Independent Project

2018 Fall - CSIT 6910A

VR Car Driving Practice

Department of Computer Science and Engineering

Hong Kong University of Science and Technology

Supervisor

Prof. David ROSSITER

Airy, Tsen-Jung TAI

Abstract

The traditional media usually face the difficulty of supporting spatial perception and only allow visual and audio information, which makes experienced-based learning impossible. The emerging of the VR technology raises a solution by creating an immersive environment and providing real-time interactive response. Moreover, large amount of money cost or loss of property can be avoided due to the virtual feature. Driving a car is a task the learning experience of which can be well captured and supported using VR technology, without possibly causing real-world damage or hurt. This project targets at offering the driving beginners a close-to-reality training through VR environment with a focus on reverse parking. Users are equipped with Oculus Rift head mounted display and digital driving simulator rigs to experience real driving control. The interior of the car is designed to be similar to a real one with a dashboard, a rear mirror, a gear shifter and an in-car monitor. The car moving algorithm is constructed based on real physics. The application is developed with Unity engine.

Acknowledgement

I would like to express my sincere appreciation to those who have inspired, encouraged and assisted me during the development of this independent project as well as the video and report. Special thanks go to my supervisor Professor David Rossiter, who has made every effort to support me with the Oculus Rift and driving simulator rigs, and has been guiding me and share his expertise with me from the very beginning towards the accomplishment of this project.

Table of Content

Table of Figures

1. Introduction

The VR technology can provide people with a more immersive and engaging experience than traditional media. It allows some real-world activities to be simulated and help the users receive training without causing any potential damage on properties.

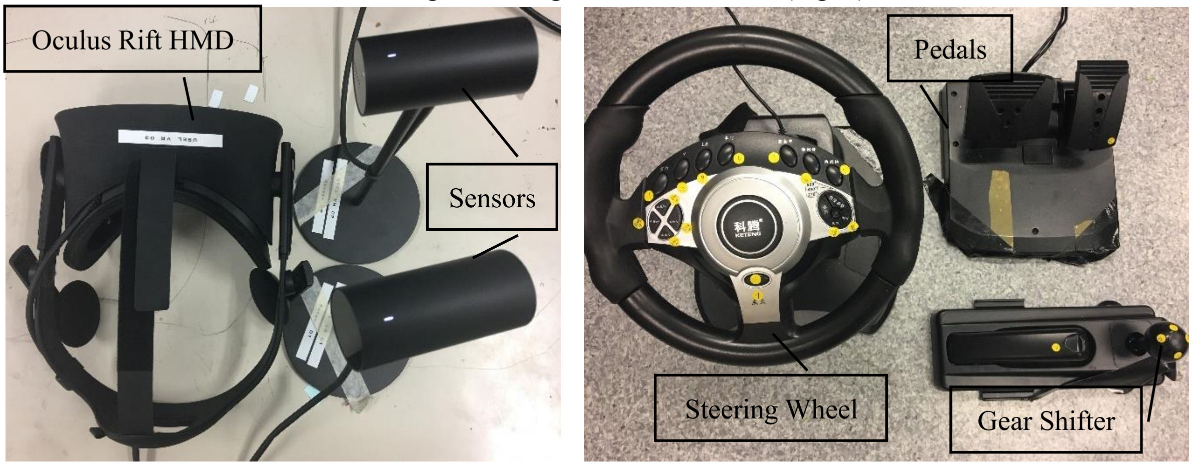
Learning to drive a car is a task which carries the risk of real damage and requires trainings under different situations. And reverse parking is one of the most difficult tasks among those relevant to car driving. There are plenty of related video tutorials online, but watching the clips only gets the users to know about the knowledge without any hands-on practice.

This project aims at preparing the users with basic skills needed to reverse parking via real-world control of digital driving rigs and real-time visual feedback through the VR head mounted display, in a way free from possibility of loss.

2. Setup

2.1 Platform Setup

This project is written in C# and is developed with Unity, a cross-platform game engine. To support VR and simulation features, Oculus Rift and digital driving simulator are used (Fig. 1).



(a) Oculus Rift set. (b) Digital driving simulator.

