Buildings and Terrain of Urban Area Point Cloud Segmentation based on PCL

To cite this article: Ying Liu and Ruofei Zhong 2014 IOP Conf. Ser.: Earth Environ. Sci. 17 012238

View the article online for updates and enhancements.
Buildings and Terrain of Urban Area Point Cloud Segmentation based on PCL

Ying LIU, Ruofei ZHONG

Key Laboratory of Resource, Environment and GIS in Beijing, Key Laboratory of 3D Information Acquisition and Application, Ministry of Education, College of Resources Environment and Tourism, Capital Normal University, Beijing, China.

E-mail:lyy_cnu@whu.edu.cn

Abstract: One current problem with laser radar point data classification is building and urban terrain segmentation, this paper proposes a point cloud segmentation method base on PCL libraries. PCL is a large cross-platform open source C++ programming library, which implements a large number of point cloud related efficient data structures and generic algorithms involving point cloud retrieval, filtering, segmentation, registration, feature extraction and curved surface reconstruction, visualization, etc. Due to laser radar point cloud characteristics with large amount of data, unsymmetrical distribution, this paper proposes using the data structure of kd-tree to organize data; then using Voxel Grid filter for point cloud resampling, namely to reduce the amount of point cloud data, and at the same time keep the point cloud shape characteristic; use PCL Segmentation Module, we use a Euclidean Cluster Extraction class with Europe clustering for buildings and ground three-dimensional point cloud segmentation. The experimental results show that this method avoids the multiple copy system existing data needs, saves the program storage space through the call of PCL library method and class, shortens the program compiled time and improves the running speed of the program.

1. Introduction

Using laser scanning technology to process spatial data has advantages of accessing rapidly, high precision and ease to deal with. 3D point cloud spatial information mining by using this technology has massive characteristics which require feature selection of point cloud data. Of all of the steps, point cloud segmentation could be vital, is an essential step of 3D virtual reconstruction work, and this step divides space, geometry and texture by analyzing the point cloud characteristics, and the results of in the same division point cloud with similar characteristics. Since point cloud data contain noise and discontinuities, current segmentation methods are mainly based on region. Some scholars selected one of eight local performance types from the Gaussian curvature as a symbol, used quadratic surface fitting method for regional growth after the first segmentation to get the final result.

This article presents a method of point cloud segmentation by using PCL, which is also known as the point cloud library. In this case, the kd-tree data structure is introduced into algorithm to organize the data to reduce the number of points in the re-sampling process, after that, with the segmentation module in PCL, 3D point cloud will be spilt from buildings and tree from ground which by using the Euclid cluster analysis.
2. Point cloud library: PCL

PCL is a large scale cross-platform open source C++ programming library, which implements a large number of point cloud universal algorithms and efficient data structures. PCL involves cloud retrieval, filtering, segmentation, registration, feature extraction, surface reconstruction, and visualization. Currently, in various ground point extracts algorithms, the difference between algorithm results and actual results is way too high compared to the actual situation. However PCL could solve problems. For 3D point cloud processing, PCL is a modular C++ library, as for an algorithmic perspective. PCL is a variety of 3D point cloud data processing algorithms set. Each set of algorithms is divided into basic classes. The processing steps in PCL can be listed as followed:

- Create objects (filtering, feature estimation, segmentation, etc.).
- Input point cloud data and process the data by using setInputCloud module.
- Set relevant parameters in the algorithm.
- Calculate and output.

PCL modules use CMake tools to develop. The specific process is as follows: First, create a folder named ‘source’ to store the source files and create a file named ‘segmentation.cpp’ to record program code, then new a file named ‘CMakeLists.txt’ in the folder to record the information of project, the file contains the PCL head file, so the compiler knows the methods used by the program as well as the link library.

Configuration requirements:
Visual Studio 2008 and CMake develop tools, CMake version greater than 2.8.3 with the main consideration in the PCL used version macro definition, low version is not compatible.

3. Point cloud segmentation

The ideological of point cloud segmentation is "divide and rule". Segmentation is an important basic operation in point cloud processing. The infrastructure of point cloud segmentation has been established in PCL. The clustering segmentation algorithm (Euclidean Cluster Extraction class) with better robustness is used in this paper. And data pre-processing like filtering is required before the point cloud segmentation.

3.1 Data preprocessing

Before the point cloud segmentation, it's necessary to do the filtering processing, down-sampling with the method of voxelization grid, both to reduce the number of points, while maintaining the point cloud shape characteristics. In this paper, using the Voxel Grid filter library in PCL to re-sample point cloud, creating a three-dimensional voxel grid with the input point cloud data, and the point in each voxel(i.e., a three-dimensional cube) is replaced by the centre of gravity of all the points in the voxel. In order to maintain the original shape of point cloud, the size of the voxel should be set as small as possible.

3.2 Kd-tree

Point cloud data acquired by radar, laser scanning, three-dimensional camera and other three-dimensional measurement device, have the disadvantages of large amount of data and uneven distribution. Most of data segmentation studies are for rules data, but the segmentation of the scattered point cloud data is much more difficult, due to the geometric topology relationship between the points. Kd-tree is a kind of spatial segmentation strategy, and a binary search tree with the other constraints. In this paper we proposed to use it. In the method, the space is split into two parts, each half is split through the same manner and so on. As a result, the space will be divided into two parts of left and right or of up and down. As shown in Figure 1 and Figure 2, in kd-tree we separate all the child nodes in the specified latitude at each level.
3.3 Point cloud segmentation

Segmentation is divided into two types of algorithm, point cloud clustering segmentation and consistency segmentation based on random sampling. The European cluster is taken in this paper, the basic principle: input data points, its number is m, define affinity clustering of certain characteristic between points in the m-dimensional space, the set of data points divided into classes, the size of all the classes is n, and then merger the minimum distance of two categories into a class, and recalculate the distance between classes, iteration until the distance between any two categories is greater than the specified threshold value, or the number of classes is less than a specified value, then completed segmentation.

