Project #5: Robotics-Academy: new exercise using Deep Learning for Visual Control

Name: ***Yiheng Zhou***

Country: China *Email: zhoum1440@gmail.com*
Twitter: @MiZhou24275429 *YouTube: @MIZhou-gl9zl*
Public repository: https://github.com/unswimmingduck

EDUCATION

School: University of Shanghai for Science and Technology *Yangpu, Shanghai, China*

Degree: junior undergraduate students

Major: Robotics

Personal statement: I'm a junior undergraduate student. My major is Robotics. In the undergraduate course, I learn lots of knowledge about the Computer Vision and AI and I have a great passion for these fields. In university, I take a part in a scientific search about automatic driving with a mentor. My main job is to train a point cloud recognition model so that we can use depth camera to detect the car, pedestrian and road. So, apart from scientific research, I want to have some practical experiences about project developing. So I want to join the JdeRobot community. I want to obtain more knowledges and applications about the Computer Vision and AI. And the JdeRobot GSoC really gives me a great opportunity to expand my perspective and coordinate with a constellation of prominent mentors. This project is also conducive to my graduation project which is aim to design an intelligent car that can achieve the goal of self-driving in a track. What's more, after this project, I can gain deeper insight into the Computer Vision and AI and this precious experience will become a stepping stone for my academic and career development. This is my first time applying for GSoC program. I like the original intention and concept of this project, GSoC is a valuable opportunity for people like me who want to gain experience. I extremely want to join in this program.

TIMELINE

Community Bonding *May 4– May 28*

In this period, my main aim is to learn about the JdeRobot community and integrate into the community. The aim can be dived into three parts:

- Learning the architecture of the web template.
- Learning the embedded lib such as the HAL and GUI
- Learning and building the simulated environment

Coding Phase 1 May 29 – June 25

There are 4 weeks in the Coding Phase 1. In this phase, I'm aim at training the naive model and let the car finish the track on the Gazebo by the naïve model.

Week 1 *May 29 – June 4*

- Developing the simulation environment on the Gazebo and installing some Libs that I may use in the next weeks.
- Sampling the relevant data (every frame of the photo and the corresponding linear velocity and angular velocity) and collating the data into datasets.

Week 2 *June 5 – June 11*

- Building a data loader for the naive model train.

- Building the skeleton of the CNN by PyTorch (reference: https://arxiv.org/abs/1604.07316).

Week 3 and **Week 4** *June 12 – June 25*

- Using the data that I collected in the week 1 to train the CNN.
- Deploying a control code with the naive model to the car.
- Testing the control code on the Gazebo.
- Modifying the model or the data to improve performance of the car.

Coding Phase 2 **June 26 – July 23**

There are 4 weeks in the Coding Phase 1. In this phase, I will focus my work on getting a simulated environment ready, building some widgets and updating the web interface for accepting models trained with PyTorch/Tensorflow as input.

Week 5 *June 26 – July 2*

- Getting a simulated environment ready and making sure that the web can connect to the Gazebo after the user press the launch button or view sim.
- Updating some widgets such as play, pause, and reset so that the users can operate the simulation as they want.

Week 6 *July 3 – July 9*

- Deploying the control code I made in week 3 and week 4 on the web.
- At the end of this week, the middle-stage assessment will start. I will show the mentor that I can successfully let the car finish the track using the code that I deploy on the web template.

Week 7 and **Week 8** *July 10 – July 16*

- Updating the web interface for accepting models trained with PyTorch/Tensorflow as input. In order to addressing this update, I will turn to the ONNX.js(https://github.com/microsoft/onnxjs) for help.
- Building a program that the model can work after the web receive the .onnx format as input.

Coding Phase 3 **July 17 – Aug 13**

There are 4 weeks in the Coding Phase 3. In this phase, although the work is close to draw, the work is still significant. My aim is to code the core application that will feed the trained model with input data and send back the results and add some new widgets to improve user experience.

Week 9 and **Week 10** *July 17 – July 30*

- Getting a simulated environment ready and making sure that the web can connect to the Gazebo after the user press the launch button or view sim.
- Updating some widget to make them more suitable to this exercise.

Week 11 *July 31 – Aug 6*

- Adding some widgets or interfaces to improve the user experience.

Week 12 *Aug 7 – Aug 13*

- Modifying some embedded Lib such as GUI and HAL to make the exercise easier to use.

Buffer Time **July 17 – Aug 13**

There are 8 buffer days in case something didn't go as planned in the weeks before

PROGRAMMING BACKGROUND

Computing Experience

Computer System

- Win 10, Ubuntu 18(in VMware)

Programming Language

- Python
- C/C++
- Django
- HTML
- Javascript
- PyTorch

Robot and Computer Vision Experience

Robot Experience:

- Developing a table tennis robot in Webots (a simulation software). Using computer vision the robot can predict where the table tennis ball will land and then the robot will plan the trajectory of the racket so that the ball will be hit back. (There are more details in my github or you can email me to know more about the source code.)
- Building an intelligent ROS car using slam to navigate.

Computer Vision Experience:

- Training a point cloud recognition model that is suitable to detect industrial parts. (Include making a virtual datasets by BlenderProc(a github project) and building suitable data loader and so on)
- Now, I'm taking part in a project about 3D point cloud instance segmentation in the university.

Preparation For This Project

I also make some preparations for this project. I find a reference named "End to End Learning for Self-Driving Cars" (https://arxiv.org/abs/1604.07316). What's more, I also build rough structure of the net. And you can see some details about the rough structure of the net and some advance preparations I make through the folder named "AdPre_for_GSoC2023" in my GitHub repository.