Robotics 2020 :Project Proposal

Project name: Efficient Point-Cloud Semantic Segmentation

Delin Feng ,Longtian Qiu, Qianjing Shi
fengdelin@cau.edu.cn, qiult@shanghaitech.edu.cn, shiqj@shanghaitech.edu.cn

Advisor: Sören Schwertfeger, Yijun Yuan

ShanghaiTech University

1 Abstract:

LiDAR point-cloud segmentation is an important problem for many applications. For large-scale point cloud segmentation, there are four paradigms for semantic segmentation: projection-based, discretization(voxel) based, point-based, and hybrid methods. Voxel-based representation is regular and has good memory locality. However, it requires very high resolution in order not to lose information. Point-based 3D modeling methods are memory effificient. but it lacks the local context modeling capability. Projection-based methods discover that the feature distribution of LiDAR images changes drastically at different image locations. This project will propose an algorithm merging the advantages of different methods to solve the problem of missing detailed structure and pose after projection or down-sampling to improve the efficiency and accuracy of the algorithm. We will conduct experiments on some open source data sets and compare them with state of the art methods to evaluate the performance of the algorithm. We will also create our own(ShanghaiTech campus) data set for experiments to verify the efficiency and accuracy of the algorithm.

2 Introduction:

Efficient semantic segmentation of large-scale 3D point clouds is a fundamental and essential capability for real-time intelligent systems, such as autonomous driving and augmented reality, also it can provide more semantic information. Recently, deep learning has been shown to be very effective for point cloud perception tasks. Conventionally, researchers rasterize the point cloud into voxel grids and process them using 3D volumetric convolutions. With low resolutions, there will be information loss during voxelization: multiple points will be merged together if they lie in the same grid. Therefore, a high-resolution representation is needed to preserve the fine details in the input data. However, the computational cost and memory requirement both increase cubically with voxel resolution. Recently, another stream of models attempt to directly process the input point clouds. These point-based models require much lower GPU memory than voxel-based models thanks to the sparse representation. However, they neglect the fact that the random memory access is also very inefficient. Projection-based methods can perform semantic segmentation more quickly, but are limited by the fuzzification of CNN output and discretization errors.In order to achieve semantic segmentation of point clouds on autonomous mobile robots, we will process point clouds based on SqueezeNet to obtain 2d images, and implement further feature fusion based on the point-voxel framework to improve the current lack of detailed structure and some of the reprojection problem. Since there is less open source data for point cloud semantic segmentation, we plan to collect data from the ShanghaiTech campus for experiments, but because the large-scale point cloud ground truth calibration is very troublesome, this is a difficult challenge for us
Our key contributions are: 1)How to improve efficiency of projection-based method, it will not worse then current method. 2)Voxel method has more requirements on memory, so it's important to improve details under the guarantee of current quality.

3 State of the Art:

Find literature and open-source-ROS packages relevant to your project. Per team member:

- Fengdelin's proposal:
  Point-Voxel CNN for Efficient 3D Deep Learning.2019
  Guo Y , Wang H , Hu Q , et al. Deep Learning for 3D Point Clouds: A Survey[J]. 2019.

Wang Y , Shi T , Yun P , et al. PointSeg: Real-Time Semantic Segmentation Based on 3D LiDAR Point Cloud[J]. 2018.
These three papers are all about the semantic segmentation of 3D point clouds, one of which is a survey of existing methods, and the algorithm improvement of the other two is my focus.