Figure 1 Oculus Rift and digital driving simulator rigs

2.2 Flow Design

This driving tutorial can be divided into two parts: the instruction part and the challenge part. The former gives a basic description of how to interact with the virtual world using the devices equipped, and the users can select the traffic rule which they are used to at the end of the instruction scene (Fig. 2).

Figure 2 Screenshot of the scene of traffic rule options

The challenge part contains six different levels with increasing difficulty which allows the users to get experience-based practice and gain necessary skills on car driving step by step. The goal of the challenge is to park the car controlled by the users inside a target slot identified by a green box, and no collision is allowed during the entire process. There is an in-game menu which provides the users with options to adjust the car mirrors and toggle the in-car monitor. The settings of the levels are as shown below, and please note that (1) the silver car is the one controlled by the users; and (2) each level inherited the constraints from its previous level; and (3) these screen captures are taken from a viewport which is different from the in-game aspect and are shown here only to help with understanding.

Level 1. Park the car next to another obstacle car (Fig. 3).

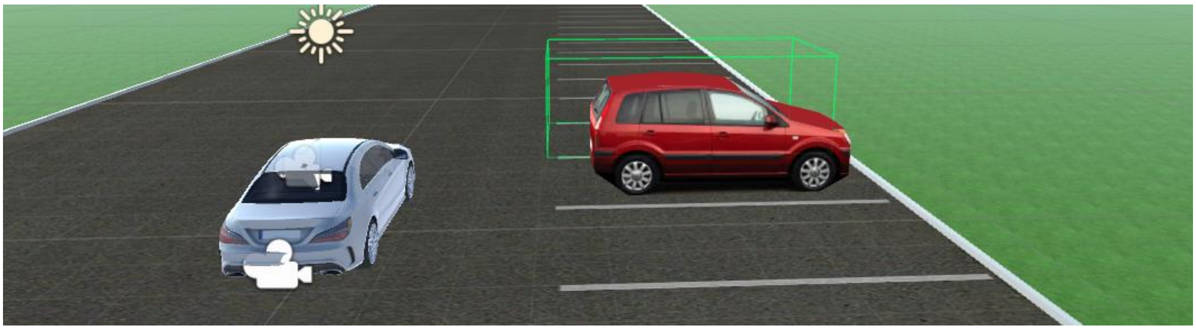
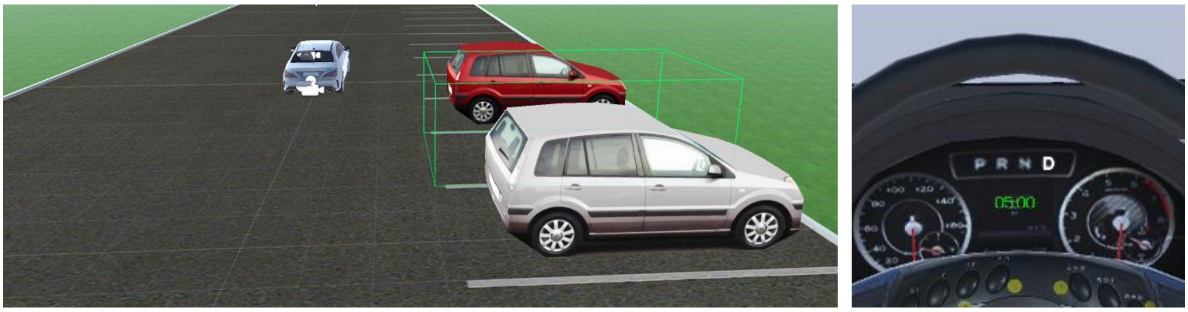


