JOURNAL OF MECHANICAL ENGINEERING AND SCIENCES
ISSN: 2289-4659 e-ISSN: 2231-8380
VOLUME 17, ISSUE 2, 2023, 9520 – 9541
DOI: https://doi.org/10.15282/jmes.17.2.2023.10.0754

اونيۏرسيتي مليسيا ڤهڠ
UNIVERSITI MALAYSIA PAHANG

RESEARCH ARTICLE

Validation of scenario-based virtual safety testing using low-cost sensor-based instrumented vehicle

J.H. Cheok[1]*, K.O. Lee[1], V. R. Aparow[1], N. H. Amer[2], C.S.P. Peter[3] and K. Magaswaran[3]

[1] Automated Vehicle Engineering System (AVES) Research Group, Department of Electrical and Electronics Engineering, University of Nottingham, Malaysia, Jalan Broga, 43500, Semenyih, Selangor, Malaysia
Phone: +6 (03) 8924 8000; Fax: +6 (03) 8924 8001
[2] Department of Mechanical Engineering, Faculty of Engineering, Universiti Pertahanan Nasional Malaysia, 53000, Sungai Besi, Kuala Lumpur, Malaysia
[3] International Innovation Hub, Centers of Excellence, Technology Commercialisation Accelerator, Malaysian Research Accelerator for Technology & Innovation (MRANTI), 43300 Kuala Lumpur, Malaysia

ABSTRACT - Autonomous vehicle (AV) requires millions of miles on road to test the reliability of safety systems. It is also difficult to test the AV for critical scenarios which are rare but will endanger road users. Therefore, virtual safety testing simulation platforms are introduced to test the safety systems of the autonomous vehicles in critical scenarios. However, developing the virtual safety testing simulation platform requires information about the environment and driving data from the real world. Besides, it is challenging to build a system to collect driving data which is normally cost intensive especially in developing countries. Paradoxically, these developing countries have poor traffic environment which can provide valuable scenarios for safety testing test cases. Therefore, in this paper, a scenario-based testing using virtual simulation platform is developed using data captured by a low-cost sensor-based instrumented vehicle. The instrumented vehicle is built by low-cost off-the-shelf components for the testing purpose. The instrumented vehicle is used for validation process in IPG CarMaker's vehicle model using SAE standards. Then, the validated vehicle model is used as an autonomous vehicle in IPG CarMaker for the virtual scenario-based safety testing. The whole validation process from data collection to data logging is carried out using various economic sensors instead of a single industrial system. This approach greatly reduce the cost of the instrumented vehicle and the result of the scenario-based testing shows that the virtual scenarios developed in IPG CarMaker can be used for validation purpose with actual scenarios using low-cost sensor based instrumented vehicle as low as 4% root mean square percentage error.

ARTICLE HISTORY

Received	:	20th Dec. 2022
Revised	:	01st June 2023
Accepted	:	13th June 2023
Published	:	28th June 2023

KEYWORDS
Autonomous vehicle
Low-cost sensor
IPG CarMaker
Safety testing
Scenarios

1.0 INTRODUCTION

The development of autonomous vehicles has been growing exponentially in the past few years. However, safety concerns for autonomous vehicles becomes one of the major challenges as government policy tends to prohibit the road testing of autonomous vehicles in the public road as it is hard to justify the accident cases with the road user [1] . Therefore, simulation of autonomous vehicles using virtual environments has been carried out on a large scale in the research community with the aim to facilitate an environment as close as possible to the real world [2]. There have been many virtual driving simulators such as IPG Carmaker, Carla [3], Baidu Apollo [4], LGSV simulator [5], CarSim and VTD Vires, which is used for the safety testing of autonomous vehicles. However, in this paper, IPG carmaker is used for the scenario- based testing as it not only contains a large amount of vehicle dataset but also having a built-in environment editor for creation of scenario, test case manager as well as traffic modelling. Using a scenario-based testing approach [6], any real- world driving condition can be recreated, and thus, endless testing can be carried out which can reduce any risk of life or damage to a vehicle. Recently, there are several publications that are using the generation of scenario in virtual environments such as using homologation process [6], extraction from real driving data.

To ensure the performance of the scenario-based testing is reliable, the vehicle used within the assessment process plays a key role. This is mainly because an unvalidated vehicle may exhibit a different behaviour which led to the variation in result. Therefore, the vehicle model must be validated with the responses from a real vehicle before it can be used with the scenario [7]. An instrumented vehicle is developed in this study, and the vehicle is used to collect the data from the assessment to support model validation process. The validation process comprises of several test cases as specified under the ISO standard [8]. Currently within the research community, an instrumented vehicle that is used for data collection are having complicated set up which not only require high power but also expensive. This paper proposed a low-cost configuration that is also effective for measuring the dynamic performance of a vehicle as well as collecting visual data for further data processing.

***CORRESPONDING AUTHOR | J. H. Cheok** | ✉ kecy6cjh@nottingham.edu.my

In order to design virtual scenarios for scenario-based testing, driving data in real-world is needed. There are various AV datasets publicly available for research. The most popular is the KITTI dataset that has been widely used as benchmark for development of AV. In recent years, there are many automotive manouevering datasets developed such as the ApolloScape, Waymo, nuScenes, ONCE, RADIATE and Boreas due to the increase gain of research interest in autonomous vehicle field as shown in Table 1. Most of the dataset provides front camera view data and have sensor configuration of GPS, IMU and LIDAR. Some dataset however, provide rear camera view, Radar sensor and driving input data. Waymo is a new dataset which is larger than KITTI and has more data diversity such as night and rain condition. However, Waymo dataset only provides front camera view and lack of rear or surrounding camera. ApolloScape, nuScenes and ONCE are some of the datasets that provide Asian road environment. Nevertheless, the ApolloScape dataset does not consist of driving input data and it still lacks driving data during rainy weather which is important especially for tropical ASEAN countries. Similar to ApolloScape, ONCE dataset is captured in Asian road environment and provides both front and rear camera view data but ONCE has more data diversity as it consists of data captured in raining weather. However, ONCE still lacks driving input data such as the throttle, braking and steering angle information. nuScenes dataset is the most comprehensive dataset including every important element for automotive research from low level to high level autonomous vehicle. RADIATE focused on Radar sensing solution instead of optical sensors so the dataset only consists of front camera data and the dataset also lacks driving input data. Similar to RADIATE, Boreas dataset is one of the latest automotive datasets available and has a large diversity of data including seasonal changes and weather such as snow falling, rain and sun. Yet, it lacks the rear camera view and driving input data.

There is also no existing dataset that focuses on developing countries. Besides, these datasets used high end sensors on their instrumented vehicle such as Velodyne HDL-64e LIDAR used in KITTI dataset, Navtech Radar used in RADIATE, and Velodyne Alpha-Prime (128-beam) LIDAR used in Boreas dataset. These sensors can provide high accuracy and high-quality data, but the cost of these sensors is quite prohibitive for a developing country’s economy which makes it difficult to implement existing data collection framework in developing countries such as ASEAN countries. In order to develop a dataset that is suitable for these developing countries, there is a need for a low-cost data collection framework that can meet reliability and performance requirements. From literature search, existing automotive dataset as shown in Table 1 were collected using expensive sensors and there is no existing platform which cost less than $USD1000. Therefore, this is the prime motivation for this research study.

Table 1. Comparison between existing autonomous vehicle dataset

Dataset	Year	Camera View		RGB solution	Sensor Configuration	Driving Input		Diversity		Location
		Front	Rear			Steering	Gas & Brake	Night	Rain	
KITTI [9]	2013	Yes	No	1392x512	GPS+IMU+Lidar	No	No	No	No	Germany
ApolloScape [10]	2018	Yes	Yes	3384x2710	GPS+IMU+Lidar	No	No	Yes	No	China
Waymo Open [11]	2020	Yes	No	1920x1080	GPS+IMU+Lidar	Yes	Yes	Yes	Yes	USA
nuScene [12]	2020	Yes	Yes	1600x900	GPS+IMU+Lidar+Radar	Yes	Yes	Yes	Yes	Boston & Singapore
ONCE [13]	2021	Yes	Yes	1920x1080	GPS+IMU+Lidar	No	No	Yes	Yes	China
RADIATE [14]	2021	Yes	No	672x376	GPS+IMU+Lidar+Radar	No	No	Yes	Yes	UK
Boreas [15]	2022	Yes	No	2448x2048	GPS+IMU+Lidar+Radar	No	No	Yes	Yes	Canada

In order to develop a low-cost instrumented vehicle for data collection, it is important to understand the type of data that are required for an autonomous vehicle dataset. Camera is the primary sensor that is usually adapted for automotive data collection. It can be expected that by the next decade, autonomous vehicles will be launched to the market. In order to ensure user acceptance, these vehicles must not only be safe and reliable, but also provide a comfortable and safe driver experience.

