MP3DG-PCC, Open Source Software Framework for Implementation and Evaluation of Point Cloud Compression

Rufael Mekuria
CWI
SciencePark 123
1098 XG Amsterdam
r.n.mekuria@cwi.nl

Pablo Cesar
CWI
SciencePark 123
1098 XG Amsterdam
p.s.cesar@cwi.nl

ABSTRACT
We present MP3DG-PCC, an open source framework for design, implementation and evaluation of point cloud compression algorithms. The framework includes objective quality metrics, lossy and lossless anchor codecs, and a test bench for consistent comparative evaluation. The framework and proposed methodology is in use for the development of an international point cloud compression standard in MPEG. In addition, the library is integrated with the popular point cloud library, making a large number of point cloud processing available and aligning the work with the broader open source community.

Categories and Subject Descriptors
H.5.1 [**Multimedia Information Systems**]: Compression, Evaluation, 3D Virtual Reality, Point Cloud Compression

Keywords: Compression, Benchmarking, Software

1. INTRODUCTION
More 3D Point Cloud data are becoming available from 3D scanning and reconstruction. These data are useful in a large range of applications such as virtual and augmented reality, 3D imaging, and 3D simultaneous localization and mapping (SLAM). Typical 3D point cloud scans contain thousands up to millions of points and are sometimes acquired at fast rates (10-30 fps using current Time-of-Flight and LiDaR scanners). Therefore, to support efficient storage and exchange of such data, efficient compression becomes critical. This is of interest of both the research community, the open source community and standardization organizations such as MPEG and JPEG that develop standards for adaption in large scale consumer electronics. Both consortia are in the process of developing standards for static and dynamic point cloud compression, to address the large amounts of data becoming available from 3D scanners.

MM '16, October 15 - 19, 2016, Amsterdam, Netherlands

ACM 978-1-4503-3603-1/16/10…$15.00
DOI: http://dx.doi.org/10.1145/2964284.2973806

Compression of point clouds has received quite some attention from the 3D Graphics research community. Much emphasis has been on compression of the geometry positions representing a point sampled surface and/or respective attributes such as colors. This research establishes some fundamental ideas for point cloud compression. However, it fails to address many of the current requirements for point cloud compression for 3D scanning/imaging in emerging VR applications. Some of these insufficiently addressed requirements are as follows.

Lossy/lossless coding of attribute data (colors, properties) is important, but it has received little attention. Often methods optimized for geometry are applied, but this is not optimal for coding attributes such as colors and/or surface properties.

Time-varying point cloud compression is important, as scans are often acquired at a fast rate. In this case exploiting inter frame redundancy can be exploited, but this is still an unexplored topic.

Real-time compression and *parallelization* are important in many emerging consumer and professionally oriented applications (virtual reality, 3D tele-immersive communication, simultaneous mapping and localization (SLAM) for robotics etc.). Most work in the literature does not support real-time and parallel coding well.

We expect that it will take multiple cycles to develop and fine tune compression techniques that address each of these requirements sufficiently well. To align such cycles, common datasets, assessment criteria and methodology are useful. For point cloud compression such a common framework is currently lacking. In this work we present MP3DG-PCC an open source framework based on common methodology, assessment criteria and datasets enabling future work on point cloud compression with a common assessment methodology. In addition, it contains an anchor codec that already addresses some of the requirements and can be developed towards a sophisticated and standardized solution (i.e. real-time inter-predictive and lossy attributes coding).

These contributions will benefit researchers that are developing codecs in open source communities and/or standardization organizations, or in industry. In addition, it will also be useful for researchers working on complex multimedia systems that want to readily integrate the anchor codecs in larger AR/VR systems.

The package is available in the MPEG software repository and the trunk version is aligned with the open source version available in an open source GitHub repository (https://github.com/RufaelDev/pcc-mp3dg). The build process uses cross platform make (CMake). The software can both be used for evaluating compression of point cloud datasets and for further development and evaluation of the codec anchor.

This paper is structured as follows. Section 2 presents an overview of Point Cloud Compression. Section 3 presents the MP3DG PCC software in detail. Section 4 concludes the work and highlights its usefulness.

2. POINT CLOUD COMPRESSION

Point cloud compression deals with compression of points and their attributes located in the 3D space. Many of the compression methods proposed in the literature utilize the octree structure to organize the unstructured data. The octree provides a sparse and easily serializable representation of the point cloud and enables many levels of detail. With emerging applications areas such as 3D Tele-immersive communications and 3D printing additional requirements need to be addressed. For example, in [1] we describe the design implementation and evaluation of a point cloud codec for 3D tele-immersive real-time conferencing addressing lossy color coding, inter predictive coding and real-time encoding.