The pvcnn paper first talked about the disadvantages of voxel-based and point-based methods.Voxel-based representation is regular and has good memory locality. However, it requires very high resolution in order not to lose information. Point-based 3D modeling methods are memory efficient. it lacks the local context modeling capability, this will lead to the irregular memory access pattern and introduce the dynamic kernel computation overhead, which becomes the efficiency bottlenecks.
PV-Conv contains two branches, one is point-based and the other is Voxel-based. The first steps of voxel-based branch operations are similar to most voxel-based methods, including normalizing coordinates, voxelization, apply a stack of 3D volu- metric convolutions to aggregate the features. If Only here, there will still be problems of resolution and loss of detial information, so the article uses point-based branch to introduce geometric detail information very cleverly to supplement the deficiencies of voxel-based; One of the highlights of this article is deoxelization. Its function is to convert the features of a voxel into the features of each point in the voxel. The article considers that the feature of each voxel corresponds to the center of each voxel, and uses the trilinear interpolation method to ensure that the features of each point are different. It seems a bit Like deconvolution in 2d deep learning.
The point-based method no longer uses the points of the neighborhood, but uses MLP to directly calculate the feature of each point.
With both individual point features and aggregated neighborhood information, it can efficiently fuse two branches with an addition as they are providing complementary information.

we maybe will use ROS Velodyne_pointcloud package to collect our own 3D point cloud dataset.
This package provides point cloud conversions for Velodyne 3D LIDARs and provides 4 ROS nodes and nodelets: CloudNodelet, CloudNodelet, cloud_node, transform_node.
Theses nodes reads raw data from the velodyne_packets ROS topic, converts to sensor_msgs/PointCloud2 format, and republishes to velodyne_points ROS topic in the original frame of reference (typically /velodyne). In addition to the XYZ points, this cloud includes fields for ”intensity” and ”ring”. You can transform raw Velodyne packets into sensor_msgs/PointCloud2 messages into the /odom frame.
We plan to use velodyne to obtain the school’s 3d point cloud data and perform experimental processing. With this ros package, you can view lidar data pointcloud2 on rviz for visualization and quantitative analysis of experimental effects.

- Qiulongtian’s proposal:
  In paper SqueezeSegV2: Improved Model Structure and Unsupervised Domain Adaptation for Road-Object Segmentation from a LiDAR Point Cloud, they introduce a new model SqueezeSegV2 that is more robust to dropout noise in LiDAR point clouds. In paper Rangenet++: Fast and accurate lidar semantic segmentation, they focus more on LiDAR-only semantic segmentation forward in order to provide another independent source of semantic information to the vehicle.

  To be more specific, in paper SqueezeSeg: Convolutional Neural Nets with Recurrent CRF for Real-Time Road-Object Segmentation from 3D LiDAR Point Cloud , they address semantic segmentation of road-objects from 3D LiDAR point clouds, particularly for cars and pedestrians. Focusing on road-object segmentation so we use (Velodyne style) 3D LiDAR point clouds. Given point cloud output from a LiDAR scanner, the task aims to isolate objects of interest and predict their categories. To feed 3D point clouds to a 2D CNN, it adopt a spherical projection to transform sparse, irregularly distributed 3D point clouds to dense, 2D grid representations. The proposed CNN model draws inspiration from SqueezeNet and is carefully designed to reduce parameter size and computational complexity, with an aimto reduce memory requirements and achieve real-time inference speed for our target embedded applications, which is very similar to our aims. In 3D projection to asphere, we shouldn’t ignore something important, which they also mention but they hadn’t tackled with.
  For package, I recommend ROS velodyne_laserscan package, which is design to extract a single ring of a Velodyne PointCloud2 and publish it as a LaserScan message. In our project, we may need to process a small part of the point cloud so this package may be helpful in our development.

- Shiqianjing's proposal:
  In paper PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space, they introduced a hierarchical neural network that applies PointNet recursively on a nested partitioning of the input point set,called PointNet ++, which is able to learn deep point set features efficiently and robustly. In paper Pointnet: Deep learning on point sets for 3d classification and segmentation, they provide analysis towards understanding of what the network has learnt and why the network is robust with respect to input perturbation and corruption.

