A Robust and Energy-efficient Control Policy for Autonomous Vehicles with Auxiliary Tasks

Xiaoxi Gong, Qiaoyun Wu, Dawei Li, Jingxuan Dong and Jun Wang

I. Supplementary Materia

In this supplementary materia, we provide more details about the experimental benchmark, the robot platform and the simulator used in section *Implementation and Performance* for our manuscript.

II. Benchmarks Used in Experiment

A. KITTI Raw Data Benchmark

The benchmark has 6 categories of recordings. Each category contains grayscale and color stereo sequences captured and synchronized at 10 Hz, and relative reading of sensors (e.g. 3D point clouds, GPS/IMU data and 3D object tracklet labels). The sensor readings are stored in text format, and each file contains 30 values, such as location of the oxts-unit, angular velocity and acceleration based on different coordinate systems etc. We pick the heading angle (e.g. positive, if counter clockwise and vice-versa) and the forward velocity (e.g. parallel to earth-surface) as indexes to evaluate our autonomous driving model. We use sequences of three categories, i.e., city, residential and road to train the autonomous driving model. For each sequence, 80% of images are selected for training and the other 20% of images are used for validating, and we select three sequences outside the training sets to evaluate the accuracy of the prediction.

B. Oxford RobotCar Benchmark

The benchmark is collected in different weather conditions, e.g. rain, snow and night etc. and road conditions, e.g., roadworks etc., over the period of May 2014 to December 2015. The dataset provides almost 20 million recordings that were collected from six cameras which are mounted on the vehicle. The dataset also provides LIDAR, GPS and INS ground truth. We use sequences of three different conditions, i.e., sun, clouds and overcast to train our autonomous driving model, and we use sequences that were captured under conditions of sun, rain, snow and night to evaluate the generalization of our autonomous driving model. At the training stage, we use sequences of three categories, i.e., city, residential and road in the KITTI Raw Data benchmark and sequences of three different conditions, i.e., sun, clouds and overcast to train our autonomous driving model. For each sequence, 80% of images are selected for training and the rest 20% of images are used for validating.

C. Mobile Robot Platform

Hardware configuration. We use a mobile robot, namely Pioneer3-AT (P3-AT) [1], in our experiment to achieve evaluations of the energy-efficiency by using different smoothness policies. P3-AT is a differential drive and all-terrain mobile robot with four rubber wheels, and it is able to pass through a variety of terrain surfaces, e.g., grass, soil. The robot uses a set of lead-acid batteries (the capacity of each battery is 12V and 7Ah typically) as power source. The robot is open-sourced and it supports multiple programming languages. It provides APIs which consist of several libraries, e.g., ARIA (motion control library), ArNetworking (data transmission library) and ARNL (localization and navigation library). In our case, ARIA is used to implement the robot control and ArNetworking is applied to achieve communication within a wireless LAN. ARNL is also used to implement the localization and navigation tasks for the robot. As Fig. 1 shown, P3-AT equips a RGB camera in the front side to capture the views in real-time. The camera is fixed at a height of 1.5 meter above the ground.

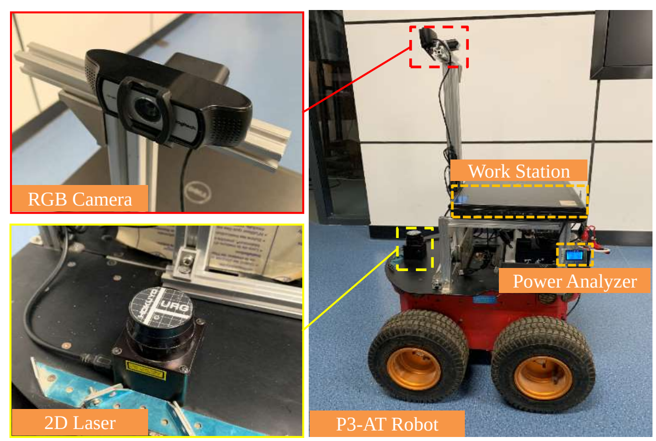


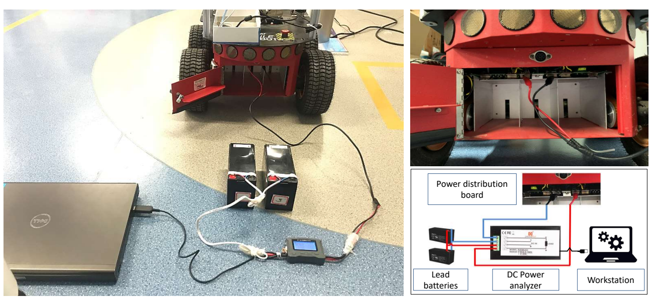
Fig. 1: The hardware configuration of P3-AT mobile robot.