To support consistent evaluation for coding with these additional requirements, we introduce MP3DG-PCC, a framework for quality assessment and benchmarking of PCC. This framework will enable more consistent quality evaluations and codec development. Before we continue we detail some of the requirements for point cloud compression that need to be taken into account in its evaluation.

Scalable coding: Progressive, or coding with many levels of detail has been considered in the literature [2]. For emerging applications scalable with few levels of detail but low complexity for encoding and decoding are desirable.

Real-time coding and parallelization is needed in many applications, while a real-time PCC codec was proposed, there is still much work to be done enabling good parallelization

Lossy geometry coding is desirable for point clouds with millions of points that need to be stored with a small number of bytes.

Lossy Color Attributes coding the coding of attributes associated to geometry points needs more attention. More efficient techniques for lossy attributes coding such as in [3] are needed.

Lossless Coding is important in many applications such as computer aided design, medical and so on

Multi-view coding sometimes directional information is important and needs to be incorporated in the codec. Depending on the angle of inspection point properties (colors), might be different.

Material properties and generic attribute coding 3D reconstruction, sometimes provides detected material properties and/or generic attributes. These may be beneficial for rendering and or further processing and should also be stored efficiently

Inter-predictive coding clouds are often captured at a fast rate, inter-predictive coding still needs attention.

The MP3DG-PCC open source framework enables assessment and codec development based on these requirements. The specific **contributions** of this framework are the following:

- Quality metrics for point cloud geometry and color attributes analogous to Peak Signal to Noise Ratio (PSNR)
- A test bench including file loaders and evaluation routines with easy configuration enabling automatic benchmarking
- Point Cloud pre-processing (outlier removal, octree bounding box normalization) ensuring consistent evaluation
- Commonly available datasets based on data re-constructed with Microsoft Kinect 1, Kinect 2 and other devices.
- A time varying PCC anchor codec. Support for benchmarking time varying point cloud compression.
- A real-time, parallelized, anchor PCC codec. Support for benchmarking real-time performance and parallelization of point cloud codecs.
- A lossy attribute coding anchor codec. Support for benchmarking lossy color attribute coding

While the presented software framework can be used in different contexts, its primary aim is to support the development of a point cloud compression in the Moving Picture Experts Group (MPEG). The Moving Picture experts group has completed the exploration on point cloud compression defining requirements [4], use cases [5] test datasets [6] and evaluation criteria [7]. Based on these a draft call for proposals was released in June 2016 [7]. The documents are available to the public on the MPEG webpage of the June 2016 meeting [8] and provide a good overview of the scope and methodology of this activity.

3. BENCHMARKING PLATFORM

3.1 MP3DG-PCC overview & config

The benchmarking platform software contains three major components that can be located in separate folders in the C++ software source code. It contains a quality metric in the folder quality for performing a PSNR like quality assessment of the point cloud (both attribute and geometry, see [1] for more details on the metric). The anchor codec contains time varying (inter-predictive) and lossy color coding techniques. It can be found in cloud_codec_v2. The benchmarking platform (including preprocessing and file loaders) can be found in compression_eval. Compression_eval compiles to an executable to run the test bench. The test bench parameters can be configured through parameter_config.txt. When the test bench is ran, an input folder containing all point clouds in the .ply format can be given as an input argument. The flow of the test bench is illustrated in Figure 1. All clouds are loaded into memory, optionally fused (if clouds from different angles have been recorded, i.e. view selective clouds). In the next step bounding box, normalization and outlier filtering is performed to enable normalized inputs to the codec (normalize all points to the 0-1 scale, such that rms distances can be consistently evaluated). Next, all codec configurations are traversed and outputs are written to a .csv file. The quality scores include bit rates (for geometry, attribute), encoding/decoding time, and quality (based on a PSNR like metric adapted for point clouds see [1]).

The flow of compression_eval is illustrated in Figure 1 and Table 1 outlines the different components in the software. The source code is embedded in the version 1.7.2 of the point cloud library, and will soon be ported to the latest version.