  However, I want to introduce this paper in detail, Fully-convolutional point networks for large-scale point clouds. Because it focus on efficiently processing large-scale 3D data. And in our project, we do need to collect data of 3D and is in large-scale.In contrast to conventional approaches that maintain either unorganized or organized representations, from input to output, its approach has the advantage of operating on memory efficient input data representations while at the same time exploiting the natural structure of convolutional operations to avoid the redundant computing and storing of spatial information in the network. Besides, it can produce either an ordered output or map predictions directly onto the input cloud, thus making it suitable as a general-purpose point cloud descriptor applicable to many 3D tasks. In our project, when we transfer a 3D-sphere into its projection, we will meet many problems in projection of how to tackle with the fact that small objects will be projected so small and computer is hard to recognize them so that they may be ignored. However, sometimes they are really important object-s such as pedestrians. As we all know, in autonomous driving, how to keep cars away them is a big question to keem pedestrians safe. So if we ignore datas of pedestrians, it will cause big problem.

  For package, I prefer velodyne_driver in my paper. This is a ROS 2 driver for Velo-dyne devices. It currently supports the 64E(S2, S2.1, S3), the 32E, the 32C, and the VLP-16. This driver is responsible for taking the data from the Velodyne and combining it into one message per revolution. If we want to turn that raw data into a pointcloud or laserscan, we can see the velodyne_pointcloud and velodyne_laserscan packages.This package provides basic device handling for Velodyne 3D LIDARs. For a list of all supported models refer to the Supported Devices section. The driver publishes device-dependent velodyne_msgs/VelodyneScan data. This driver supports all current Velodyne HDL-64E, HDL-32E, VLP-32C, and VLP-16 models. There is little difference in the way it handles those devices. For our project, we will use some data collecting from campus using 32-beam Lidar , which scans the whole campus. So I think this package is very important to our project.

4 System Description:

For large-scale point cloud segmentation, there are four paradigms for semantic segmentation: projection-based, discretization(voxel)based, point-based, and hybrid methods. Among them, the projection-based method is often more suitable for automatic driving and autonomous mobile robots.In our project, we plan to project a 3d point cloud in spherical coordinates based on the SqueezeSeg framework to obtain a 2d image. Semantic segmentation of the image is carried out through the deep neural network, and then the image is remapped to 3d to obtain the 3d point cloud after semantic segmentation.However,projection, this intermediate representation inevitably brings several problems such as discretization errors and occlusions.Because the 3d point cloud is projected to 2d, it will produce some details information loss. And after predicting the label for each pixel, once the labels are projected into the original point clouds, two or more points stored in the same range of image pixels will obtain the same semantic label.

The existing solutions to the problem of the lack of down-sampling detail information include Rangenet++ using range information to reconstruct to three dimensions, and using effective point labels KNN search. SqueezeSeg combines the CNN image prediction results and the original point cloud data into CRF to refine the label map. PointSeg uses dilated convolutional layers to obtain multi-scale features and broaden the receptive field. RandLA-Net expands the receptive field by dilated residual block.

Our current rough idea is to combine the depth map to supplement the detailed information on the basis of the projection-based method, and consider the point-voxe (PVCNN)l architecture to extract features more effectively.

We do not yet know whether it can be combined with the point-voxel framework and whether its effect is good.

5 System Evaluation:

We will test the semantic map computed by our algorithm using semantic KITTY and the point cloud data we scanned from our campus. The result will be evaluated mainly by two part, efficiency and effect. To be more specific, we may compare the memory usage, run time, mIOU and accuracy of our semantic map with other latest algorithms from others' paper. Except for the basic comparison, we may check the detail of semantic map with others' semantic map, which may focus on the accuracy of a small area of map.

6 Work Plan:

Time	Schedule
Proposal due to midterm report	Do some advanced survey for algorithm improvement and design our algorithm.
	Acquire the point cloud data of shanghaitech campus.
	Learn basic knowledge about Point Cloud and Deep Learning.
Midterm report to final report	Implementation of algorithm based on open source code
	Evaluate our code's effect by comparing with others' algorithm.

7 Conclusions:

In this paper, we will proposed SqueezeSeg as our projection-based method without bad segmentation performance than the original SqueezeSeg. We believe it can be used as a general-purpose feature descriptor by evaluating it on challenging benchmarks at different scales, namely semantic scene and part-based object segmentation. Overall, we hope our approach will be able to outperforms the state of the art both in accuracy and run time. Maybe our research can be a new way to optimize semantic segmentation for autonomous vehicles and robots.