



UNIVERSITI TEKNIKAL MALAYSIA MELAKA

**DRIVE RIGHT: DRIVING SIMULATION TO ALERT
THE DRIVER USING ARTIFICIAL INTELLIGENCE
FOR PRUDENTIAL DRIVING BEHAVIOUR**

This report is submitted in accordance with the requirement of the Universiti Teknikal Malaysia Melaka (UTeM) for the Bachelor of Computer Engineering Technology (Computer Systems) with Honours.

by

NOOR ANISAH BINTI A SULAIMAN

B071610602

971206385092

FACULTY OF ELECTRICAL AND ELECTRONIC ENGINEERING
TECHNOLOGY (FTKEE)

2019

BORANG PENGESAHAN STATUS LAPORAN PROJEK SARJANA MUDA

Tajuk: Drive Right: Driving Simulation to Alert the Driver Using Artificial Intelligence for Prudential Driving Behaviour

Sesi Pengajian: 2019

Saya **Noor Anisah Binti A Sulaiman** mengaku membenarkan Laporan PSM ini disimpan di Perpustakaan Universiti Teknikal Malaysia Melaka (UTeM) dengan syarat-syarat kegunaan seperti berikut:

1. Laporan PSM adalah hak milik Universiti Teknikal Malaysia Melaka dan penulis.
2. Perpustakaan Universiti Teknikal Malaysia Melaka dibenarkan membuat salinan untuk tujuan pengajian sahaja dengan izin penulis.
3. Perpustakaan dibenarkan membuat salinan laporan PSM ini sebagai bahan pertukaran antara institusi pengajian tinggi.
4. **Sila tandakan (X)

SULIT*

Mengandungi maklumat yang berdarjah keselamatan atau kepentingan Malaysia sebagaimana yang termaktub dalam AKTA RAHSIA RASMI 1972.

TERHAD*

Mengandungi maklumat TERHAD yang telah ditentukan oleh organisasi/badan di mana penyelidikan dijalankan.

TIDAK
TERHAD

Yang benar,

Disahkan oleh penyelia:

.....

Noor Anisah Binti A Sulaiman

Alamat Tetap:

No 40 Jalan Laksamana,

Rumah Awam 1,

35800 Slim River.

.....

Dr Jamil Abedalrahim Jamil Alsayaydeh

Cop Rasmi Penyelia

Tarikh:

Tarikh:

*Jika Laporan PSM ini SULIT atau TERHAD, sila lampirkan surat daripada pihak berkuasa/organisasi berkenaan dengan menyatakan sekali sebab dan tempoh laporan PSM ini

DECLARATION

I hereby, declared this report entitled Drive Right: Driving Simulation to Alert the Driver Using Artificial Intelligence for Prudential Driving Behaviour is the results of my own research except as cited in references.

Signature:

Author : Noor Anisah Binti A Sulaiman

Date:

APPROVAL

This report is submitted to the Faculty of Mechanical and Manufacturing Engineering Technology of Universiti Teknikal Malaysia Melaka (UTeM) as a partial fulfilment of the requirements for the degree of Bachelor of Mechanical Engineering Technology (Computer Systems) with Honours. The member of the supervisory is as follow:

Signature:

Supervisor : Dr Jamil Abedalrahim Jamil Alsayaydeh

Signature:

Co-supervisor: Dr Adam Wong Yoon Khang

ABSTRAK

Dalam laporan ini, siasatan dilakukan untuk mengenal pasti dan menganalisis kelakuan pemandu di jalan raya. Siasatan yang mendalam dilakukan untuk mengenal pasti bagaimana Kecerdasan Buatan boleh digunakan sebagai penyelesaian untuk mengatasi atau mengelakkan kemalangan yang disebabkan oleh kelakuan berhemat pemandu di jalan raya. Simulator pemandu ini digunakan untuk mencipta tingkah laku yang buruk. Dari senario pra-kemalangan NHTSA, 3 senario dipilih. Semua senario ini akan disimulasikan untuk melihat output dan kesan. Rangkaian saraf bertindak sebagai pengelas untuk senario. Fungsi TensorFlow yang merupakan model pengesanan objek API.A prototaip peranti dibangunkan untuk mengenali senario pra-kemalangan dan akan memberi amaran kepada pemandu untuk mengelakkan. Model TensorFlow akan dilatih untuk melaksanakan senario. Prototaip boleh membantu pemandu memandu dengan selamat. Model yang berbeza akan digunakan dalam pengesanan objek, Prestasi prototaip boleh diuji dengan pemandu di jalan raya. Dari pengujian keputusan dapat disimpulkan kinerja setiap model dan prototipe terakhir.