Figure 3　Level 1

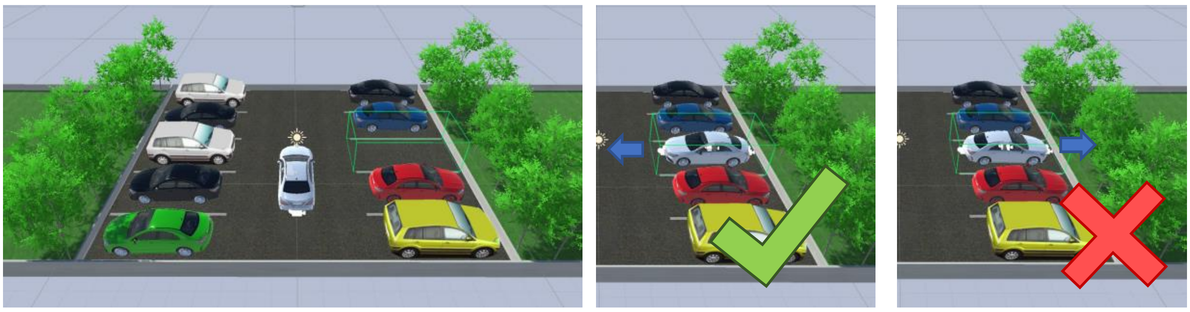
Level 2. Park the car between two obstacle cars, and the time limit is 5 minutes (Fig. 4).



(a) Level setting.　(b) The timer on dashboard.

Figure 4　Level 2

Level 3. The space is more limited, and the car must be reversed in (Fig. 5).



(a) Level setting.　(b) Correct reverse-in.　(c) Incorrect.

Figure 5　Level 3

Level 4. The terrain has certain slope (Fig. 6). The car will be dragged downhill quickly when in N mode or when the wheels are not held static.

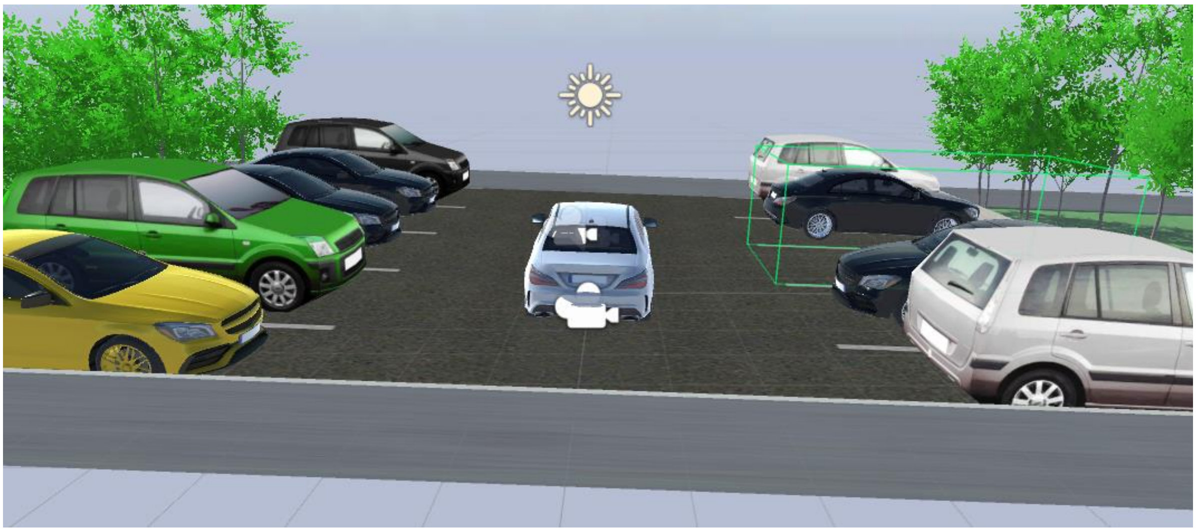
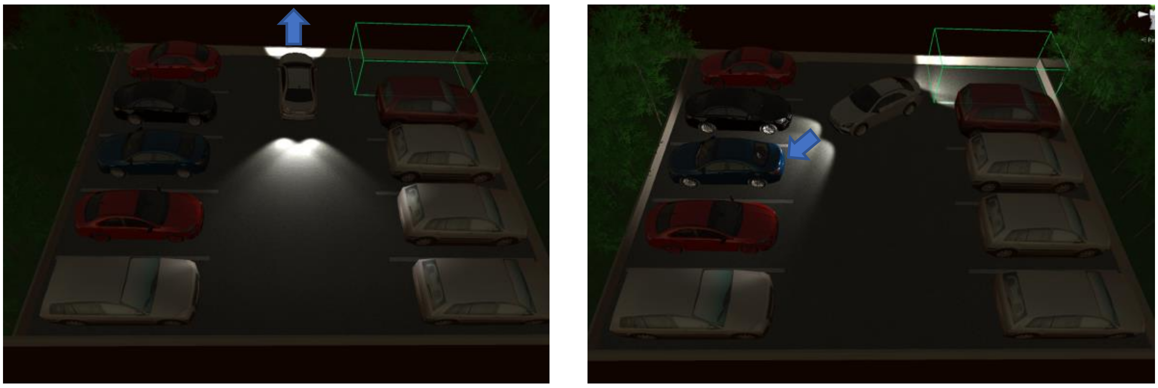


