The Journal of Open Source Software

pointcloudset: A python package for working with multiple pointclouds recorded over time

Thomas Goelles*[1], Birgit Schlager[1,2], Stefan Muckenhuber[1,3], Sarah Haas[1], and Tobias Hammer[1]

1 Virtual Vehicle Research GmbH, Inffeldgasse 21A, 8010 Graz, Austria **2** Graz University of Technology, Rechbauerstrasse 12, 8010 Graz, Austria **3** University of Graz, Heinrichstrasse 36, 8010 Graz, Austria

DOI: DOIunavailable

Software

- Review
- Repository
- Archive

Editor: Pending Editor

Reviewers:

- @Pending Reviewers

Submitted: N/A
Published: N/A

Summary

Point clouds are a very common format for representing three dimensional data. Point clouds can be acquired by different sensor types and methods, such as lidar (light detection and ranging), radar (radio detection and ranging), RGB-D (red, green, blue, depth) cameras, photogrammetry, etc. In many cases multiple point clouds are recorded over time. E.g., automotive lidars record point clouds with very high acquisition frequencies (typically around 10-20Hz) resulting in millions of points per second. Analyzing such a large collection of point clouds is a big challenge due to the huge amount of measurement data. The python package `pointcloudset` provides a way to handle, analyse and visualize large datasets consisting of multiple point clouds recorded over time. `pointcloudset` features lazy evaluation and parallel processing and is designed to enable development of new point cloud algorithms and their application on big datasets.

`pointcloudset` builds on several well established python libraries and packages for data processing and visualization, such as dask Rocklin (2015), pyntcloud (Pyntcloud Development Team, 2021), open3D (Zhou et al., 2018), plotly (Inc., 2015) and pandas McKinney (2010).

Statement of need

Considering recently emerging, promising lidar technologies, such as micro-electro-mechanical systems, optical phased array, vertical-cavity surface-emitting laser, single photon avalanche diode etc., combined with large efforts invested in particular by the automotive industry Thakur (2016) to further develop low-cost lidar systems, lidar sensors have the potential to enable a new cost-efficient way to perceive and measure the environment. This will not only have a strong impact on automotive applications but bears also large potential for other research fields and application domains, such as robotics, geophysics, etc. Already today, state-of-the-art lidar sensors designed for automotive applications, such as the Ouster OS-1 [https://ouster.com] or the Velodyne Ultra Puck [https://velodynelidar.com], offer many advantages: they are small in size, light in weight, robust, have a low eye safety class, and support high scanning speed. The expected substantial decrease in costs and increase in performance in the upcoming years will open up many new application areas for lidar systems.

Apart from the progress in the lidar sector, technological improvements, as well as size and cost-reduction can also be observed for other 3D sensing technologies, such as radar and RGB-D cameras. This will additionally open up new possibilities and application areas for 3D sensing methods and increases the importance of python packages that are able to process large amounts of point cloud data.

*corresponding author Thomas.Goelles@v2c2.at

Other python packages for point clouds, such as open3D and pyntcloud, focus on processing single point clouds. ROS (robot operating system) (Stanford Artificial Intelligence Laboratory et al., 2018) provides a way to store, access, and visualize multiple point clouds stored as `rosbags`. However, these `rosbags` are meant to be accessed only in a serial fashion, which is not ideal for post processing and not well suited for extracting subsets of the point cloud dataset.

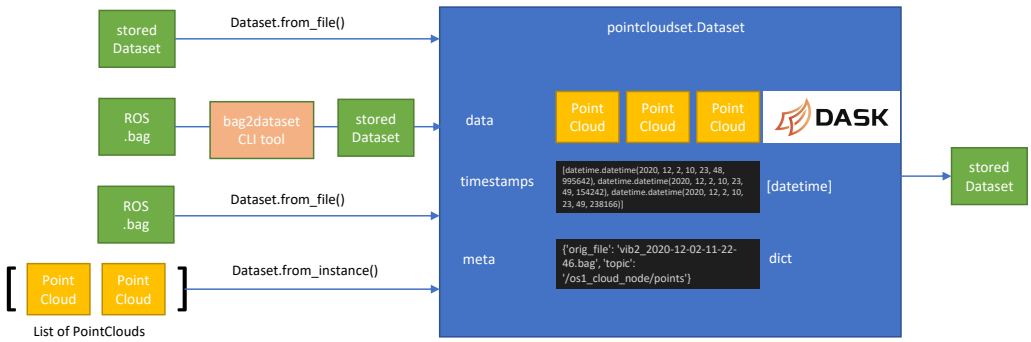

Figure 1: Dataset object with main properties and ways to read and write data.