Although comfort is a subjective task and highly influenced by driving style, some studies have shown that the users would prefer that an autonomous vehicle drives similar to their own driving style [16]. Recent studies such as [17] and [18] developed human driver-like manoeuvring to provide a personalized driving experience to users to increase user acceptance of AV. Therefore, driving inputs such as steering angle and pedal inputs are important elements for an AV dataset for personalized driving style profile. To capture driving inputs, most existing frameworks obtain steering angle and pedal inputs from CAN bus[19]. However, CAN bus data requires access to the vehicle’s CAN bus which is not suitable for most vehicle since there could be Intellectual Property (IP) protection issue. Whereas study from [20] shows sensors on systems that can be added to existing steering wheel to measure the steering input applied by the driver. Other research demonstrated steering wheel angle estimation using Inertial Motion Unit (IMU) data as well as IMU on consumer portable devices such as tablets and smartphones [21]. Besides, there is also study that utilize camera to capture the steering wheel rotation and obtain the steering wheel angle using optical flow-based estimation [22]. For gas and braking pedal inputs, research from [23] employs specially designed load cell that can be fixed to the braking pedal to measure the braking force applied by the driver. Furthermore, IMU is important to obtain the lateral, longitudinal accelerations and jerks of the vehicle to improve user comfort level. Finally, GPS is also an important sensor to obtain data for an AV in terms of navigation information and position of the vehicle.

Generally, it is expensive to set up an instrumented vehicle for data collection using the above-mentioned sensor. To obtain better reading from the sensor, industrial-grade sensor is normally being used. It also requires a specific target machine to work with instead of normal computing devices. There are different types of instrumented vehicle with wide varitey of sensors configuration. The use of high-fidelity instrumented vehicle is great for autonomous vehicle data collection. Zhao et al. [23] develop an instrumented vehicle to generate data on the vehicle trajectory. The instrumented vehicle contains a GPS/IMU setup, 4 2D LIDARs and also an omnivision system. Together with vehicle state estimation, the vehicle is able to generate the vehicle trajectory. Feraco et al. [24] equipped an instrumented vehicle with a Velodyne LIDAR sensor and a stereo camera for object detection. It will be able to detect multiple objects and map them together in an instant with the computing power of Nvidia Jetson AGX. The configuration of the instrumented vehicle above are not ergonomic and it usually needs a specific device to run them. Also, the configuration of the instrumented vehicle can be different based on the type of data needed to be collected.

In analysing the driving behaviour of different drivers, Agnoor et al. [25] used only GPS and IMU to record the vehicle movement under different conditions. In this occasion,the data collected by the instrumented vehicle are stored and used for recording the environment information, the driver input formation (throttling,braking and steering) and the parameter of the vehicle. Then, the validation process can be carried out with a vehicle model under different test. There is validation process carried out in the past but through industrial grade sensor which is costly. Therefore, an alternative validation procedure using a low-cost sensor on the instrumented vehicle is proposed in this paper. The instrumented vehicle will be able to capture real time data for processing while maintaining a high accuracy output.

In the same time, there are various kind of simulation testing carried out such as Li et al. [26] focusing on devloping a lane change framework by using deep reinforcement learning within CARLA simulator. Rempe et al. [27] developed an algorithm that can automatically generate scenario based on the traffic condition while Cheng et al. [28] provieded a review on using Simultaneous Localizaiton and Mapping (SLAM) to capture real world information.There are also other virtual driving platofrm which is VISTA [29] and MetaDrive [30] that are for specific driving task. Although the work mentioned above was able to utilize virtual scenario for algorithm testing or study of vehicle behaviour, but it still lacks real driving scenario. On top of that, the driving habit is relatively different on each country due to several factor such as road condition, traffic regulation, population area. To overcome this problem, this paper focusing on capturing real world scenario which is solely based on Malaysia environment and replicate it in a virtual world.

In this work, a low-cost data collection framework that consists of camera, steering angle sensor, pedal sensor, IMU and GPS is presented for capturing road environment of a developing country. Scenarios are then identified from the data collected which were used to be recreated in IPG CarMaker for autonomous vehicle scenario-based safety assessment. The remaining of the paper is organized as follows: Section 2 describes the development of the instrumented vehicle which includes the selection of sensor and the overview of the complete set up. Section 3 discusses the vehicle model validation using low-cost based instrumented vehicle. Section 4 summarizes the design of the scenario in IPG CarMaker which is taken from the data collection, followed by the explanation of the result.

2.0 DEVELOPMENT OF INSTRUMENTED VEHICLE SYSTEM

A data collection vehicle using low-cost sensors was designed for synchronously video recording process and other vehicle data as drivers operated the vehicle on public roadways. In this section, the instrumented vehicle sensor setup is described followed by the detail for data recording.

2.1 Sensor Setup

Various sensors were mounted on a Volkswagen Polo Sedan with a 1600 cc displacement engine and an automatic transmission, as shown in Figure 1. The vehicle has a weight of 1182 kg, wheelbase of 2552 mm, width of 1466 mm, hydraulic-powered rack and pinion steering, MacPherson Strut front suspension, torsion beam rear suspension. The vehicle's braking system uses a hydraulic system with ventilated disc on front axle and drum on rear axle and has Anti-Lock Braking system (ABS). The data acquisition system consists of two cameras (front and rear camera), a steering angle sensor, two pedal sensors (gas and brake pedal), an Inertial Measurement Unit (IMU), a Global Positioning System (GPS) and a laptop as HOST PC for synchronization of sensor data and data logging.

The camera sensors were installed on the front and rear of the vehicle to capture the surrounding traffic and road environment. The camera sensors were mounted on the windshield of the vehicle using suction cup windshield car mount. The camera sensors used are Raspberry Pi 8 Megapixels IMX219 camera from Sony with up to 30 fps when recording at full binned Field of View (FOV) resolution of 1640x1232 pixels. By default, the camera comes with 62.2 degrees of horizontal FOV lens which is too narrow for the usage of capturing surrounding environment. Therefore, the Pi camera lens is replaced with a wide-angle lens with horizontal FOV of 160 degrees so that more information can be captured by the camera. The cameras were connected to Raspberry Pi single board computer (SBC) through MIPI CSI connection.

However, the Raspberry Pi SBC only has a single MIPI CSI port on-board. So, in this case, two Raspberry Pi SBC are required for two cameras. Raspberry Pi Zero SBC is selected in this research for low-cost, low-power (can be powered from laptop's USB port) and has sufficient computation power for video recording task. The SBC encodes the captured raw data from the camera into motion jpeg video and stream the video to the laptop through USB communication.

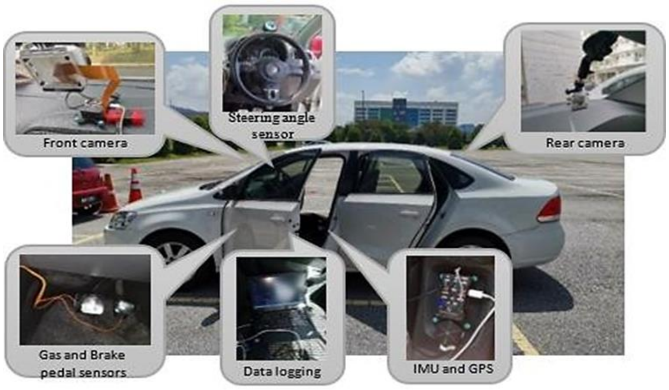
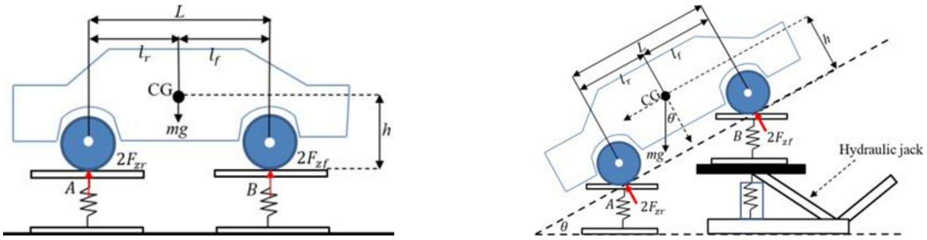


Figure 1. Sensor configuration of the instrumented vehicle

The steering wheel angle was measured using an optical encoder. The optical encoder used is RE30E from Nidec which provides up to 500 pulse per revolution which means that one pulse is equivalent to 0.72 degrees. To configure the optical encoder for angular position measurement of the steering wheel, an Arduino Uno microcontroller is used to obtain pulse signal from the sensor and convert the pulse signal into angular position measured. The angular position value measured is then transmitted to the laptop for data logging through USB serial communication. The gas and braking pedal forces applied by the driver was measured by using load cells that were fixed on the gas and braking pedal. In this research, the DYZ-105 sensor which is specially designed for this purpose was used to obtain the pedal force applied by the driver. The operating range of the pedal sensor is from 0 N to 2000 N. To read the output produced from the sensor, a load cell amplifier, HX711 is used to read the changes in the resistance of the load cell, amplify the analogue signal as well as acting as a 24-bit Analog-to-Digital Converter (ADC). An Arduino Uno is used as a bridge to connect the laptop to the HX711 through I2C interface. The data obtained is then transmitted to the laptop through serial communication. To obtain the attitude of the vehicle, a Pixhawk (PX4) flight controller which consists of ST Micro L3GD20 3 axis gyroscope, ST Micro LSM303D 3-axis accelerometer/magnetometer and Invensense MPU-6000 accelerometer/gyroscope.



