An open source self-driving podcar platform for automation research

Yao Chen*, F. Camara*†, Gabriel P. P. Walton†, Chris Waltham†, C. W. Fox*†
*Institute for Transport Studies, University of Leeds, UK
† College of Computer Science, University of Lincoln, UK
‡ Ibex Automation Ltd., UK

***Abstract*—Here......**

***Index Terms*—Autonomous Vehicles, Pedestrian - Vehicle Interaction [3 maximum]**

I. Introduction

[?] presents a game theory model...

[?] did some experiments with human participants....
The growth of automotive industry was accompanied by the Industrial Revolution and it had designed vehicles with a completely mechanical system of safety, driving and interaction for human drivers. The structure of human-driving vehicle decided the current transport system related to road design, congestion condition, travel behaviour, policy designing, establishment of public infrastructures and so on. However, now automotive industry is facing a new revolution: autonomous technology, and might be turned upside down by it caused by the development of computerization over the past century. The impacts of autonomous vehicles(AVs, also called automated or self-driving vehicles) would not only relate to automotive industries, but also spread to transport system, changing the whole process of people's travelling behaviour. Thus, increasing automotive industries and various new technology companies all involve in AVs research to catch this opportunities while governments and global organizations launched various policies and specifications to promote and standard this field.

Driving automation is defined in 5 levels: driver level, assistance, partial automation, high automation, and fully automation [?]. Recently many semi-autonomous features had been commercially available. For example, General Motors (GM) focused on a semi-automatic prototype - the Super Cruise - especially for the congested condition and highway cruise in Cadillac line [?]. Park Assist Pilot was also successfully applied in high product lines of Volvo and Mercedes Benz. But full autonomous technology were still limited in deep learning mainly by mature companies and still needed a long way to achieve. Google self-driving car was a well-known sample. It was firstly announced in 2010 for the purpose of traffic accidents reduction with the prediction of launching the product within one decade by project director Chris Urmson, but in 2014 he indicated though the major technology of fully computer-control had been worked out, Google AVs might be postponed to 30 years later due to the challenge of real word interaction with pedestrians [?][?].

This project has received funding from EU H2020 interACT: Designing cooperative interaction of automated vehicles with other road users in mixed traffic environments under grant agreement No 723395.

The competitions of AVs development among different companies are increasingly intense. It required employees in various background including mechanical engineering, software engineering, robotics, data science, transport, etc. Students should prepare the relative learning experience for future career. However, funding students a platform for driving automation research is significantly costly. Only big companies and organizations could afford it to limited top universities such as in UK, Institute for Transport Studies, University of Leeds as a top educational organization had the opportunity of doing research with Westfield pod car which cost over 100,000 GBP. Most students in normal universities are only taught in theories especially in developing countries. An inexpensive platform of self-driving car for automation research is required for academic purpose.

This paper's aim is to show a process of modifying the operating system of an electric pod car to self-driving system.

For academic purpose, driving automation [?]

II. Related Work

About 20 references or more if you can (at least one page) about similar work, SLAM, vehicle control and simulation, open source projects....

Cite this project from UC berkeley: http://www.barc-project.com/

Trajectory: [?]

III. Mechanical Design

The PIHSIANG TE-889XLSN cabin scooter was used as the pod car platform to be converted with a autonomous driving system. It was an electric car powered by two 12V batteries connected in series to provide 24V operating voltage and contain 75Ah. The steering was controlled by human operating via a loop handle bar, while speeding and braking system were both powered by an electric motor and an electric brake via the trans-axle assembly, controlled by AC2 digital controller receiving different voltage signals to drive forward or brake. The manual speeding and braking systems were controlled by three buttons connected in series on the handle bar: A toggle switch in parallel with a resistor (10k) to choose speed mode high or low; A speed dial knob via a variable resistor (20k) to choose a maximum speed value; A throttle lever connected with a potentiometer (5k), 2.5k to 2.6k for each side to speed or brake. To achieve driving control by computer commands, via an micro- controller a linear actuator mounted under the chassis to car's front axle via bearings could be controlled to replace the steering column function. Moreover, micro-controller could also spoof the corresponding electric signals for the required speed to the AC2 digital controller instead of the output via the three speed buttons.