The test bench can be configured using the parameter_config.txt file. This file contains all options for running the test bench using anchor codecs. The option octree bit-settings and color bit-settings can be used to control the number of octree levels. The color coding types can be set to 0,1,2 which correspond to DPCM, JPEG based (grid), and jpeg based lines. Table 1 Components in the software these modes map the depth first scan of the octree to an image grid for further compression. The enh_bit_settings can be used to set the bits in the enhancement layer (beyond the octree). We found that the octree compression only works well upto a certain level of detail, after which another mode of compression is preferable. This mode is in the enhancement layer. The option keep_centroid can be used to also code the centroid in the leaf node, enabling smoother low res representations. The bounding box expand factor can be set, which slightly enlarges the octree bounding box to enable inter predictions (consistent bounding box frames). The output .csv file containing all results can be set. Then, optionally decoded .ply files can be written. With do_delta_frame coding inter-predictive coding in the anchor can be enabled. The pframe_quality_log sets the output csv for inter-predictively encoded frames. For inter predictive coding the macroblock size for inter prediction in the anchor can be set in macroblock size. The option test bbalign can be used to test the effectiveness of the boundingbox alignment only in a single test run. The code_color_offset can be used to code color offsets in the anchor codec in inter-predicted frames. The options related to the radius outlier filter enable pre-filtering of the point clouds to avoid outliers that disturb the octree based coding mechanism. The parameter_config.txt can be located at the location of the binary or one level beneath.

Table 1 Component in framework for PCC

Quality	Quality metric
Compression_eval	Test bench
Cloud_codec_v2	Anchor PCC Codec

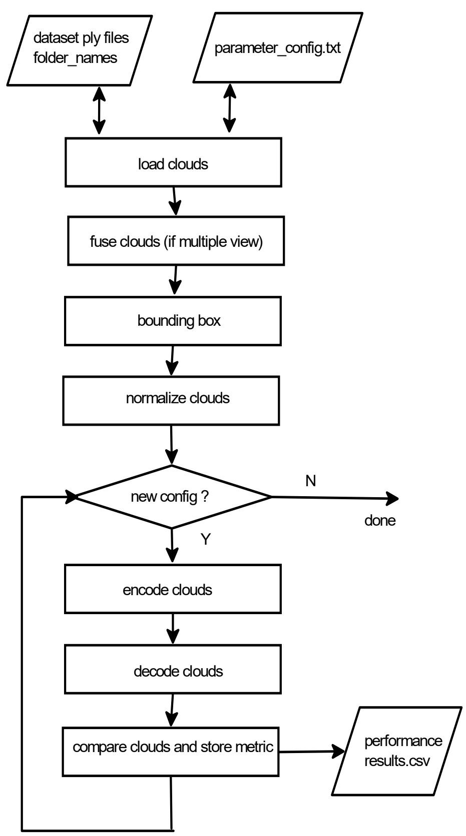

Figure 1 Compression eval testbench

Figure 2 Some examples of .ply testdata available for benchmarking

Table 2 settings for codec (parameter_config.txt)

#configuration file for testing the color encoding
octree_bit_settings=9
color_bit_settings=8
color_coding_types= 0 # color coding types 0=dpcm 1=grid 2=lines
enh_bit_settings = 0
keep_centroid=0 # option for keeping centroid instead of voxel centers only
bb_expand_factor=0.20# expansion factor used to keep bounding box constant over time
output_csv_file=intra_frame_quality_11_DPCM.csv
write_output_ply=1
do_delta_frame_coding=1
icp_on_original=0
pframe_quality_log=predictive_quality_11_DPCM.csv
macroblocksize=16
testbbalign=0
code_color_off=0
radius_outlier_filter=0
radius_size=0.1
jpeg_value=75
omp_cores=0
scalable=0

3.2 Datasets

Test datasets from 4 CERTH/ITI [1] based on [10] and Queen Mary University captured with Kinect 1 and Kinect 2 are available online, and form cultural heritage (reconstruction from multiple images) [2]. Screenshots of the data are shown in Figure 2. In addition in [7] additional datasets have been proposed such as from Microsoft Research.

3.3 Anchor Codec

The open source framework includes an anchor codec. It is based on octree, inter prediction and lossy color coding by mapping to an image grid. This codec is described in detail in [1], in addition the codec described in using octree and DPCM for color coding can be used as an anchor [11]. Improvements in each of the blocks can be tested using the test bench can be easily evaluation. This

[1] http://vcl.iti.gr/reconstruction/

[2] http://wg11.sc29.org/content//MPEG-04/Part16 Animation_Framework_eXtension_(AFX)/3DGraphics/pointCloud

enables collaborative improvements of the test model. For more information on the anchor codec we refer to [1].

3.4 Dependencies and installation guidelines

The package shares dependencies with the point cloud library [12] and can be build in similar fashion. Additionally, the anchor coding uses the JPEG codec using the libJPEGTurbo library which should also be installed. The framework does depend on boost, Flann, and Eigen, but not on QT, VTK etc. The package can be build on platforms such as Linux, windows and Mac. The build system uses CMake build scripts and can be build using the CMake configuration and generation GUI.