(a) (b)

Figure 2. LD test procedure: (a) Vehicle placed on weighing scale and (b) Vehicle on weighing scale with inclination

To obtain the desired vehicle parameter from the IMU, the PX4 has to be placed on the Centre of Gravity (CoG) of the vehicle. The CoG location by the common LD (lifting detection) method [31]. The vehicle is placed on the weighing scale to measure the initial front and rear axle load. Then, a hydraulic jack is used to lift the front axle up to an inclination degree, θ. Once the value is obtained,the height of the CoG can be calculated using formula below:

$$h_0 = \frac{mg\ cos\ \theta\ lr\ -\ 2FzfL}{mg\ sin\ \theta} \tag{1}$$

In this setup, instead of installing the IMU on the actual CoG of the vehicle, it is attached on the dashboard. To compensate the offset, the effect of angular acceleration was considered from the reading of the IMU [32]. The CoG position offset, denoted as vector $p^{\rightarrow}$ can be obtained as follows:

$$\underset{a_c}{\rightarrow} = \underset{a_p}{\rightarrow} + \begin{vmatrix} -\omega_y^2 - \omega_z^2 & \omega_x\omega_y - \dot{\omega}_z & \omega_x\omega_z + \dot{\omega}_y \\ \omega_x\omega_y + \dot{\omega}_z & -\omega_x^2 - \omega_z^2 & \omega_y\omega_z + \dot{\omega}_x \\ \omega_x\omega_z - \dot{\omega}_y & \omega_y\omega_z + \dot{\omega}_x & -\omega_x^2 - \omega_y^2 \end{vmatrix} \underset{p}{\rightarrow} \tag{2}$$

Here, ω is the angular velocity in terms of x, y and z axes, while the vector ap is the linear acceleration recorded from the IMU on the offset position. With the transformation matrix multiplied with the offset position, the linear acceleration about the centre of gravity can be obtained. Besides, a U-blox M8N GPS is connected to the PX4 controller to access GPS data of the vehicle. The M8N GPS is a low-cost low-power GPS when compared to the centimetre-level GPS such as Real-Time Kinematic (RTK) or Trimble Real-Time eXtended (RTX) GPS which are very expensive. Nevertheless, the M8N is still able to deliver high precision and sub-metre accuracy of up to 0.6 meters.

A laptop namely Lenovo ThinkPad is used as the HOST PC for data recording. The laptop is powered by an Intel Core i5 10th gen processor with 8 GB of RAM. Since video data recording task requires high disk write throughput, a 1

TB SSD with write speed of 500 MB/s was used for data storage. For connectivity with the sensors, the laptop only supports up to two USB 3.0 Type A ports and one USB 3.1 Type C port while the sensor setup required five USB ports. Therefore, an external USB hub is used to provide additional USB ports required. The laptop also has a built-in 2.995 Ah/46 Wh battery that can be used to power the laptop, as well as sensors through USB port's 5V power line without any other external power supply. Despite having small sized battery, by using low powered laptop, Single Board Computer (SBCs) and Microcontroller Unit (MCUs) in the setup, the built-in battery was able to provide runtime of up to 3 hours in a single full charge which is sufficient for most data recording applications. Table 2 shows the power consumption measured during data recording. The connection of the sensors to the laptop for data recording is shown in Figure 3.

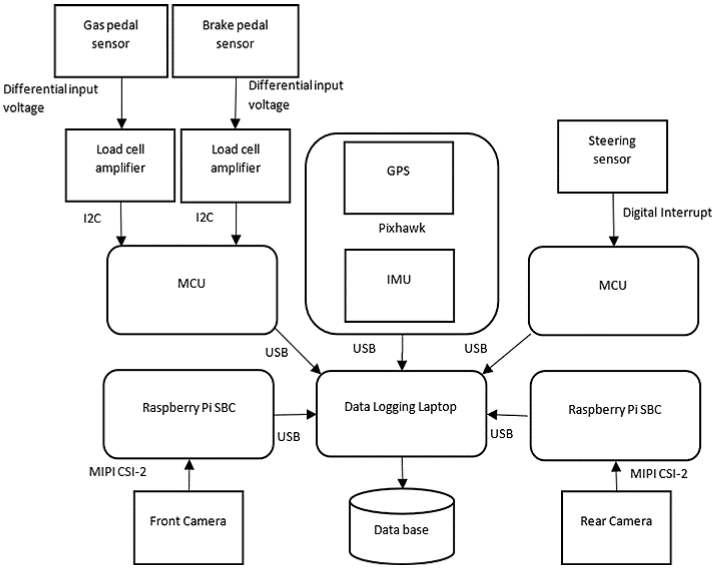


Figure 3. System architecture of data acquisition system

Table 2. Power consumption of data acquisition system measured during operation

Component	Voltage, V (Volt)	Maximum Current, A (Ampere)	Power Consumption, W (Watt)
Laptop	17.2	0.638	10.974
Raspberry Pi Zero + camera	5	0.385	1.925
Arduino Uno + 2x pedal sensor	5	0.058	0.290
Arduino Uno + steering sensor	5	0.062	0.310
Pixhawk + GPS	5	0.840	4.200

2.2 Sensor Calibration

Although some of the sensors being chosen can be used directly on the instrumented vehicle, some of the sensors requires callibration. The IMU sensor is critical for carrying out callibration to ensure global orientation is aligned with the orientation of the IMU. Multi position method which is a popular way of IMU callibration is being used for the process [32]. The IMU sensor is being placed in a different orientation while the reading is recorded according to the magnitude of gravity. The load cell sensor that is needed to measure the pedal force needed to go through callibration process. This is done by measuring the reading of the sensor during idle and maximum force conditions. The driver needed to fully press the pedal to obtain the maximum force reading. By obtaining both values, the maximum depth of the pedal can be calculated by mapping the output reading to the values.

2.3 Sensor Synchronization

Since every sensor mentioned in the previous section has a different sampling data rate as shown in Table 3, there's a need to synchronize these sensors. All the sensors were synchronized to the Network Time Protocol (NTP) time which is synchronized to the UTC time reported by the laptop. For each sensor, a buffer was created to hold the latest readings reported by the sensor. To update all the sensor's simultaneously, threaded processes were created for each sensor, enabling each sensor to update the reading at their sampling rate. Another threaded process which sampled at 100 Hz will receive all sensor data from the buffers and then record the sensor data and present timestamps in the database with file name with format of Sensordata-YYYY-MM-DD-HH-MM-SS.csv. Each data point recorded in the file has nine fields:

[t,lat,lon,pitch,roll,yaw,gas,brake,steer] where t is the timestamp, (lat, lon) is the latitude and longitude data obtained from the GPS, (pitch, roll, yaw) is the attitude data obtained from the PX4 IMU, (gas, brake) is the gas and braking force measured by the pedal sensors and steer is the steering angle obtained from the steering angle sensor.

Table 3. Output signal of sensors

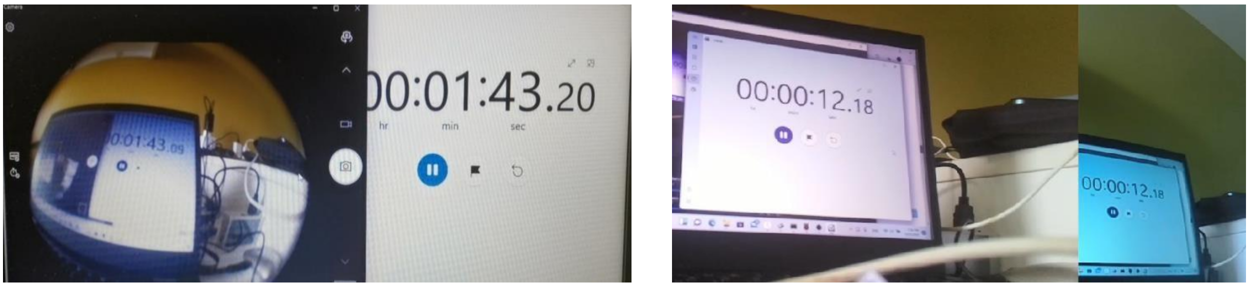
Sensor	Specification	Sampling Rate	Channel	Data Captured
Camera	Raspberry Pi 8MP Sony IMX219 Camera	30 fps	2	Front and rear view of road
Steering sensor	RE30E-500-213-1 optical encoder	100 Hz	1	Steering angle
Pedal sensors	DYZ-105 force sensor	80 Hz	2	Gas and brake pedal force applied
IMU	Pixhawk L3GD20 3-axis 16- bit gyroscope, LSM303D 3- axis 14-bit accelerometer /magnetometer and MPU6000 3-axis accelerometer /gyroscope	25 Hz	1	Linear acceleration and angular velocity in x-y-z axis
GPS	U-BLOX NEO-M8N GPS with compass	10 Hz	1	Lateral and longitudinal position, and heading

For camera video recording, free and open-source software, Open Broadcaster Software (OBS) was used. An OBS plugin is added to the OBS software to add timestamp to each frame of the recorded footage. Besides, due to the camera raw data required video encoding process to be recorded as video file, there is a time delay between the real world and the recorded video. To measure the latency from the camera to the laptop, the camera is pointed to a laptop screen displaying a stopwatch and the difference in time between the stopwatch time and the time captured by the camera. As a result, the latency from the camera to the laptop is measured as 0.11 s as shown in Figure 4(a). Furthermore, the latency between the front and rear camera can be measured using the same method. As shown in Figure 4(b), it can be observed that the time captured by the two camera Is the same, therefore, the latency between the two camera is zero. The recorded data is separated into different folders and identified by the location, date and time at which they were recorded with the format of Location-YYYY-MM-DD-HH-MM. The data for each recording is organized as shown below:

Location-YYYY-MM-DD-HH-MM

o YYYY-MM-DD-HH-MM-SS.mkv

o Sensordata-YYYY-MM-DD-HH-MM-SS.csv



(a) (b)

Figure 4. (a) Time delay between captured frame and real world and (b) Time delay between front and rear camera

3.0 SECTION 3 VALIDATION USING LOW-COST BASED INSTRUMENTED VEHICLE

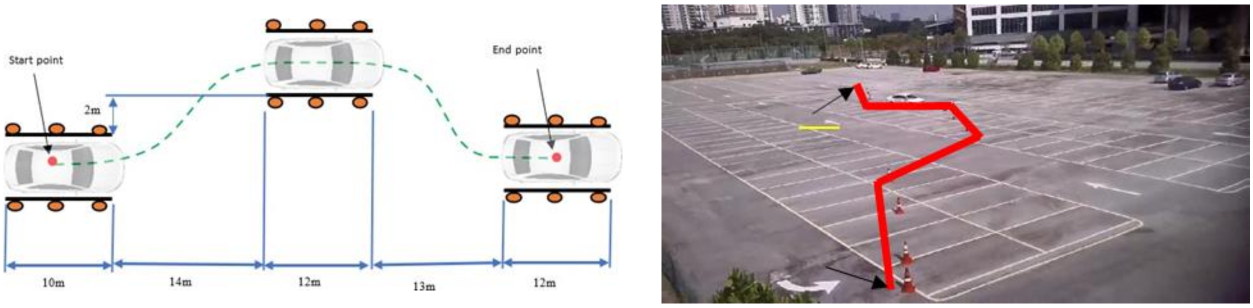
During the experiment, the driver performed field testing according to SAE vehicle safety testing standard for validation of the virtual vehicle. In order to validate IPG CarMaker's virtual vehicle using the data recorded from the instrumented vehicle, three types of field test scenario such as the double lane change test, step steer test and emergency braking test were carried out. All the tests were carried out at the parking lot of Asia Pacific University of Technology and Innovation (APU), which has a dimension of 100 x 60 m of asphalt road. Traffic pylons were used to define the test tracks according to the ISO 3888-2 SAE standard for double lane change test, step steer lateral test according to ISO 4138, and acceleration and brake longitudinal test according to ISO 3833 SAE standard. The tests are carried out starting at a constant speed of 20 km/h and increasing the speed of 10 km/h between experiments until 30 km/h. Figure 5 shows an outline of the SAE tests performed where there are three types of manoeuvre test for three test speeds. A total of 9 test cases were designed and each test case was repeated for three times to ensure the result is consistent, reproducible, and more reliable.

Speed 20km/h
Double lane change
Step Steer
Accelerate & Brake
Speed 30km/h
Double lane change
Step Steer
Accelerate & Brake

Figure 5. Test carried out according to SAE standards

3.1 Double Lane Change Test

In double lane change (DLC) test, the vehicle accelerates until the vehicle's speed reach a constant speed of 20 km/h and 30 km/h. Then, steering input is applied by the driver to the vehicle so that the vehicle passes through the route defined by the traffic pylons. The test track was setup according to the ISO 3888-2 standard for double lane change as shown in Figure 6(a). Figure 6(b) shows the instrumented vehicle carried out double lane change test. All sensor data that were collected throughout the test was recorded in storage to be analysed later.

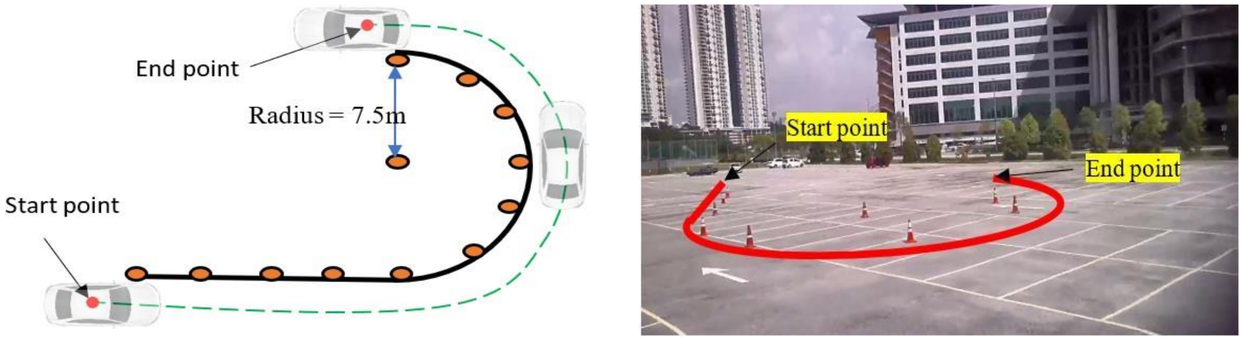


(a) (b)

Figure 6. (a) ISO 3888-2 test and (b) Double lane change field test

3.2 Step Steer Test

In the step steer test, the vehicle moves forward with a constant speed of 20 km/h and 30 km/h, when a constant steering angle is applied to the vehicle so that the vehicle turns left following the circular route defined by the traffic pylons. When the vehicle reached the exit, the constant steering angle applied is released so that the vehicle moves in a straight line. The test track was setup according to the SAE ISO 4138 standard for lateral test as shown in Figure 7(a). Figure 7(b) shows the instrumented vehicle carried out step steer manoeuvre. All sensor data collected throughout the test was recorded in storage to be analysed later.

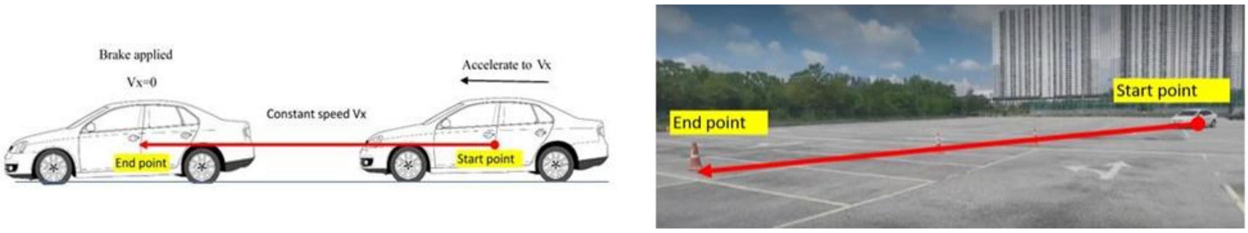


(a) (b)

Figure 7. (a) ISO 4138 test and (b) Step steer field test

3.3 Sudden Braking Test

The test begins with the vehicle in a stationary position. The driver provides sudden acceleration input by applying full throttle input. An emergency braking action will be given by the driver when the vehicle reaches the desired test speed to stop the vehicle. This test is conducted according to the ISO 3833 longitudinal vehicle test as shown in Figure 8(a). The gyroscope and the accelerometer inside the PX4 microcontroller are used in this test to record the longitudinal acceleration and the pitching angle of the vehicle for the validation test. The test was conducted as shown in Figure 8(b) using the instrumented vehicle.

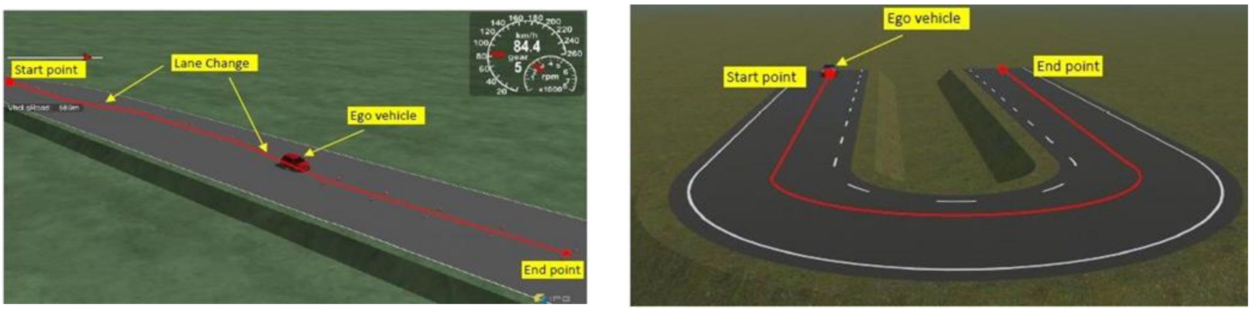


(a) (b)

Figure 8. (a) ISO 3883 test and (b) Accelerate and braking test

3.4 Model Validation using Instrumented Vehicle Data

In order to validate the simulated vehicle with data recorded from instrumented vehicle, the same test tracks used in the field test were created in the vehicle simulation platform[33]. IPG CarMaker, a simulation software commonly used by automotive industry and validated in is used to perform the lane change test. A Toyora camry model vehicle is used during the simulated test. The test tracks created are shown in Figure 9(a) for DLC test and Figure 9(b) for Step Steer test.