Figure 6 Level 4

Level 5. Poor lighting condition (Fig. 7). Note that in this level, it is easier to do reverse-in rather than forward-in due to the very limited space behind the car.

Figure 7 Level 5

Level 6. Not enough space is provided ahead of the target slot for reversing-in so that the users have to adjust the orientation of the car head first by utilizing the lane space (Fig. 8).



(a) Very limited space ahead of the target slot, which makes reverse parking impossible.

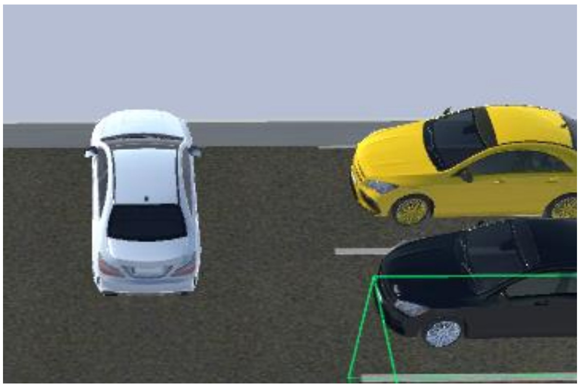
(b) Users have to adjust the orientation of the car so that more space is available.

Figure 8 Level 6

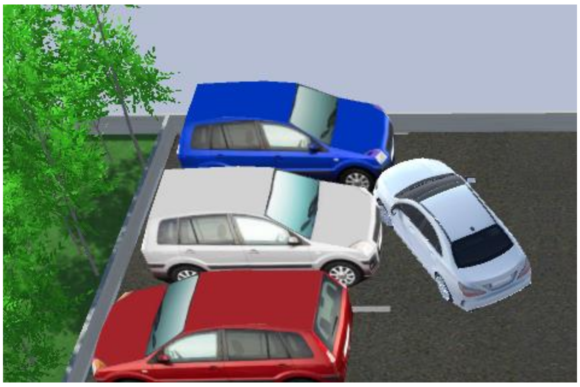
At the beginning of each level, a popup tip is displayed to help the users recall the task of the current level (Fig. 9).

Figure 9 The Popup tip shown the task of each level

Either hitting a wall or an obstacle car will result in failing the current level (Fig. 10). Since the purpose of the tutorial is to help people with safe driving in the real world, no collision should be tolerated during training.



(a) Birdview of hitting the wall.   (b) Users view when hitting the wall.



(c) Birdview of hitting obstacle cars.   (d) Users view when hitting obstacle cars.

Figure 10 Failing screen due to collision

The users can call the setting menu at anytime in game to adjust the angles of the mirrors or toggle the in-car monitor (Fig. 11). This provides the freedom to customize the mirror positions to suit various needs for people with different heights and makes the tutorial more practical.

(a) The menu with options of mirror adjustments and in-car display.

(b) The screenshot after selecting and adjusting the left mirror. Note that there is a popup tip about how to perform specific actions using the input devices, and the different reflections shown in the left mirror.

Figure 11 Menu screen

At the end of the trial, a summary of the performance is provided for review (Fig. 12), so that the users can track their progress for each round of practice.