4. Discussion and Conclusion

This work presented a full benchmarking platform for point cloud compression algorithms including an improved anchor codec and PCC quality metrics. This framework addresses the need in the industry for improved compression and consistent evaluation of point cloud compression. The code is based on the popular point cloud library which is used as a defacto standard for point cloud processing in academia and industry and written in standard C++. We further discuss two use cases of this work on point cloud compression. The first is in the development of international and de-facto standards for point cloud compression. The second is the direct application in the development of AR/VR and other advanced 3D Multimedia Systems.

International Standardization of point cloud compression is still in an early stage. The Moving picture experts group (MPEG) consortium, has developed many industry standards for compression of visual media in the past. The rise in 3D point cloud data made it of increasing interest to MPEG. The development of an MPEG PCC standard will enable more requirements to be addressed and a wider range of experts on media coding to contribute to the codec, compared to previous work on PCC. In addition, it can enable fast industry adoption. The MPEG consortium will further develop this standard in the coming years. The MPEG group on 3D Graphics initiated this work. The currently presented test bench serves as a starting point for the standardization process. The trunk version of the MPEG source is aligned with the open source version available in a GitHub repository (https://github.com/RufaelDev/pcc-mp3dg).

Close to MPEG, the Joint Picture Experts Group (JPEG), known for developing compression for still image coding, included the possibility of point cloud compression in its emerging JPEG PLENO standard[3]. This software might also be of use there, but this standard addresses slightly different use cases targeting compression of data coming from plenoptic cameras. Still this work might also benefit work in JPEG PLENO.

Secondly, this framework can benefit the multimedia research community at large such as in the development of point cloud streaming protocols and advanced systems. An example of a point cloud streaming in mixed reality was developed in earlier work as can be shown in Figure 3. In this Figure, a captured 3D point cloud is transmitted to a remote site, and rendered in a 3D virtual world, enabling a mixed reality tele-immersive experience. In this framework, users could navigate remotely as a realistic representation of themselves. Further, 3D point clouds may be used as a more generic 3D Video format, in this case HTTP based streaming of this kind of data will becoming a hot research topic. The anchor codec developed in this framework can be an excellent starting point to start an investigation of adaptive streaming of point clouds over HTTP or other protocols.

Figure 3 Point Cloud rendering in a virtual environment

[3] https://jpeg.org/items/20150320_pleno_summary.html

5. REFERENCES

[1] R.N. Mekuria K. Blom, P. Cesar, "Design, Implementation and Evaluation of a Point Cloud Codec for 3D Tele-immersive Video," *IEEE Transactions on Circuits and Systems for Video Technology*, september 2016.

[2] Y. Huang, J. Peng, C. Kuo, C. Gopi , "A generic scheme for progressive point cloud coding," *IEEE Transactions on Visualization and Computer Graphics*, vol. 14, no. 2, pp. 440-453, 2008.

[3] C. Zhang , D. Florencio, C.Loop, "Point cloud attribute compression with graph transform" in *IEEE ICIP*, Paris, France, 2014, pp. 2066-2070.

[4] MPEG "N16330 Requirements for Point Cloud Compression" ISO/IEC JTC1 SC29 WG11 Geneva, June 2016

[5] MPEG "N16331 Use Cases for Point Cloud Compression" ISO/IEC JTC1 SC29 WG11 Geneva, June 2016

[6] MPEG "N16332 Evaluation Criterion for Point Cloud Compression" ISO/IEC JTC1 SC29 WG11 Geneva, June 2016

[7] MPEG "N16333 Draft Dataset for Point Cloud Compression", ISO/IEC JTC1 SC29 WG11 Geneva, June 2016

[8] MPEG "N16334 Draft Call for Proposals for Point Cloud ISO/IEC JTC1 SC29 WG11 Compression", Geneva, June 2016

[9] "MPEG 115" http://mpeg.chiariglione.org/meetings/115 accessed 1/8/2016

[10] D. Alexiadis, D. Zarpalas, P. Daras , "Real-Time, full 3D Reconstruction of moving foreground objects from multiple consumer depth cameras," *IEEE Transactions on Multimedia*, vol. 15, no. 2, pp. 339-358, Februari 2013.

[11] J. Kammerl. , N. Blodow, R.B. Rusu, S. Gedikli, and E. Steinbach M Beetz, "Real-time compression of point cloud streams," in *Robotics and Automation (ICRA), 2012 IEEE International Conference on , pp.778,785, 14-18 May*, 2012.

[12] Rusu, R.B. "3D is here: Point Cloud Library," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, Shanghai, 2011.