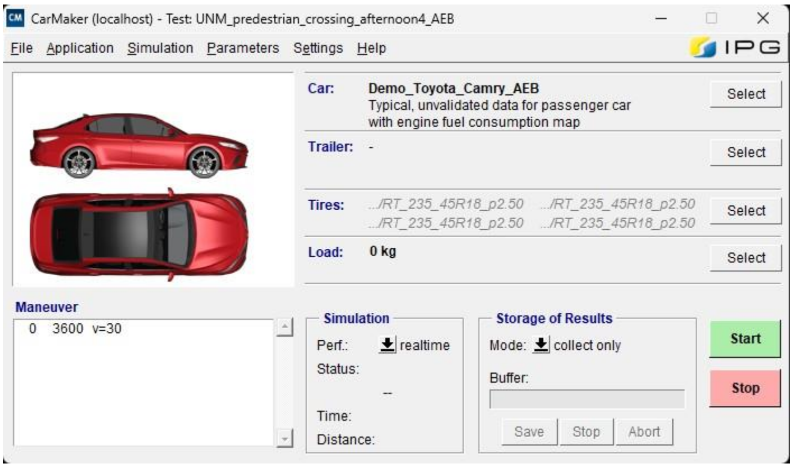


(a) (b)

Figure 9. (a) DLC test track and (b) Step steer test track developed using IPG carmaker

3.5 Vehicle Model Validation using SAE Standards

By providing the same human driving input such as steering angle, gas and brake inputs to the virtual vehicle in IPG CarMaker, the field tests are recreated in the virtual environment. The IPG CarMaker configuration window is shown in Figure 10. It should be noted that eventhough 3D model in IPG CarMaker is not exactly the same as the instrumented vehicle to be considered (Volkswagen Polo Sedan Vs. Toyota Camry), but the vehicle parameters used in the simulation model is studied and similar to the instrumented vehicle such as weight, width, length, center of gravity, vehicle suspensions and vehicle inertia. The vehicle states obtained from the virtual vehicle models are then compared against the sensor data obtained from the instrumented vehicle for double lane change test, step steer test, and acceleration and brake test.


CarMaker (localhost) - Test: UNM_predestrian_crossing_afternoon4_AEB

Car: Demo_Toyota_Camry_AEB
Typical, unvalidated data for passenger car
with engine fuel consumption map

Trailer: -

Tires: .../RT_235_45R18_p2.50 .../RT_235_45R18_p2.50
.../RT_235_45R18_p2.50 .../RT_235_45R18_p2.50

Load: 0 kg

Maneuver
0 3600 v=30

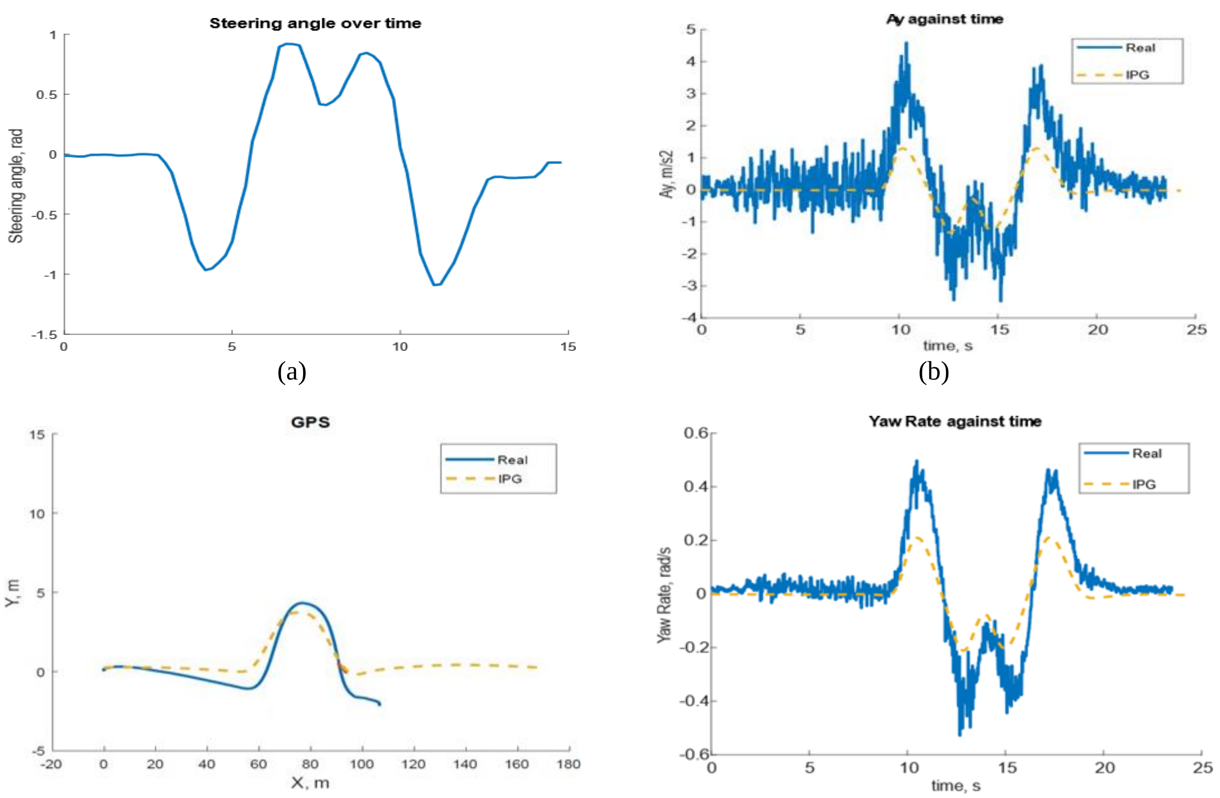
Simulation
Perf.: realtime
Status: --
Time:
Distance:

Storage of Results
Mode: collect only
Buffer:

Figure 10. CarMaker configuration window

3.6 Double Lane Change

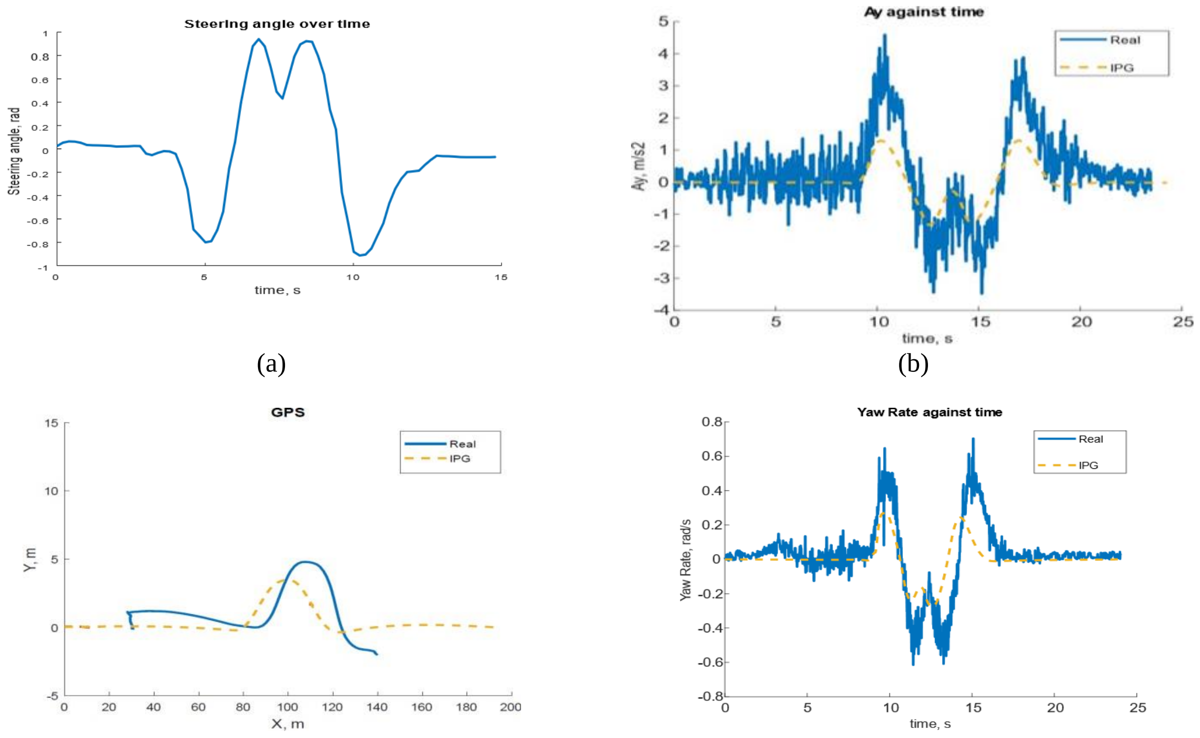
The result for double lane change at constant speed of 20 km/h is shown in Figure 11(a-d). Meanwhile, the result for double lane change at 30 km/h is shown in Figures 12(a-d) The responses of the vehicle model that are investigated are the lateral acceleration, local position and yaw rate of the vehicle.



(a) (b)

(c) (d)

Figure 11. (a) Steering angle, (b) Lateral acceleration, (c) Local position and (d) Yaw rate response for 20km/h



(a) (b)

(c) (d)

Figure 12. (a) Steering angle, (b) Lateral acceleration, (c) Local position and (d) Yaw rate response for 30 km/h

From the plot, it can be observed that the vehicle responses from virtual vehicle of the IPG CarMaker shared the similar characteristic with the vehicle state data obtained using the instrumented vehicle. Table 4 shows the percentage of RMS values for the overall responses from the sensor measurements when compared to the vehicle response from IPG CarMaker. Validation of the lateral acceleration, yaw rate and displacement of the vehicle moving at 20 km/h show the percentage difference of RMS errors about 7.23%, 5.46% and 2.25%, respectively.