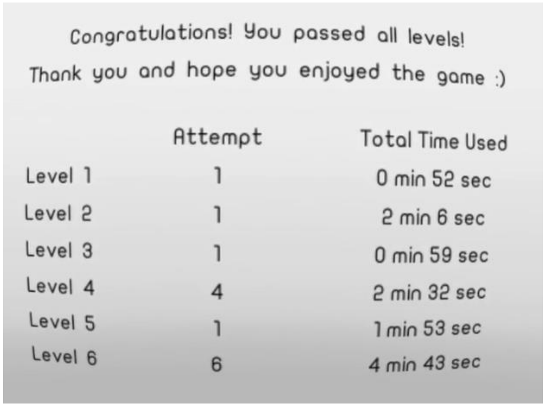


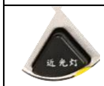
Figure 12 Review screen

3. Implementation

3.1 Device Input

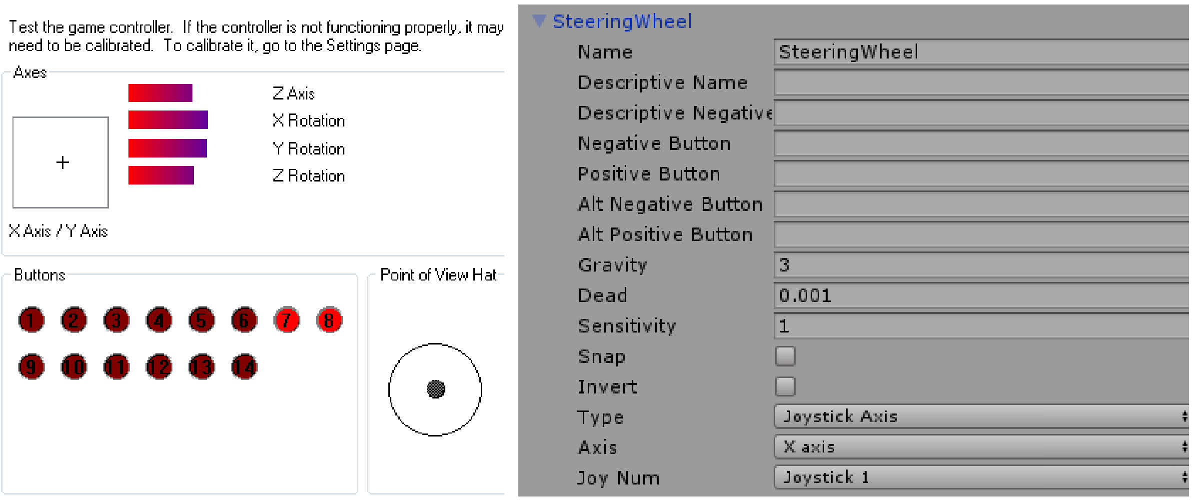
The Oculus Rift input is handled by Oculus Software [1] and its unity integration. The digital driving simulator signals are divided into two types: axis signals and button signals. The axis signals are captured via Unity input setting which requires to be manually defined, with their names assigned by the joystick driver (Fig. 13), as shown below, and please note that only those signals used in this project are listed.

Usage	Name	Joystick Port	Description
Car Control	Steering Wheel	X Axis	When the steering wheel is turned left, a negative signal ranging between 0 to -1 is returned; when turned right, a positive signal ranging between 0 to 1 is returned; both are through X axis.
	Accelerator and Brake	Y Axis	When the pedal for acceleration is pressed, a negative signal ranging between 0 to -1 is returned; when the pedal for braking is pressed, a positive signal between 0 to 1 is returned; both are through Y axis.

Mirror Control	Turning Light	5th Axis		When the left turning light button is pressed, a negative signal ranging between 0 to -1 is returned according to the force applied.
				When the right turning light button is pressed, a positive signal ranging between 0 to 1 is returned according to the force applied.
	Headlight	6th Axis		When the high beam button is pressed, a positive signal ranging between 0 and 1 is returned according to the force applied.
				When the low beam button is pressed, a negative signal ranging between 0 and -1 is returned according to the force applied.

The button signals are captured in the same way the keyboard inputs are captured, with their names assigned by the joystick driver as shown below (Fig. 13).

Usage	Name	Joystick Port	Description	
Toggle Menu	Y	Button 3		When the button Y is pressed, a signal of button down of code 3 is sent.
Gear Shifter	Push Shifter	Button 5		When the gear shifter is pushed forward, a signal of button down of code 5 is sent.
	Pull Shifter	Button 4		When the gear shifter is pulled backward, a signal of button down of code 4 is sent.



