Point Cloud Based Machine Learning for vehicle Target Detection and Visual Navigation Milestone Report

Xinyi Wang 12010239, Jianxiang Ni 11812723, Baoyi Huang 12010437,
Yujie Zhang 12010142, Wanghongjie Qiu 12010310

I. Introduction

Navigation and positioning is important for moving robot.SLAM is a normal method used to solve the localization and map construction problem.3D point clouds for robot navigation is commonly used in the industry.How to make robots understand point cloud, or the feature recognition of point cloud, has always been a hot problem in machine learning. In this project ,we use KITTI data set and reinforcement learning method to let the robot use point cloud signal for target detection and navigation decision.We simulate it in ros firstly and then we are going to implement it in the real world.

II. Problem Statement

Navigation and positioning is important for moving robot. SLAM (Simultaneous Localization and Map Construction) is a process in which a moving object calculates its position according to the information of the sensor and builds an environment map at the same time, so as to solve the localization and map construction problem of a robot moving in an unknown environment.Point cloud-based robot navigation means that the robot obtains three-dimensional point cloud information of the environment through lidar or other equipment, and then extracts key information in the environment, such as obstacles, ground and walls, through processing point cloud data, and finally carries out path planning and navigation according to the information3D point clouds for robot navigation is commonly used in the industry.How to make robots understand point cloud, or the feature recognition of point cloud, has always been a hot problem in machine learning.

III. Literature Review

Liu et.al.,wrote a training network of 3DCNN plus RNN algorithm to identify environmental information in point clouds[1]. Tiator et.al. proposed an application method of reinforcement learning in point cloud segmentation. This has a lot to do with our path planning.[3]Lobos et.al present a construction process of the whole learning simulation platform and the problems they encounter in the deployment of the robot platform, which is of great reference value for our construction.[2]

IV. Technical Approach

In object section part,we choose YOLOv5 model to accomplish our objection detection task, and Kitti dataset to generate our point cloud simulation in rviz of ROS.

A. Object Detection

1) YOLOv5 introduction: YOLOv5 is a object detection model that based on pre-trained object detection on the COCO dataset.It represents Ultralytics' open source research into future vision AI methods, incorporating lessons learned and best practices developed over thousands of hours of research and development.Comparing with traditional R-CNN deep learning algorithm,YOLOv5 have several advantages.

TABLE I
Advantages of YOLOv5

PRONS
Using the Pytorch framework user-friendly and easy to train your own data set
code is easy to read integrates a lot of computer vision technology very conducive to learning and reference
easy to configure model training is fast batch reasoning produces real-time results
Ability to reason directly on different resource even webcam port input effectively
Pytorch weight file can be easily converted to other format can direct deployment to the mobile app

2) current results: In this part,we use the camera shown in Fig.1 to capture the indoor environment.

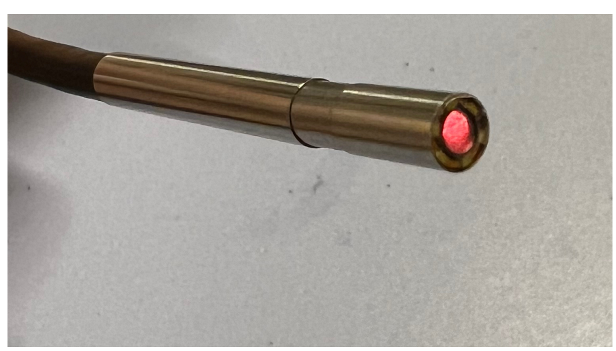

Fig. 1. Camera

Based on YOLOv5 model,we can easily use our camera to detect the specific object that we preset as the target before the experiment.In this experiment,The bottle and sport ball are put in the Camera visibility area.And the feedback video through YOLOv5 model is shown that the target we want to capture are surrounded by different colored squares. The test result is showed as Fig.2.