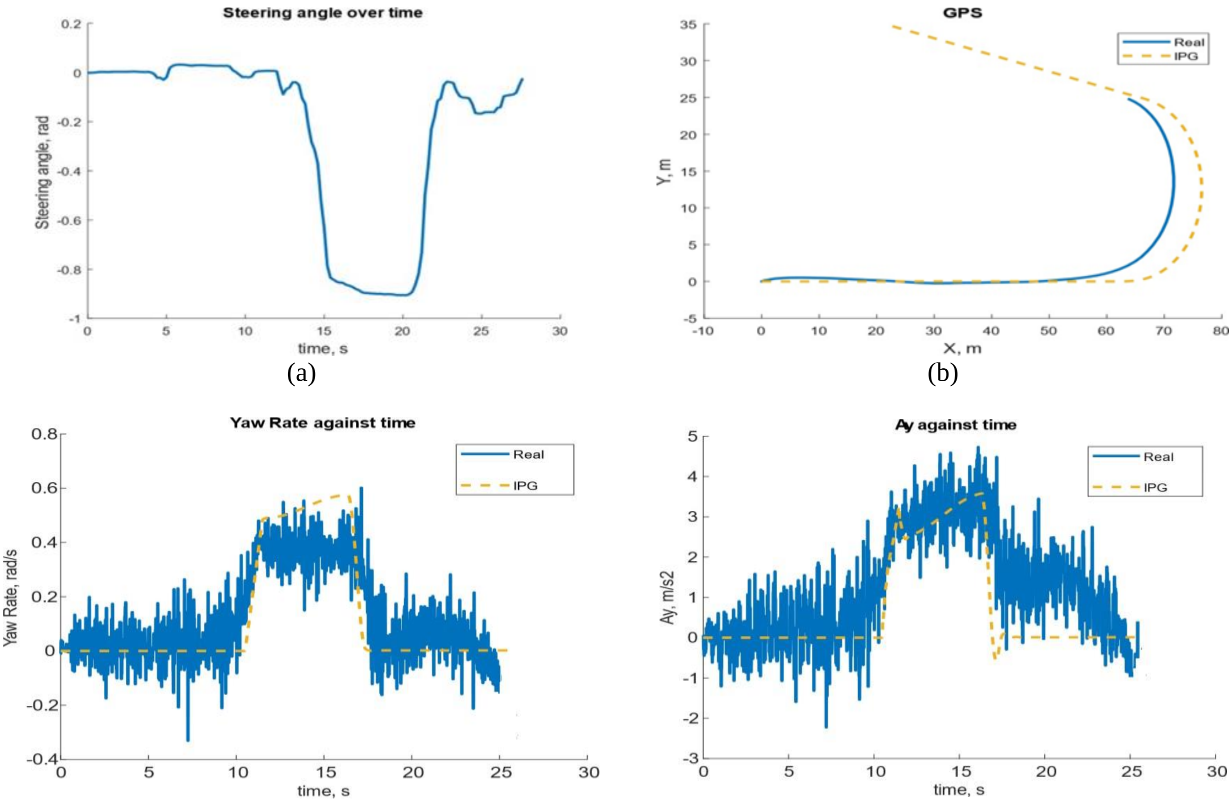
Table 4. Root mean square error for double lane change data

	Root mean square error	
20 km/h	Lateral acceleration, m/s^2	7.233%
	Yaw rate, rad/s	5.459%
	Position, m	2.245%
30 km/h	Lateral acceleration, m/s^2	7.494%
	Yaw rate, rad/s	5.336%
	Position, m	3.617%

Meanwhile, the lateral acceleration, yaw rate and displacement of the vehicle moving at 30 km/h show the percentage difference of RMS errors about 7.49%, 5.34% and 3.62% respectively. It can also be observed that the real vehicle is much more responsive and has a much obvious peak and low value. The reason is due to the suspension parameters such as spring constant and the damper coefficient are different between the two vehicles. Besides, the minor deviation observed from the signal of the experiment data is due to noise and vibration during the testing procedure. However, the root means square error (RMSE) calculated between the real model and IPG model for both 20km and 30km double lane change test are still under acceptable range.

3.7 Step Steer Test

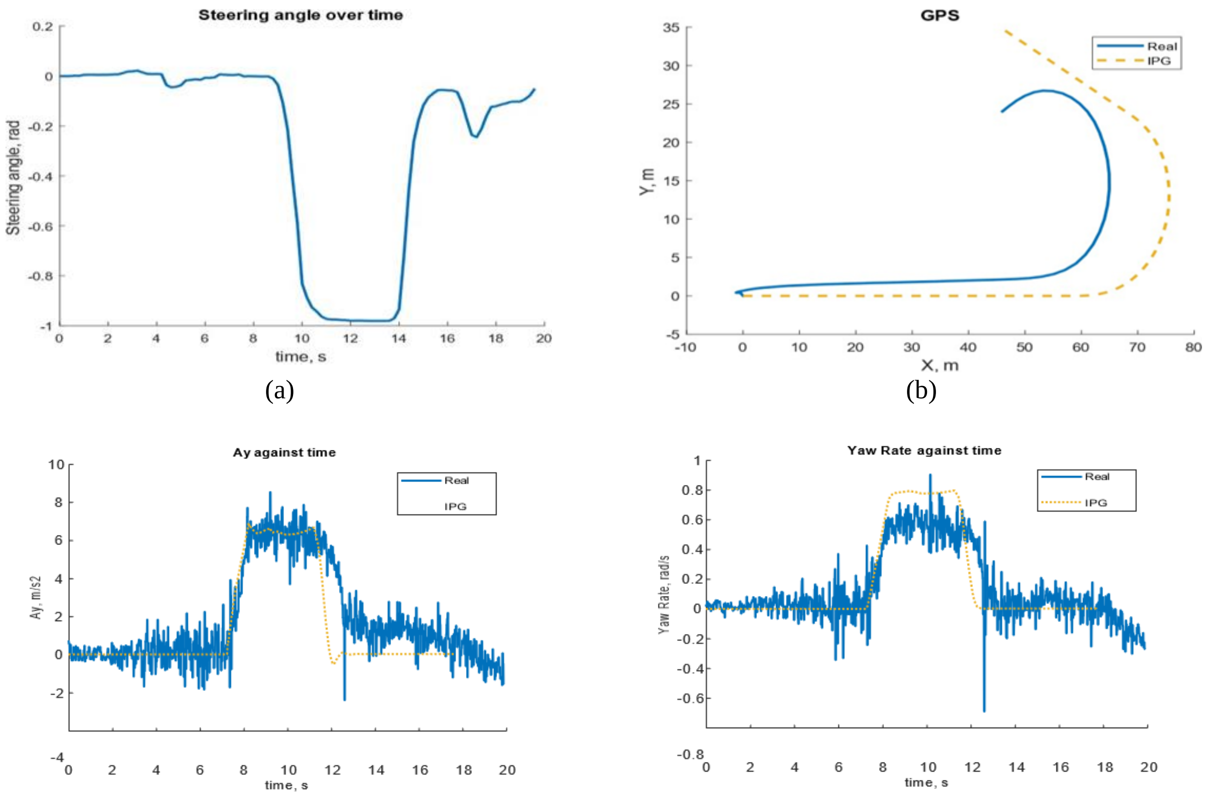
For step steer test, the results obtained at constant speed of 20 km/h are shown in Figures 13(a-d). Whereas the result for step steer test at 30 km/h are shown in Figures 14(a-d). Similarly, the response of the vehicle model that are investigated are the lateral acceleration, longitudinal acceleration, yaw rate and the displacement of the vehicle. From the plot, it can be observed that the response of the vehicle obtained from real vehicle shared similar characteristic with the virtual vehicle from IPG CarMaker. Table 5 shows the summary of RMS percentage error for step steer at 20km/h and 30km/h when compared to the data obtained from the IPG CarMaker's model. The lateral validation of acceleration, yaw rate and displacement of the vehicle moving at 20 km/h show the percentage difference of RMS errors about 3.617%, 7.156% and 11.13% respectively. Meanwhile for the lateral acceleration, yaw rate and displacement of the vehicle moving at 30 km/h show the percentage difference of RMS errors about 4.163%, 7.394% and 1.142% respectively.



(a) (b)

(c) (d)

Figure 13. (a) Steering angle, (b) Lateral acceleration, (c) Local position, (d) Yaw rate response for 20km/h



(a)

(b)

(c)

(d)

Figure 14. (a) Steering angle, (b) Lateral acceleration, (c) Local position and (d) Yaw rate response for 30 km/h

Table 5. Root mean square error for step steer test data

Root mean square error		
20 km/h	Lateral acceleration, m/s^2	3.617%
	Yaw rate, rad/s	7.156%
	Position, m	1.113%
30 km/h	Lateral acceleration, m/s^2	4.163%
	Yaw rate, rad/s	7.394%
	Position, m	1.142%

Similar to the double lane change tests, higher peak and lower low values for the lateral acceleration were measured from the real vehicle data compared to the simulated vehicles. This can be explained by the displacement plot of the real vehicle and the simulated vehicle as shown in Figures 14(a-d). From the displacement plot, it is observed that the radius of the circular motion of the real vehicle during step steer is smaller than the simulated vehicle. Since the steering angle input is similar for both vehicles, it can be deduced that simulated vehicles have more understeer characteristics than the real vehicle. Therefore, this caused the simulated vehicles to turn less than the real vehicle even though the steering angle input is the same. As a result, the yaw rate and lateral acceleration of the simulated vehicles are lower than that of the real data. However, the root means square error (RMSE) calculated between the real model and IPG model for both 20 km/h and 30 km/h step steer test are still within acceptable range. Hence, it is validated that the vehicle model developed is capable of producing the vehicle state of a real vehicle when performing step steer manoeuvre.