(a) The joystick driver determines the calling name of each button or axis.

(b) The example of connecting the input port in Unity and the output port of the joystick.

Figure 13 Joystick driver and Unity input setting

3.2 Driving Algorithm

The core of the project is the implementation of the driving algorithm, which is designed to simulate real-world physics, taking motor torque, brake torque, aerodynamic drag, rolling resistance and gravity into consideration [2]. The motor torque and brake torque are applied when the accelerator or braking pedals are pressed, and are calculated according to the depth of pressing. The air drag depends on the current velocity of the car and is proportional to the square of the speed. It is calculated as

$$F_d = -C_d * V * |V|,$$

where C_d is a constant, and the negative sign indicates the force prevents the car from moving in the direction of the current velocity.

The rolling resistance, also determined by the current velocity, represents the friction between the wheel and the ground. It is defined as

$$F_r = -C_r * V,$$

where C_r is also a constant, and the negative sign depicts it as resisting force.

The effect of the gravity is computed as

$$F_g = mg\, sin\theta,$$

where g is the acceleration due to the gravity, normally measured as 9.80 m/s^2, and θ is the slope of the terrain. The force remains zero until entering those levels with sloping ground.

3.3 In Target Detection

Whether the car is parked in the target slot is determined by bounding boxes checking [3]. Each vertex of the bounding box of the car is checked one-by-one that whether it is inside a 3D bounding box indicated using the green lines in game. The method is shown below (Fig. 14):

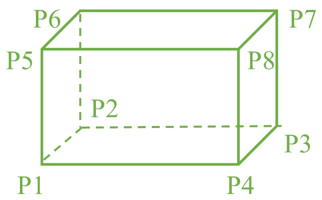


Figure 14 Bounding box of the target parking location

When all the edges are perpendicular, first compute:

$$u = P_1 - P_2,$$
$$v = P_1 - P_4,$$
$$w = P_1 - P_5;$$

A point p lies inside the bounding box only if the following conditions are satisfied:

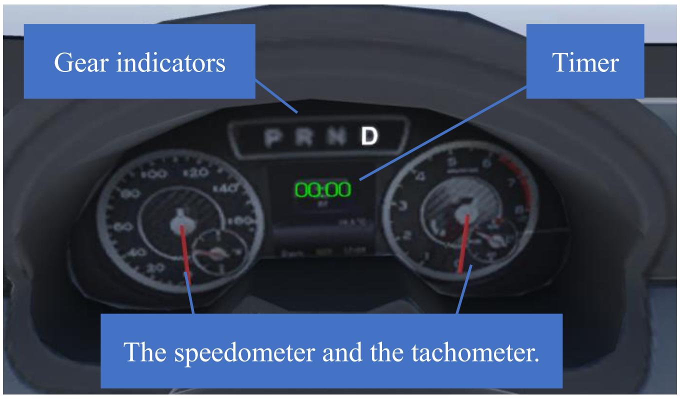
$$u \cdot p \text{ lies between } u \cdot P_1 \text{ and } u \cdot P_2,$$
$$v \cdot p \text{ lies between } v \cdot P_1 \text{ and } v \cdot P_4,$$
$$w \cdot p \text{ lies between } w \cdot P_1 \text{ and } w \cdot P_5.$$

3.4 Randomize Settings within Levels

To increase the practicability, the target slots are randomly chosen each time at the beginning of a level, and the obstacle cars are randomly picked from a set of cars with different styles and colors.

3.5 Visual Feedback

To get the users with better understanding of how their controls affects the movement of the car, multiple visual feedbacks are provided (Fig. 15). The steering wheel in the virtual world rotates synchronously with the one in the real world controlled by the users. The speedometer and the tachometer on the dashboard indicate the current speed and rpm of the car. If the users turn their head, they can see a gear shifter as the one in a real car and read the light indicators next to the shifter or on the dashboard to know which gear mode the car is currently in. The in-car monitor shows a birdview of the car on its left and on its right a forward view when the car in P, N or D mode, or a backward view when the car in R mode.



(a) The view of dashboard.

(b) The in-car display and the gear shifter.