First, create a kd-tree object as the search method used in the extraction of point cloud, and then create a point cloud index vector ‘cluster_indices’ used to store the actual point cloud index information, each detected point cloud clustering be saved.

```cpp
pcl::search::KdTree<pcl::PointXYZ>::Ptr tree(new pcl::search::KdTree<pcl::PointXYZ>);
tree->setInputCloud(cloud_filtered);
```

Here ‘cloud_filtered’ is the point data after filtering. Then create a segmentation object, set the radius of the search, the minimum and maximum number of points of the clustering, as well as point cloud search mechanism. Last, iteratively visit all the clustering index of point clouds, until segmentation all clustering.

```cpp
pcl::EuclideanClusterExtraction<pcl::PointXYZ> ec;
```

4. Result and analysis

The data this experiment used is obtained by Capital Normal University moving vehicle measurement system, shown in Figure 3, the scanning range is Capital Normal University Library and its vicinity, contain three million scattered points. When filtering the point cloud and re-sampling, considering the distance between the point and point in about 3cm, set the filter parameter that the volume of the voxel selects 0.3m.
When dividing the point cloud, first use the plane segmentation mode to do the processing of the point cloud, extract all points on a plane, shown in Figure 4. Using the method in this paper, the point cloud is well segmented, in Figure 5 is the segmentation results of the buildings, Figure 6 is the tree segmentation results.
We can see that the experimental results are very satisfactory segmentation. During the experiment, I want to make some opinions here:

- Using CMake compiler, is a good way of programming, code and header files and link libraries written separately and can be very easy to find errors in an error.
- The program runs not very slow, relative to other types of segmentation methods have been greatly improved, which is due to the large amount of data, which is a basic problem faced by the researchers on point cloud.
- In the beginning of the experimental, it is difficult to select the right parameters, the clustering segmentation iteration of the loop, the program will appear bugs. After several tests, come to the results of this paper. This result may not be ideal, future research will be to find better parameters.

5. Conclusion

Experimental results show that this article take the ideological, first point cloud data filter processing, then create a kd-tree object as the search method used to the extraction of point cloud, and at last create an index vector use to storage index information of point cloud, is correct and has obviously segmentation outcome. This method avoids the multiple copy system existing data needs, saves the program storage space, through the call of PCL library method and class, shortens the program compiled time, improves the running speed of the program, and finally gets ideal segmentation results. For improving the later work from the following two aspects: 1) to find a simple way for the pre-processing of the data filtering, currently use of the centre of gravity of the voxel approximation method has slow program running speed, affect procedures efficiency; 2) the current distribution of closed approximate density point cloud model has good segmentation results, the next step we need to eliminate the point cloud density point cloud model closed the algorithm of image.

Acknowledgments

The work of this paper is supported by the open fund from State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University (NO:11P02) and the National Science and technology support program (NO: 2012BAH34B01).
References
[1] Radu Bogdan Rusu and Steve Cousins 2011 3D is here: Point Cloud Library (PCL) IEEE International Conference on Robotics and Automation (ICRA) Shanghai China.
[2] Bo Shi, Xiushan lu and Yunfang chen 2008 J. Planar segmentation of building’s unorganized point cloud based on Kd-tree Science of surveying and mapping Vol.33 No.1.
[3] Huan Fu, Li Liang, Fei Wang and Jizhong Zhao 2012 J. A Point Cloud Segmentation Algorithm Using Local Convexity and Octree Journal of Xi’an JiaoTong University Vol.46 No.10.
[4] Moosmann F, Pink O and Stiller C 2009 Segmentation of 3D lidar data in non-flat urban environments using a local convexity criterion Proceedings of IEEE Intelligent Vehicles Symposium Piscataway NJ USA:IEEE pp 215-220.
[5] Miao Wang and Yihsing Tseng 2010 J. Automatic segmentation of Lidar data into coplanar point clusters using an octree-based split-and-merge algorithm Photogramme Thrice Engineering and Remote Sensing 76(4): pp 407-420.
[6] Douillard B, Underwood J and Kuntz N.et al 2011 On the segmentation of 3D LIDAR point clouds Proceedings of IEEE International Conference on Robotics and Automation Piscataway NJ USA: IEEE pp 2798-2805.
[7] Mario R and Markus V 2009 Point cloud segmentation based on radial reflection Proceedings of 13th International Conference on Computer Analysis of Images and Patterns Heidelberg Germany: Springer Verlag pp 955-962.
[8] Hongyan Sun, Xiaopeng Sun and Hua Li 2006 J. 3d Point Cloud Model Segmentation Based on K-means Cluster Analysis Computer Engineering and Applications pp 42-45.
[9] Jin Liu, Zhongke Wu and Mingquan Zhou 2011 J. Review of Point Cloud Segmentation and Application Computer Science Vol.38 NO.4.