A Hokuyo 2D laser (the model is: URG2.0) is mounted on the deck of the robot. Here, we use the ARNL library to implement the mapping, localization and avoidance in the outdoor environment, so as to collect the driving data automatically on Path 02, since there are pedestrians moving along this path. The strategy of avoidance is generated by the algorithm, which is much stable that human drivers. An on-board computer (work station) is amounted on the deck of the robot to process the image input, and then pass forward to our autonomous driving model to generate the driving commands. The on-board computer is powered by an external battery system, since the robot's lead-acid cells can not provide sufficient electricity. In addition, we create a local area network (LAN) to build a connection between the server and client terminal. The server terminal, which is the on-board computer, is fixed on the mobile robot to handle

[1] http://wiki.ros.org/action/show/AdeptMobileRobots

the sensor data, e.g., pose, velocity and real-time image, and then transfers the data stream to the client terminal (another laptop), by using TCP protocol. So that, volunteers can drive the robot remotely and we can achieve the remote monitoring, in real-time. Besides, we add a power analyzer [2] between the battery sets and the power distribution board of the robot to measure the instantaneous power when the robot is running. The connection detail is shown in Fig. 2.

Generally, a mobile robot will create a higher instantaneous current when starting to move from a static state. Different road conditions such as roughness of the road surface and the slope could also cause the distinct electricity consumption. For instance, a mobile robot needs a high start-up current when it starts to move from a resting state. On the contrary, it needs relatively lower current under the state of uniform motion. Hence, the most efficient way to manipulate a mobile robot is to keep it moving in a straight line with constant speed as much as possible. The ever-changing speed value and steering angle can lead to relatively higher electricity consumption for an autonomous driving vehicle.



(a) (b)

Fig. 2: The connection detail of the power analyzer (a). The power analyzer is connected in series between lead-acid batteries and the power distribution board (b) of the robot.

Driving data collection. Due to the limitation of P3-AT's battery capacity and the time resource, we only invited three volunteers to drive the robot manually by using the camera's live stream as reference. As a result, the collected datasets may be insufficient. However, since Section IV-B only focus on evaluations of the energy-efficiency, we use the same lanes (Path 01, Path 02 and Path 03) to implement the data collection and the testing. For Path 01 and Path 03 (daytime and nighttime), each volunteer had two chances to drive the robot, since the first driving is inexperienced and the second one would be relatively smooth and efficient. For Path 02, we advice the volunteers to keep a certain distance (approx. 1.5m) with pedestrians during driving. However, the results are not satisfied, since it is relatively hard for a new hand to estimate distance visually. Hence, we use P3-AT's 2D laser and the ARNL library to create 2D map (as shown in Fig. 3), then to achieve navigation and avoidance automatically. At the end, we collected 24 samples

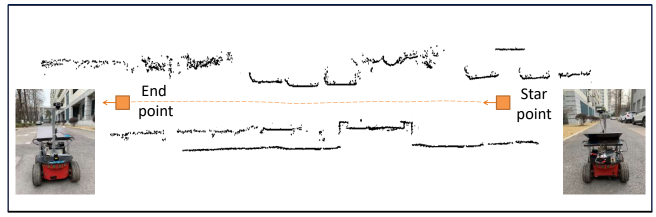


Fig. 3: 2D mapping, navigation and avoidance by using the ARNL module.

and we use these samples and the benchmarks (e.g. KITTI, RobotCar) to train our upcoming-view synthesis model and autonomous driving model, respectively. Note that, for the benchmarks, the ground truth of predicted vehicle speed (unit: kilometre per hour) is mapped to the range of [0, 1.2] (unit: meter per second) which is P3-AT's minimal and maximum velocity range.

D. *Udacity Self-Driving Car Simulator*

The Udacity self-driving car simulator [3] is an open source project, which aims to clone driving behavior through deep neural networks and convolutional neural networks. The simulator allows us to collect images and the relative control commands, which can be used in training, validating and testing. In addition, the simulator supports manipulation through the interface of Keras and this allows us to do the online test. The simulator uses the left and right arrow keys on the key board to control the steering angle which has range of [25, 25] (unit: degree); The up arrow key is used to control the forward velocity and the maximum speed value is up to 30 MPH (miles per hour). Three virtual cameras are deployed on the head of the vehicle, which are able to capture the right, center and left views of the scene. In experiments, we use the simulator to determine the relationship between driving behavior and driving efficiency. Specifically, we invited 10 volunteers to join the simulated driving under different virtual scenes, in order to record their control commands which imply the unique control behavior. For each scene, each volunteer plays a round trip for 3 times, so we collect 60 samples in total.We encourage people to drive the virtual car with efficiency and safety, in which the unnecessary operations and the failure cases can be avoided as much as possible.

[2] https://peacefair.en.made-in-china.com/product/jsnElmKyZrVB/China-Chinese-Manufacturer-DC-6-5-100V-20A-4in1-LCD-Electronic-Kwh-Meter-Voltmeter.html

[3] https://github.com/udacity/self-driving-car-sim