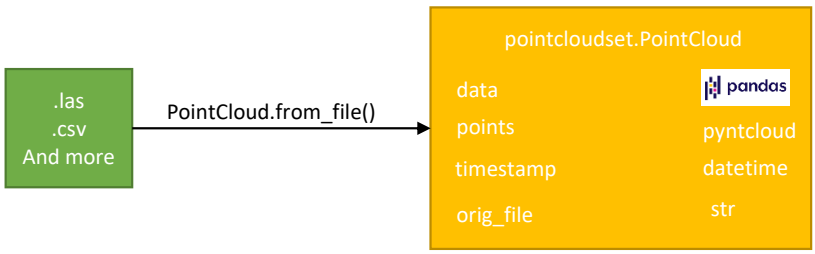

Figure 2: PointCloud set with main properties and ways to read and write data.

Figure 1 illustrates the structure of the `Dataset` class including import and export possibilities. A Dataset consist of many PointCloud objects which can be accesses like list elements in Python. Alternatively a `PointCloud` object can also be created directly from files, as illustrated in Figure 2.

Acknowledgements

The publication was written at Virtual Vehicle Research GmbH in Graz, Austria. The authors would like to acknowledge the financial support within the COMET K2 Competence Centers for Excellent Technologies from the Austrian Federal Ministry for Climate Action (BMK), the Austrian Federal Ministry for Digital and Economic Affairs (BMDW), the Province of Styria (Dept. 12) and the Styrian Business Promotion Agency (SFG). The Austrian Research Promotion Agency (FFG) has been authorised for the programme management.

References

Dask Development Team. (2016). *Dask: Library for dynamic task scheduling*. https://dask.org

Hecht, J. (2018). Lidar for Self-Driving Cars. *Optics & Photonics News*, *29*(1), 26. https://doi.org/10.1364/OPN.29.1.000026

Inc., P. T. (2015). *Collaborative data science*. Plotly Technologies Inc. https://plot.ly

McKinney, Wes. (2010). Data Structures for Statistical Computing in Python. In Stéfan van der Walt & Jarrod Millman (Eds.), *Proceedings of the 9th Python in Science Conference* (pp. 56–61). https://doi.org/10.25080/Majora-92bf1922-00a

Pyntcloud Development Team. (2021). *Pyntcloud*. https://pyntcloud.readthedocs.io/en/latest/introduction.html

Rocklin, M. (2015). Dask: Parallel computation with blocked algorithms and task scheduling. In K. Huff & J. Bergstra (Eds.), *Proceedings of the 14th python in science conference* (pp. 130–136).

Stanford Artificial Intelligence Laboratory et al. (2018). *Robotic operating system* (ROS Melodic Morenia) [Computer software]. https://www.ros.org

team, T. pandas development. (2020). *Pandas-dev/pandas: pandas* (latest) [Computer software]. Zenodo. https://doi.org/10.5281/zenodo.3509134

Thakur, R. (2016). Scanning LIDAR in advanced driver assistance systems and beyond: Building a road map for next-generation LIDAR technology. *IEEE Consumer Electronics Magazine*, *5*(3), 48–54. https://doi.org/10.1109/MCE.2016.2556878

Warren, M. E. (2019). Automotive LIDAR technology. *2019 Symposium on VLSI Circuits*, C254–C255. https://doi.org/10.23919/VLSIC.2019.8777993

Zhou, Q.-Y., Park, J., & Koltun, V. (2018). Open3D: A modern library for 3D data processing. *arXiv:1801.09847*.