/3(default): flexible packing with 2, 4, or 8 orientations) |
| –safeGuardDistance=0 | Number of empty blocks that must exist between the patches (default=1) |
| –textureBGFill=1  --lossyMissedPointsPatch=0  –minNormSumOfInvDist4MP  -- Selection=0.35  –lossyMppGeoQP=4  –globalPatchAllocation=0  –apply3dMotionCompensation=1  –geometry3dCoordinatesBitdepth=10  –geometryNominal2dBitdepth=8  –nbPlrmMode=0  –patchSize=0  –enhancedProjectionPlane=1  –minWeightEPP=0.6  –additionalProjectionPlaneMode=0 | Selects the background filling operation for texture only (0: anchor’s dilation,  1(default): push-pull algorithm)  Lossy missed points patch(0: no lossy  missed points patch, 1: enable lossy  missed points patch (default=0)  Minimum normalized sum of inverse distance for missed points selection: double value between 0.0 and 1.0 (default=0.35)  QP value for geometry in lossy missed  points patch (default=4)  Global temporally consistent patch  allocation. (0: anchor’s packing method(default), 1: gpa algorithm)  Use auxilliary information for 3d motion  compensation.(0: conventional video coding, 1: 3D motion compensated)  Bit depth of geomtery 3D coordinates  Bit depth of geometry 2D  Number of PLR mode  Size of Patch for PLR  Use enhanced Projection Plane  (0: OFF, 1: ON)  Minimum value  additional Projection Plane Mode 0:none  1:Y-Axis 2:X-Axis 3:Z-Axis 4:All-Axis  5:apply to portion  The value determines the partial point  cloud. It’s available with only  additionalProjectionPlaneMode(5) |
| **Metrics** |  |
| –computeChecksum=1 | Compute checksum |
| –computeMetrics=1 | Compute metrics |
| –normalDataPath=“” | Input pointcloud to encode. Multi-frame sequences may be represented by %04i |
| –resolution=1023 | Specify the intrinsic resolution |
| –dropdups=2 | 0(detect), 1(drop), 2(average) subsequent points with same coordinates |
| –neighborsProc=1 | 0(undefined), 1(average), 2(weighted average), 3(min), 4(max) neighbors with same geometric distance |
| **Decoder parameters** |  |
| **Parameter=Value** | **Usage** |
| –help=0 | This help text |
| **Global** |  |
| -c,–config=... | Configuration file name |
| –compressedStreamPath=“” | Input  compressed bitstream |
| –reconstructedDataPath=“” | Output decoded pointcloud. Multi-frame sequences may be represented by %04i |

–startFrameNumber=0 Fist frame number in sequence to encode/decode

|  |  |
| --- | --- |
| –colorTransform=1 | The colour transform to be applied:  0: none  1: RGB to YCbCr (Rec.709) |
| –colorSpaceConversionPath=“” | Path to the HDRConvert. If unset, an internal color space conversion is used |
| –inverseColorSpaceConversionConfig=“” | HDRConvert configuration file used for  YUV420 to RGB444 conversion |
| –videoDecoderPath=“” | HM video decoder executable |
| –videoDecoderOccupancyMapPath=“” | HM lossless video decoder executable for occupancy map |
| –nbThread=1 | Number of thread used for parallel processing |
| –keepIntermediateFiles=0 | Keep intermediate files: RGB, YUV and bin |
| **Metrics** |  |
| –testLevelOfDetailSignaling=0 | Disable patch sampling resolution scaling; use in conjunction with same parameter in encoder |
| –patchColorSubsampling=0 | Enable per-patch color up-sampling |
| **Metrics** |  |
| –computeChecksum=1 | Compute checksum |
| –computeMetrics=1 | Compute metrics |
| –uncompressedDataFolder=“” | Folder where the uncompress input data are stored, use for cfg relative paths. |
| –startFrameNumber=0 | Fist frame number in sequence to encode/decode |
| –frameCount=0 | Number of frames to encode |
| –groupOfFramesSize=32 | Random access period |
| –uncompressedDataPath=“” | Input pointcloud to encode. Multi-frame sequences may be represented by %04i |
| –reconstructedDataPath=“” | Output decoded pointcloud. Multi-frame sequences may be represented by %04i |
| –normalDataPath=“” | Input pointcloud to encode. Multi-frame sequences may be represented by %04i |
| –resolution=1023 | Specify the intrinsic resolution |
| –dropdups=2 | 0(detect), 1(drop), 2(average) subsequent points with same coordinates |
| –neighborsProc=1 | 0(undefined), 1(average), 2(weighted average), 3(min), 4(max) neighbors with same geometric distance |
| –nbThread=0 | Number of thread used for parallel processing |
| –minimumImageHeight=0 | Ignore parameter |
| –flagColorPreSmoothing=0 | Ignore parameter |
| –surfaceSeparation=0 | Ignore parameter |

### Metrics parameters

|  |  |
| --- | --- |
| **Parameter=Value** | **Usage** |
| –help=0 | This help text |
| –computeChecksum=1 | Compute checksum |
| –computeMetrics=1 | Compute metrics |
| –startFrameNumber=0 | Fist frame number in sequence to |
|  | encode/decode |
| –frameCount=0 | Number of frames to encode |
| –uncompressedDataPath=“” | Input pointcloud to encode. Multi-frame |
|  | sequences may be represented by %04i |
| –reconstructedDataPath=“” | Output decoded pointcloud. Multi-frame |
|  | sequences may be represented by %04i |
| –normalDataPath=“” | Input pointcloud to encode. Multi-frame |
|  | sequences may be represented by %04i |
| –resolution=1023 | Specify the intrinsic resolution |
| –dropdups=2 | 0(detect), 1(drop), 2(average) subsequent |
|  | points with same coordinates |
| –neighborsProc=1 | 0(undefined), 1(average), 2(weighted |
|  | average), 3(min), 4(max) neighbors with |
|  | same geometric distance |
| –nbThread=0 | Number of thread used for parallel |
|  | processing |
| –minimumImageHeight=0 | Ignore parameter |
| –flagColorPreSmoothing=0 | Ignore parameter |
| –surfaceSeparation=0 | Ignore parameter |

# References

1. Common test conditions for point cloud compression, ISO/IEC JTC1/SC29 WG11 Doc. w18665, Gothenburg, July 2019.