3.8 Sudden Braking Test

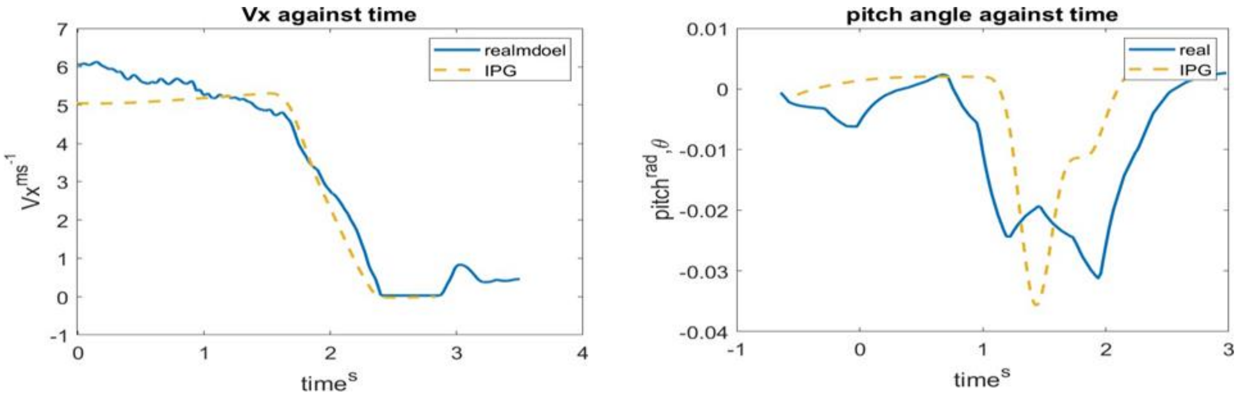
In this test, the vehicle will move at a constant speed and then applied with sudden braking input to come to halt the vehicle. This test is used to test the deceleration responses from the vehicle. The longitudinal speed, and the pitch angle will be used for comparison to show the similarity between the two models. The results obtained at constant speed of 20 km/h are shown in Figures 15(a) and 15(b), respectively. Meanwhile, the results for braking test at 30 km/h are shown in Figures 16(a) and 16(b), respectively. From the response from the plot, it is shown that during the acceleration phase of the vehicle, both models shared similar characteristics which proves that the engine dynamic model and the engine mapping graph are reliable. Table 6 shows the RMS percentage error of the longitudinal velocity and pitch angle of the vehicle moving at 20 km/h and 30 km/h which are about 6.9%, 12.2%, 9.02% and 10.4%, respectively.

While at the point of braking, the IPG CarMaker model had a slightly longer braking time to reach stationary than the real vehicle at both 20km/h and 30km/h test as shown in Figure 15(a) and Figure 16(a), respectively. This is due to the

the nonlinearity in the braking behaviour of the real vehicle, whereas factors such as thermal characteristics and brake piston are not considered in the development of virtual vehicle model. For pitching angle, both real and simulated vehicles shared similar patterns during braking. However, the real vehicle has more obvious pitch down value. The reason is due to the suspension parameter such as spring constant and the damper coefficient of the simulated vehicles are estimated and are not the same to the real vehicle. This is because most of these parameters are confidential data of the automotive company which will not be released to the public. For the IPG Carmaker vehicle model, it can be observed that the pitch angle value is higher than the real vehicle which is due to the same reason as mentioned above. The root means square error calculated between the real model and virtual model for both 20 km/h and 30 km/h test are still under acceptable range for the validation.

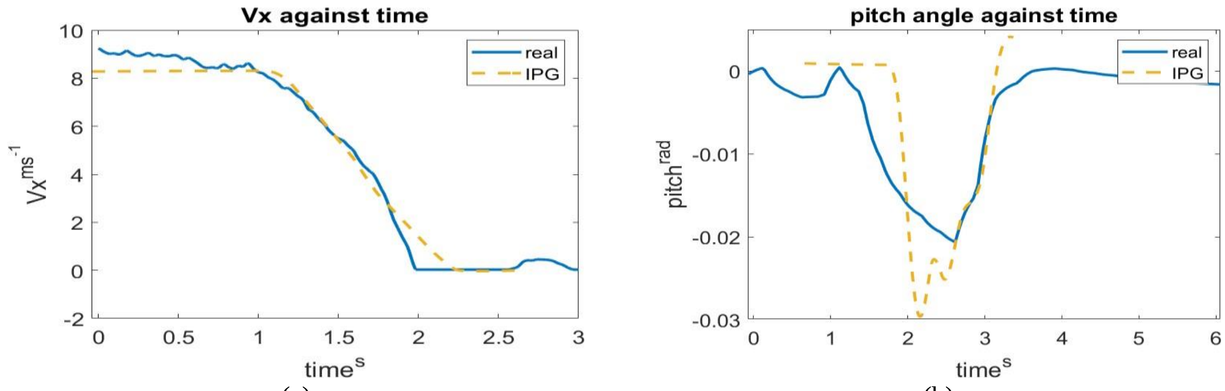
Table 6. Root mean square error for braking test

Root mean square error		
30 km/h	Velocity, *m/s*	6.89%
	Pitch angle, *rad*	12.21%
20 km/h	Velocity, *m/s*	9.02%
	Pitch angle, *rad*	10.41%



(a) (b)

Figure 15. (a) Longitudinal velocity and (b) Pitch angle against time at 20 km/h



(a) (b)

Figure 16. Longitudinal velocity and (b) Pitch angle against time at 30 km/h

3.9 Summary of Validation Test

From the analysis of the validation result, the validation carried out using low-cost sensor configuration instrumented vehicle is able to achieve RMS error under 15%. On top of that, it can be shown that some discrepancies are shown during the validation test and this can be considered mainly due to the mechanical friction between vehicle's internal components during the testing [34]. From this observation, it can be noted that using this sensor configuration, the validation test can achieve a high similarity during steady state. On the other hand, additional errors are re-tuned during the transient state (vehicle acceleration, sudden steering) and a possible reason is due to the low sampling rate of the IMU sensor. However, considering the low cost configuration, which is much cheaper, the overall performance of the configuration is acceptable as the RMS error value is still within the accepatable range.

4.0 DESIGNING SCENARIO-BASED TESTING

Once the validation testing is completed using the instrumented vehicle, the instrumented vehicle is used to collect driving data and traffic environment data on public road. The collected data is then analysed, and critical scenarios are extracted from the data recordings. In this work, the dataset was collected by driving the instrumented vehicle on a repeated route in University of Nottingham Malaysia for several weeks. This is because the campus route is highly populated with road users moving in, out and around the campus. The data collections were carried out in four different scenarios, i.e., morning, afternoon, evening, and rain. A total of 75 hours of driving data in University of Nottingham Malaysia were collected. The video recordings were fed to a deep learning neural network which is YOLOv5 for object detection. Based on the traffic object detection result, the critical scenarios occurring in the campus can be identified and extracted as presented in the preliminary work from [35]. Eight different road actors were found in the datasets including car, van, truck, bus, bicycle, motorcycle, pedestrian, road obstacle. Figure 17 shows examples of the scenes identified.

