INTERNATIONAL ORGANIZATION FOR STANDARDIZATION ORGANISATION INTERNATIONALE DE NORMALISATION ISO/IEC JTC 1/SC 29/WG 7 MPEG CODING FOR 3D GRAPHICS AND HAPTICS

ISO/IEC JTC 1/SC 29/WG 7 nXXXXX
October 2022, Mainz

Title:	mpeg-pcc-dmetric v0.14.0 user manual	
Source:	WG 7, MPEG CODING FOR 3D GRAPHICS AND HAPTICS	
Purpose:	User manual	
Status:	Draft	
Author(s):	David Flynn	*davidflynn@apple.com*
	Julien Ricard	*julien.ricard@interdital.com*
	Jonathan Taquet	*jonathant@xiaomi.com*

mpeg-pcc-dmetric v0.14.0 user manual

Abstract

This document is a user manual describing usage of reference software for the mpeg-pcc-dmetric project. It applies to version 0.14.0 of the software.

Contents

1 General Information

Reference software is being made available to provide a reference implementation of the MPEG 3DG G-PCC and V-PCC standard being developed by MPEG (ISO/IEC SC29 WG11). One of the main goals of the reference software is to provide a basis upon which to conduct experiments in order to determine which coding tools provide desired coding performance. It is not meant to be a particularly efficient implementation of anything, and one may notice its apparent unsuitability for a particular use. It should not be construed to be a reflection of how complex a production-quality implementation of a future standards would be.

This document aims to provide guidance on the usage of the reference software. It is widely suspected to be incomplete and suggestions for improvements are welcome. Such suggestions and general inquiries may be sent to the general MPEG 3DGC email reflector at mpeg-3dgc@gti.ssr.upm.es (registration required).

1.1 Bug reporting

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

2 Obtaining the software

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

Each released version may be identified by a version control system tag in the form `release-v0.14.0`.

An example:

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

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

3 Building

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.

3.1 Build scripts

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

3.2 Build manually

Standard CMake build commands can be used to build the software depending on the system you used.

3.2.1 Linux

```
cmake ./source -B build/Release
cmake --build build/Release --config Release
```

3.2.2 OSX

```
cmake ./source -B build/Release
cmake --build build/Release --config Release
```

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

3.2.3 Windows

```
cmake ./source -B build/Release
cmake --build build/Release --config Release
```

Open the generated visual studio solution to build it.

4 Using the mpeg-pcc-dmetric

4.1 Usage

```
./test/pc_error  [--help] [-c config.cfg] [--parameter=value]
```

The metrics takes as input a PLY files: source(A), test(B) and soruce normal(N) and compute the distance between A and B accordring the normal stored in N.

The outputs are writing in the terminal as trace and can be catch in log files.

4.2 Examples

```
./test/pc_error --help

./test/pc_error
 --fileA=./queen/frame_0010.ply
 --fileB=./S22C2AIR01_queen_dec_0010.ply
 --inputNorm=./queen_n/frame_0010_n.ply
 --color=1
 --resolution=1023
```

5 General options

The parsing options process have been updated to uniformize the PCC softwares and used: dependencies/program-options-lite. This library defined a parsing process different than the Boost library, previously used

The command line options must be updated and:

- the short options without parameters must be updated and now take an argument: (“-c” => “-c 1”)
- the long options are required to use the “–option=value” form, rather than the previous “–option value” form.

The next table presents the software options.

Parameter	Value	Usage
–help=0	0	This help text
-a, –fileA=“”	“”	Input file 1, original version
-b, –fileB=“”	“”	Input file 2, processed version
-n, –inputNorm=“”	“”	File name to import the normals of original point cloud, if different from original file 1n
-s, –singlePass=0	0	Force running a single pass, where the loop is

		over the original point cloud
-d, –hausdorff=0	0	Send the Haursdorff metric as well
-c, –color=0	0	Check color distortion as well
-l, –lidar=0	0	Check lidar reflectance as well
-r, –resolution=0	0	Specify the intrinsic resolution
–dropdups=2	2	0(detect), 1(drop), 2(average) subsequent points with same coordinates
–neighborsProc=1	1	0(undefined), 1(average), 2(weighted average), 3(min), 4(max) neighbors with same geometric distance
–averageNormals=1	1	0(undefined), 1(average normal based on neighbors with same geometric distance)
–mseSpace=1	1	colour space used for mse calculation 0(identity), 1(Rec. ITU-R BT.709), 8(YCgCo-R)
–nbThreads=1	1	Number of threads used for parallel processin