ABSTRACT

An investigation is conducted in this report to identify and analyze driver behavior on the road. A more in-depth investigation is carried out to identify how to implement Artificial Intelligence as a solution to resolve or avoid collisions induced by drivers' imprudential actions on the road. The bad behavioral was recreated by this driver simulator. Three scenarios were chosen from NHTSA's pre-crash scenarios and observe the performance and effects after all these scenarios be simulated. The neural network functions acts as a scenario classifier. TensorFlow's feature, which is object detection model API. A prototype unit created to recognize pre-crash scenarios and warn the driver to avoid them. TensorFlow models for the implementation of the scenarios to be received training. The model will help drivers to be alert and safely ride on the road. Different models will be used to detect objects design output can be evaluated on the road with driver. The performance of each model and the final prototype can be concluded from the test results.

DEDICATION

To my beloved parents, thanks for the help, concern and understanding while I'm in developing this project.

ACKNOWLEDGEMENTS

I would like to thanks my supervisor, Dr Jamil Abedalrahim Jamil Alsayaydeh for his guiding during my final year project period so that I will be able to done through the Projek Sarjana I subject successfully. I also would like to thanks my co-supervisor, Dr Adam Wong Yoon Khang for his concern and care about my project during my final year project period. Besides that, I would like to give my appreciation to my final year project panels, Sir Aiman Zakwan Bin Jidin and Madam Dayanasari Binti Abdul Hadi for giving their time to observe my presentation and evaluate my work. I want to thanks them because giving a recommendation and advices in this project.

In addition, I would like to thanks to my beloved parents. My father Mr. A Sulaiman and my mother, Mrs. Sarifah for their understanding and giving support during my hardship. Thanks also to them for their patience to understand me.

Lastly, I would like to thanks my beloved friends at Universiti Teknikal Malaysia Melaka who giving me support, advices, and help me when I need them throughout my project.

TABLE OF CONTENTS

	PAGE
TABLE OF CONTENTS	x
LIST OF TABLES	xvi
LIST OF FIGURES	xvii
LIST OF APPENDICES	xxi
LIST OF ABBREVIATIONS	xxii
CHAPTER 1 INTRODUCTION	1
1.1 Background	1
1.2 Problem Statement	2
1.3 Objectives	4
1.4 Project Scopes	5
1.5 Report Significant	6
1.6 Report Structure	6
CHAPTER 2 LITERATURE REVIEW	8
2.1 Introduction	8
2.2 Driver Assistance System Background	8
2.2.1.1 Drowsiness Detection with OpenCV	9
2.2.1.1. Strengths	11

2.2.1.2.	Weaknesses/Limitations	11
2.2.1.2	Autonomous Object Detection and Tracking Using Raspberry Pi	12
2.2.2.1	Strengths	16
2.2.2.2	Weaknesses/Limitations	16
2.2.1.3	Driver Assistance System for Lane Detection	17
2.2.3.1	Strengths	21
2.2.3.2	Weaknesses/Limitations	22
2.3	Overcoming the Weaknesses	22
2.4	Hardware Overview of the System	23
2.4.1.1	Raspberry Pi 3 B+	23
2.4.1.2	Camera Module	25
2.4.1.3	CARLA Simulator	26
2.4.1.4	TensorFlow	29
2.5	Pre-Crash Scenarios	30
2.6	Comparison Between Existing System And Update System Developed	33
CHAPTER 3	METHODOLOGY	34
3.1	Introduction	34
3.2	Project Work Flow	34
3.2.1.1	Project Methodology	35
3.2.1.2	Research and Data Collection	36

3.3	Software and Hardware Requirements	36
3.3.1.1	Software requirements.	37
3.3.2.1	Anaconda3 1.9.7	37
3.3.2.2	Python 3.6	38
3.3.2.3	TensorFlow 1.9	38
3.3.2.4	CARLA Simulator	38
3.3.2.7	VNC Viewer 6.19	39
3.3.2.8	Microsoft Office 2016	39
3.3.1.2	Hardware requirements.	40
3.3.2.1	Raspberry Pi 3 Model B+	40
3.3.2.2	Webcam	41
3.3.2.3	MicroSD Card	41
3.3.2.4	Speaker	42
3.3.2.5	Car Mount	42
3.3.2.6	Workstation	42
3.4	Project Gantt Chart	43
3.5	Project Design	45
3.6	System Architecture	45
3.6.1.1	Simulation	48
3.7.2.1	Run CARLA Simulator	49