Figure 15 The visual feedbacks

4. Conclusion

The project utilizes the power of virtual reality to provide its users a real-life-like environment to gain car driving skills without taking any risk of causing real damage. It is designed to simulate the real-world physics and driving operations as much as possible, with freedom attached to the change of four gear modes and the adjustment of mirror, and with the visual feedback from the dashboard and the in-car display. The levels are created in a way avoiding a steep learning curve so that the users can gain necessary knowledge and feelings of control gradually to achieve the final goal. There is still much work left to modify the project and help it reach the commercial level. Adding different whether conditions is believed to be a rewarding one, such as a rainy level which the driver's view is blurred, or a snowy level with snow covering the road, resisting the car movement while leaving the ground uneven. With more advanced programming and data training, some obstacle cars with machine intelligence can be developed and added to drive out or into parking slots when the users are performing their own parking tasks. This will lead the users to have some interesting interactions with other cars that they shall experience in real life, and force them to think about both the spatial and temporal efficiency.

This project also presents the possibility of taking the advantage of VR technology to produce practical training with low cost and safety environment which can hardly be achieved by traditional means. And VR technology is believed to play a role in not only entertainment, but also the education and training field in the future.

Reference

[1] Oculus [Online]. Available: https://www.oculus.com/ [Accessed: 2018, Sep 21]

[2] P. Srisuchat, "Development of a car physics engine for games," Bournemouth University, 2012. [Online]. Available: https://nccastaff.bournemouth.ac.uk/jmacey/MastersProjects/MSc12/Srisuchat/Thesis.pdf [Accessed: 2018, Sep 30]

[3] Check if a point is inside a rectangular shaped area (3D). Mathematics. [Online]. Available: https://math.stackexchange.com/questions/1472049/check-if-a-point-is-inside-a-rectangular-shaped-area-3d [Accessed: 2018, Oct 20]

Appendix: Minutes of Meetings

1st Meeting

Date: Thursday, 27 Sep 2018

Time: 5 p.m.

Place: Room 3554

Attending: Prof. David Rossiter, Airy TAI

Absent: None

Recorder: Airy TAI

Approval of minutes

This is the first meeting, no minutes to be approved.

Report on Progress

Some interesting topics which are worth to be worked as a project.

Discussion Items and Things to Do

The devices and potential technical problems of the project of car driving.

Meeting adjournment

The meeting was adjourned at 5:30 p.m.

2nd Meeting

Date: Friday, 19 Oct 2018

Time: 2 p.m.

Place: Room 3554

Attending: Prof. David Rossiter, Airy TAI

Absent: None

Recorder: Airy TAI

Approval of minutes

The minutes of the last meeting were approved without amendment.

Report on Progress

The driving algorithm was completed.

Discussion Items and Things to Do

Add visual feedback and levels with increasingly difficult constraints.

Meeting adjournment

The meeting was adjourned at 2:15 p.m.

3rd Meeting

Date: Monday, 29 Oct 2018

Time: 12 p.m., Midday

Place: Room 3554

Attending: Prof. David Rossiter, Airy TAI

Absent: None

Recorder: Airy TAI

Approval of minutes

The minutes of the last meeting were approved without amendment.

Report on Progress

Add in-target detection and first 3 levels.

Discussion Items and Things to Do
Add more visual feedback to assist the users to perceive the input controls. Fix some of the controlling problems.
Meeting adjournment
The meeting was adjourned at 12:30 p.m.

4th Meeting
Date: Wednesday, 14 Nov 2018
Time: 12 p.m., Midday
Place: Room 3554
Attending: Prof. David Rossiter, Airy TAI
Absent: None
Recorder: Airy TAI
Approval of minutes
The minutes of the last meeting were approved without amendment.
Report on Progress
The development of the application was mostly done, with gear shifter, dashboard, mirrors and another 2 levels added.
Discussion Items and Things to Do
Future minor fixes included (1) adding the P and N gear modes; (2) modifying the failing message; (3) spliting the level 5 into 2 levels with one requiring no adjustment of the car head orientation and another does. Start the plan of the video.
Meeting adjournment
The meeting was adjourned at 12:30 p.m.