GLA750-P 12V DC linear actuator with position feedback was used with 8mm/s full load (750N) speed and 250mm stroke length (installation length is 390mm). Due to the existing hole in the right front wheel axle, mount the linear actuator via rear hole to the left side of front chassis and connected it through the front hole of actuator with the hole in the car's right front wheel axle via bearings shown in Fig. **??**. The front wheels could be turned by micro-controller sending commands to linear actuator for extending and retracting. The relationship between the required central turning angle θ of the pair of front wheels and extending length l of linear actuator is,

$$\theta = \alpha - \arctan(\frac{W}{2H}) \tag{1}$$

$$\beta = \alpha - \frac{\pi}{2} \tag{2}$$

$$x = r_1 * \cos(\beta) \tag{3}$$

$$y = r_1 * \sin(\beta) \tag{4}$$

$$l = \sqrt{(x_0 - x)^2 + (y_0 - y)^2} - L + l_0 \tag{5}$$

Where r_1, x_0, y_0, W, H and L are the geometric coefficients shown in Fig.1. Among them, the value of y_0 is negative. l_0 is the initial value of the linear actuator position feedback.

- talk about using Arduino to drive linear actuator and about PID controller.Confuse about whether to combine this part with Speed control code or separate them.

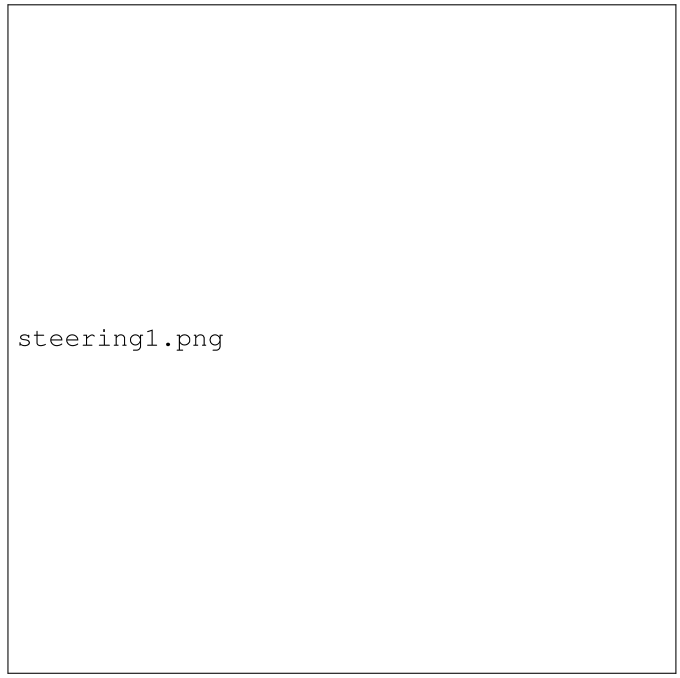

Fig. 1: The bottom view of front wheels steering relationship including geometric coefficients

- insert Figure 2 about the wiring diagram of speed buttons and micro-controller.
- talk about safety (dead-man button).And speed code part with details.

IV. ELECTRONIC DESIGN

Arduino UNO was used as the micro-controller to send electric sigals to vehicle's motor controller instead of using speed buttons. The ranges of voltage in different speed mode were measured shown in Table 1. Thus, the range of electric signal provided by Arduino UNO is 2.54V-3.59V for forward speed from 0mph to 4mph, , and 1.03V-0V for back speed with a negative correlation. It was also found 1.74V was the stop voltage for vehicle to work when starting, corresponding the speed throttle lever in the neutral position. Otherwise a bleep error would come out for safety reason and fail to start the car.

Speed mode		Forward	Back
Low	min	2.54	1.03
	max	3.08	0.50
High	min	2.69	0.89
	max	3.59	0

V. TRAJECTORY PLANNING

Talk about the simple car model(use a figure and talk about posture(including the position and orientation)) and assume it will only drive forward. see [**?**] how to describe the whole path

To calculate the whole car path in a map, the trajectory between a pair of postures (p_1,p_2) is the unit. Because the speed of car turning angle is a non-linear variable, using a set of spiral curves is not suitable for current steering system due to the continuing curvature changes of the trajectory. With the time limitation of this dissertation, this paper will use two circular arcs to compose the common trajectory unit. It is one of the simplest way to build a trajectory of two postures. Besides, for the initial version, it can allow car driving on the path with the smallest deviation regardless of the shortcoming of the stop situation on tangent point of the two circular arcs. The rest paragraphs will describe the process of building this trajectory model and required formulas in computer logic.

Firstly, for a pair of postures which can be linked by a straight line, two circular-arc segments is not necessary. Its trajectory is common and simple. Thus, here we only show the way of judging this condition shown in the bellow equation:

$$\theta_1 = \theta_2 = \arcsin(\frac{y_2 - y_1}{d_0}) = \arccos(\frac{x_2 - x_1}{d_0}) \tag{6}$$