3.7	TensorFlow Model Architectures	52
	3.7.1.1 SSD (Single-Shot Detector)	52
	3.7.1.2 Faster RCNN	53
3.8	Implementation	54
3.9	Simulation	54
	3.9.1.1 Scenarios	54
3.10	Training Custom Object Detection	58
	3.10.1.1 Hardware Requirements.	58
	3.10.1.2 Hardware Requirements.	59
	3.11.2.1 TensorFlow 1.9	59
	3.11.2.2 LabelImg	60
	3.10.1.3 Training	61
	3.10.1.4 Run	66
	3.10.1.5 Implementing scenarios	68
	3.10.1.6 Prototype of DriveRight	70
	3.10.1.7 Conclusion	72
CHAPTER 4	TESTING AND DISSCUSSION	73
4.1	Introduction	73
4.2	Test Plan	73
	4.2.1.1 Test Environment	73

4.2.1.1	SSD Inception v2	75
4.2.1.2	Faster RCNN	77
4.2.1.3	SSD Mobilenet in Raspberry Pi	79
4.3	Comparison between models	82
4.4	Conclusion	83
CHAPTER 5	CONCLUSION	84
5.1	Introduction	84
5.2	Observation on Strengths and Limitations	84
5.2.1	System strengths	84
5.2.2	System Limitations	84
5.3	Proposition for Improvements	85
5.4	Contribution of the Project	85
5.5	Future Planning	85
5.6	Conclusion	86

REFERENCES	87
APPENDIX A	89
APPENDIX B	91
APPENDIX C	97

LIST OF TABLES

TABLE	TITLE	PAGE
Table 2.1:	The comparison between Raspberry Pi 3 B+ and B	24
Table 2.2:	Comparison among existing system and updated system developed.	33
Table 3.1:	Gantt Chart of Project Activities of BDP 1	43
Table 3.2:	Gantt Chart of Project Activities of BDP 2	44
Table 4.1:	Object detection model comparison.	82
Table :	List of pre-crash scenarios,	89

LIST OF FIGURES

FIGURE	TITLE	PAGE
Figure 2.1:	Calculate the ratio of eyes aspect.	9
Figure 2.2:	identifies the closed eyes for a sufficiently long time.	10
Figure 2.3:	Architecture of Object Detection	13
Figure 2.4:	Raspberry Pi 3 Model B	14
Figure 2.5:	IR Receiver	14
Figure 2.6:	ATemga32 Microcontroller	15
Figure 2.7:	Webcam	15
Figure 2.8:	Vehicle coordinate system	19
Figure 2.9:	(a) Unedited Image (b) Canny Edged map image	19
Figure 2.10:	(a)Main straight lines (b) Vanishing point detection	20
Figure 2.11:	a) Raw image b) Output of point extraction	21
Figure 2.12:	Image of Raspberry Pi with labels	23
Figure 2.13:	Camera module for Raspberry Pi 3	25
Figure 2.14:	CARLA Simulator. Adapted from CARLA Simulation Documentation	27
Figure 2.15:	3 Sensing Modalities. Adapter from CARLA Documentation	27
Figure 2.16:	Example of object detection on street using TensorFlow	30

Figure 3.1: Project Methodology.	35
Figure 3.2: Flowchart of project simulation	45
Figure 3.3: Custom Training Object Detection Flowchart	46
Figure 3.4: Prototype Flowchart	47
Figure 3.5: Prototype Diagram	48
Figure 3.6: CARLA Documentation	49
Figure 3.7: Activating carlasim environment	49
Figure 3.8: Installing required modules	50
Figure 3.9: CaleUE4 to start the server.	50
Figure 3.10: 80 NPC cars added to the map.	51
Figure 3.11: Running manual_control.py	51
Figure 3.12: pygame window and can drive the car with WASD keys	52
Figure 3.13: SSD Architectures.	52
Figure 3.14: Faster RCNN Architectures.	53
Figure 3.15: LabelImg annotating Images.	55
Figure 3.16: Colors of lights in traffic light.	56
Figure 3.17: Two-seconds rule before the tree pass	57
Figure 3.18: Two-seconds rule after the tree passes.	57
Figure 3.19: Creating environment.	59
Figure 3.20: Activate and install tensorflow version 1.9.	59
Figure 3.21: Installing all the prerequisites.	60