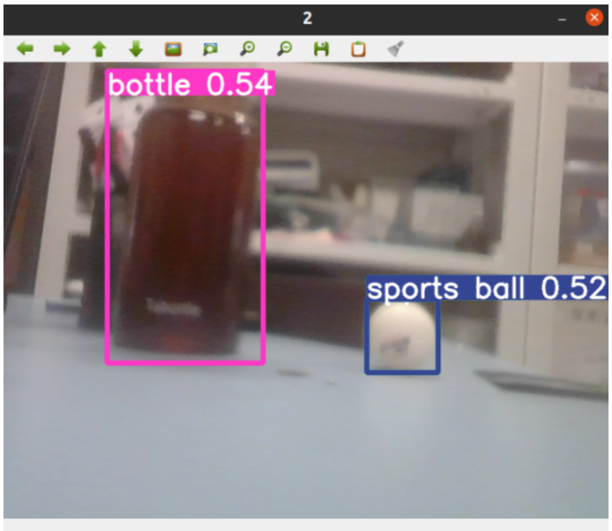

Fig. 2. Object Detection Result

B. ROS Point Cloud real-time target detection based on openPCDet and Kitti

1) Introduction: OpenPCDet uses a deep learning-based approach for 3D object detection from point clouds. It employs a two-stage detection pipeline, consisting of a proposal generation stage and a refinement stage. The key components of the OpenPCDet pipeline, such as the RPN and second-stage network, are typically based on deep neural networks, which are trained on large annotated datasets to learn to detect and classify objects in 3D point clouds.

2) process: On ROS, we subscribe the point cloud message and the release of the message of the detection box, so as to realize a visual function of the detection effect. First, the rotation parameter in openPCDet file is modified to set the radar deflection parameter with deflection Angle to 0. Then, change the threshold to remove inappropriate checkboxes. After that, each frame point cloud matching the detection results is published. Finally, the Kitti data set was converted into .bag format and tested with pointpillars model.

3) results: In Kitti dataset, a 3D real-state surrounding environment is generated in rviz, and we can see the roads, architectures, running cars, and walkers are well simulated by point clouds and the specific shapes can be recognized. So far the codes can generate a real world scenes using point clouds.

C. Simulation of Binocular Vision Navigation in ROS

Before finally implementing the binocular vision navigation in the real world, we plan to simulate it using ROS. It is divided into the following steps:

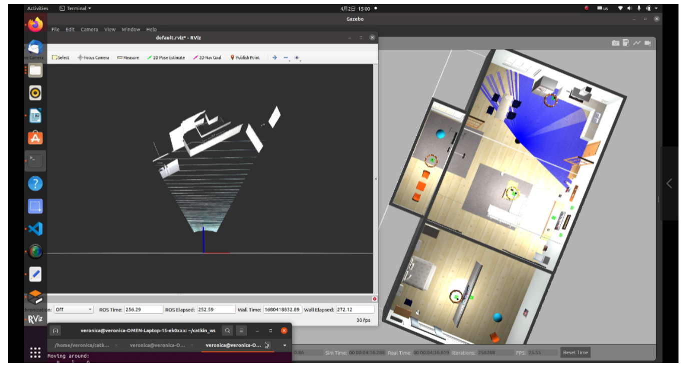

Fig. 3. navigation result

1) Build the ROS robot model and the Gazebo world model: The robot model is a simple model built based on the ROS project completed by our group members last semester, consisting of a base, 2 drive wheels, and a laser scan. We added a kinect depth camera to this model to complete the point cloud task. Gazebo indoor environment from GitHub: https://github.com/aws-robotics/aws-robomaker-small-house-world

2) Map creation using SLAM algorithm: Using the gmapping method from ROS to build the map of Gazebo indoor environment, the car robot is controlled by keyboard to make the laser scan the whole indoor environment to build the map.

3) Using camera Pointcloud to perceive the world: After writing the sensor code for the kinect camera, open Gazebo World and rviz, and add the relevant plug-in in rviz to see the robot's "vision" in the pointcloud state.

4) Realize autonomous robot navigation: Based on the fact that the map obtained in the second step may produce errors in actual operation, we consider using the cost map for navigation and adjusting the reference until the navigation works well. The 2D Nav Goal in rviz is used to set the purpose of navigation.

5) current result: The maps created are shown above, and videos of the relevant results of the simulations can all be seen in the submitted materials.

References

[1] Fangyu Liu, Shuaipeng Li, Liqiang Zhang, Chenghu Zhou, Rongtian Ye, Yuebin Wang, and Jiwen Lu. 3dcnn-dqn-rnn: A deep reinforcement learning framework for semantic parsing of large-scale 3d point clouds. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[2] Kenzo Lobos-Tsunekawa and Tatsuya Harada. Point cloud based reinforcement learning for sim-to-real and partial observability in visual navigation. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5871–5878. IEEE, 2020.

[3] Marcel Tiator, Christian Geiger, and Paul Grimm. Point cloud segmentation with deep reinforcement learning. In *ECAI 2020*, pages 2768–2775. IOS Press, 2020.