Figure 3.22: Folders inside TensorFlow folder	60
Figure 3.23: Installing dependencies and compiling package	60
Figure 3.24: Folders	61
Figure 3.25: Folders inside training_demo	61
Figure 3.26: Terminal command to navigate to folder and run the LabelImg.	62
Figure 3.27: LabelImg	62
Figure 3.28: Converting .xml to .csv file	63
Figure 3.29: Converting .cxv to record file	63
Figure 3.30: ssd_inception_v2_coco.config file from Object detection folder.	64
Figure 3.31: Contents from extracted file	64
Figure 3.32: config file filling the paths for test and training .record files.	65
Figure 3.33: Training in progress.	65
Figure 3.34: Graph of total losses decreasing by steps.	66
Figure 3.35: Files exported	66
Figure 3.36: Running object detection with custom trained model.	67
Figure 3.37: Route of test run the object detection.	68
Figure 3.38: Code snippet for the Stop Sign alert.	69
Figure 3.39: Detection of Stop Sign and alert text.	69
Figure 3.40: Code snippet for determining the distance.	70
Figure 3.41: Condition for just warn if the vehicle is in center.	70
Figure 3.42: Code snippet for warning.	71

Figure 3.43: Warning and sound alert is working in RasPi.	71
Figure 3.44: Prototype attached to webcam.	72
Figure 4.1: Contingency Table. Adapted from ResearchGate.	74
Figure 4.2: Recorded Video Footage	74
Figure 4.3: Lossless graph of SSD Inception v2	75
Figure 4.4(a) & Figure 4.3 (b)	76
Figure 4.5: Live feed camera detection	77
Figure 4.6: Lossless graph of Faster RCNN model.	77
Figure 4.7: Object detection running Faster RCNN model.	79
Figure 4.8: Prototype mounted	80
Figure 4.9: Side view of the prototype.	80
Figure 4.10: Detecting the car is about to turn.	81
Figure 4.11: Warning and the beep alerted	81

LIST OF APPENDICES

APPENDIX	TITLE	PAGE
Appendix A	List of Pre-crash scenarios	89-90
Appendix B	Tensorflow installation	91-96
Appendix C	Python script for Object Detection	97-98

LIST OF ABBREVIATIONS

PCA	Principal Component Analysis
FYP	Final Year Project
LDAS	Lane Departure Assistance Systems
NHTSA	National Highway Traffic Safety Administration
MIROS	Malaysian Institute of Road Safety Research
ADAS	Advanced Driver Assist Systems
PCAM	Pedestrian Crash Avoidance/Mitigation (US)
DAS	Driver Assistance System
FPS	Frame Per Second
ITS	Intelligent Transportation System

CHAPTER 1

INTRODUCTION

1.1 Background

Over the past 30 years, autonomous vehicle growth has made quick progress, but major challenges persist, including the ability to review that systems are safe and robust to perform. Nonetheless, semi-autonomous security systems can already gain from this progress today. A good example is the Lane Departure Assistance Systems (LDAS), which already has great potential on their own to improve traffic safety. Many major car manufacturers currently offer several types of lane keeping aid. The spectrum varies from pure collision avoidance systems to lane servicing systems that actively maintain the car in the middle of the lane. D Hoehener (2016). These are sight-based systems which use the vehicle's place in the lane, its heading and a limited look-forward horizon (focused on the capability of the sensor) to ascertain whether alert or steering input is needed.

Techniques can be found primarily in the autonomous vehicles literature to make sure the vehicle continues to remain in a provided line of traffic. As they are constantly trying to keep the automobile in the middle of the lane, this very safety equipment is commonly referred to as lane support structures. One way to accomplish this objective would be to consider the task of getting the lane as a stabilization challenge where one attempts.

Advanced driver assists systems (ADAS) also support the person who drives the car during the travelling. They need to increase the safety features in vehicle and specifically in traffic safety if it is engineered in such a secure human-machine operating

system. Major road accidents were caused by faulty. Advanced driver support technologies are automation, adaptation and improvement systems for car safety and driving. By reducing human error, the automatic system produced by ADAS to the automobile have been shown to minimize traffic fatalities. Security features are aimed at preventing crashes and fatalities by offering systems that warn the driver to possible issues, or by applying protections and assuming vehicle control to prevent injuries and deaths.

Optimized functionalities can optimize lighting, include automatic braking and lane departure warning, minimize pedestrian crash avoidance/mitigation (PCAM), integrate navigation and traffic alerts, link up to gadgets, notify drivers whenever the potential risk is high, warning road alert system for departure, centering the vehicle on the lane automatically and view objects or people in the peripheral vision. The latest vehicles have developed hardware compatibility system such as digital stability control, warning given when the driver moves to another the lane marking, parking sensors and brake assist. Mechanical alignment adjustments can affect these systems. This kind of systems can be influenced by retailers to allow digital restarts for such technologies, ensuring the tyre aligner to meet these regulatory standards after a mechanical calibration is performed.

1.2 Problem Statement

According to MIROS, percentage of traffic accidents are occurred from human error are 80%. In addition, the loss of 65,850 individuals in road crashes between 2004