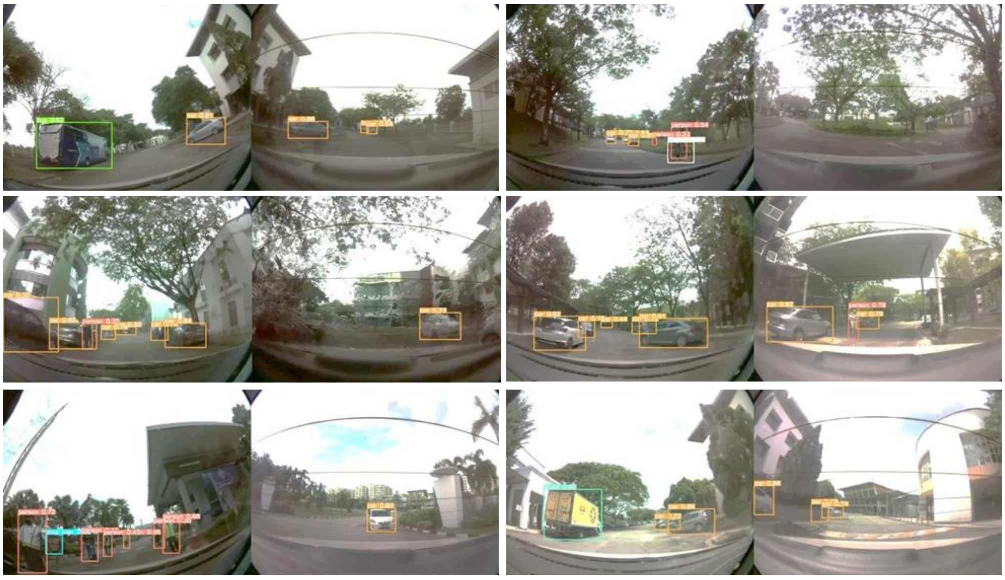
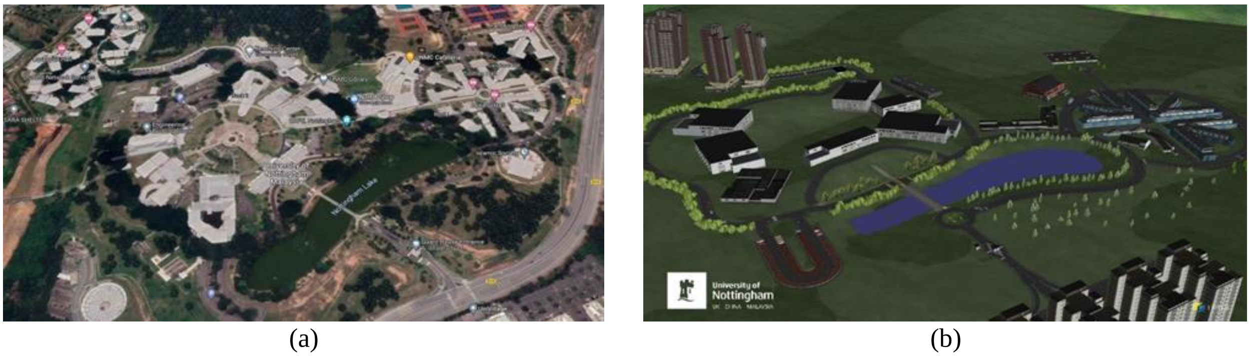


Figure 17. Examples of scenarios captured by the instrumented vehicle focusing on parked vehicles, cyclist and pedestrians along the road

With preliminary studies from [36], by referring to the satellite view of the campus as shown in Figure 18(a), a 3D virtual road model is developed using Scenario Editor tool in IPG CarMaker as shown in Figure 18(b). In this work, four different test scenarios have been selected from the classified dataset of Malaysian Traffic Scenarios framework. The first test case is the stop and go situation at a roundabout. The ego vehicle will follow a car in front around a roundabout which has a bump near the pedestrian bridge to slow down vehicles for pedestrians to cross the road safety. The second scenario is the overtaking scenario where there is a motorcycle stopped in front at the centre of the road and blocked the ego vehicle from moving forward. Therefore, an overtaking manoeuvre is needed to avoid the stationary motorcycle. The third scenario is the avoiding road obstacle scenario. A road obstacle constructed using traffic pylons is placed at the middle of the road. Therefore, the ego vehicle will need to apply manoeuvre to avoid the road obstacle. The fourth scenario is the road crossing pedestrians' scenario where a group of pedestrians is crossing the road and the driver need to slow down and stop ego vehicle for the pedestrians to cross the road. Virtual scenarios were developed based on the actual scenarios identified.



(a) (b)

Figure 18. (a) Satellite map of the Uni. of Nott. Malaysia and (b) 3D virtual map of the Uni. of Nott. Malaysia

4.1 Stop and Go-Roundabout

The first scenario is focusing on vehicle following scenario within a curved lane. This scenario emphasizes a roundabout scenario located after the main guard house of the university. For this scenario, the ego vehicle is driven at constant speed before decelerating to halt in the middle of the roundabout before cruising at the back of the blue vehicle as shown in Figure 19. Meanwhile, Figure 20 shows the comparison between the real world and virtual world using camera vision sensor installed in the instrumented vehicle. The scenario from actual environment is modelled using Scenario Editor in IPG CarMaker for simulation testing.

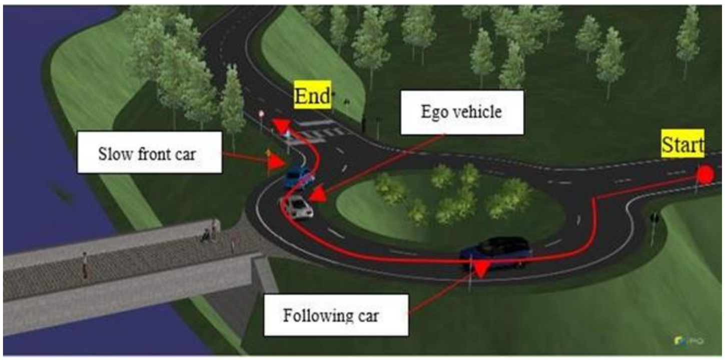


Figure 19. Roundabout scenario

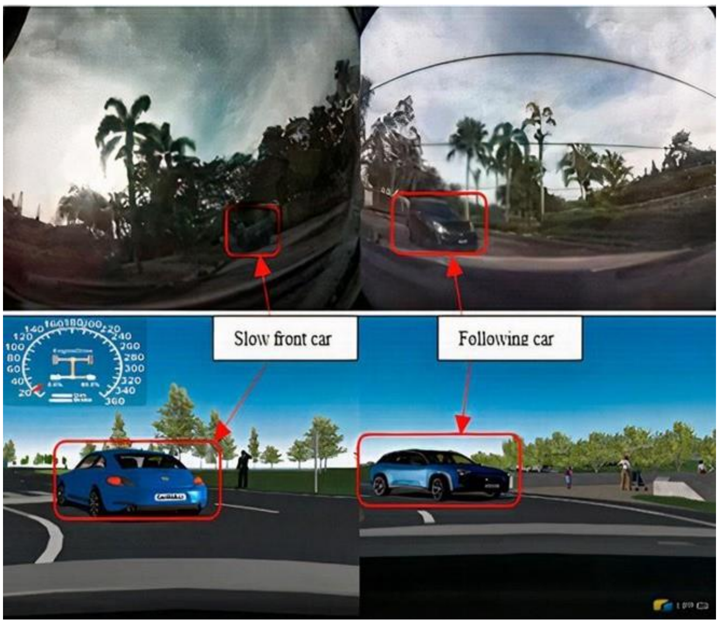


Figure 20. On-board camera comparison for roundabout scenario

4.2 Junction Overtaking

The second scenario will focus on the overtaking scenario at left-turning junction. In this scenario, the ego vehicle is needed to execute a left turning at a junction. However, a bike is in stationary condition at the front of the left turn junction. Therefore, the ego vehicle has to do an evasive turning before proceeding with the left turn manoeuvring. Figure 21 shows the virtual simulation testing with the trajectory of the ego vehicle to overtake the bike before the junction. Meanwhile, Figure 22 shows the comparison between the vehicles in real and virtual environment model. The real scenario is obtained from the recorded data using the low-cost instrumented vehicle.

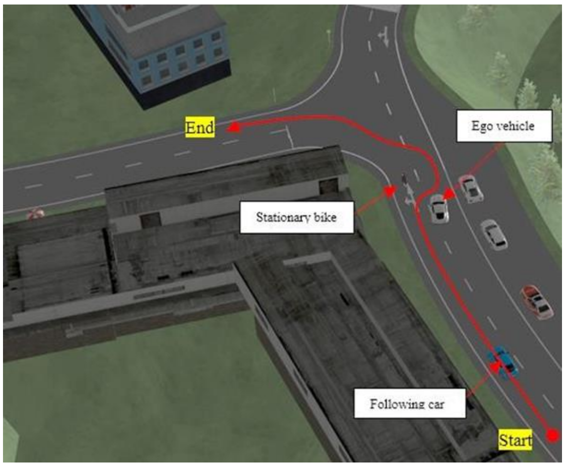


Figure 21. Bird eye view

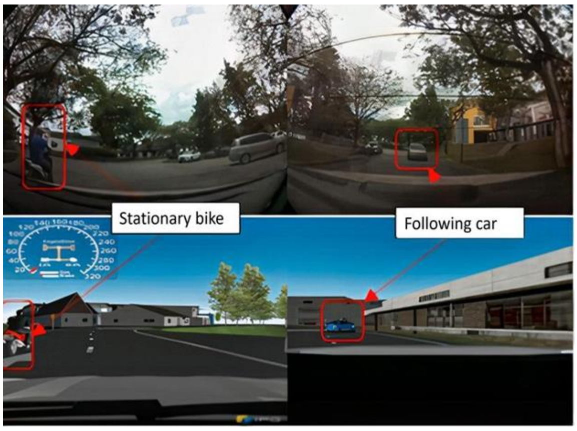


Figure 22. On-board camera comparison for junction overtaking scenario

4.3 Road Object Avoidance

For the third scenario, it focuses on the dynamic obstacles in the ego-vehicle's path. The ego vehicle is moving in longitudinal directions and static obstacles are placed in lateral direction on the ego vehicle's path. Therefore, the ego vehicle needs to apply manoeuvre adjustment using steering input to the left side of the road to avoid collision with the road object. Similar to the previous cases, IPG CarMaker is used to recreate scenarios in virtual environment. Figure 23 shows the road object avoidance test scenario developed using Scenario Editor in IPG CarMaker and Figure 24 shows the comparison between the video frame captured from the instrumented vehicle and the video frame captured from the virtual vehicle. The top figure shows the video frame recorded from instrumented vehicle and the bottom figure shows the video frame from IPG CarMaker. The frame on the left is the front view, while the right is the rear view.

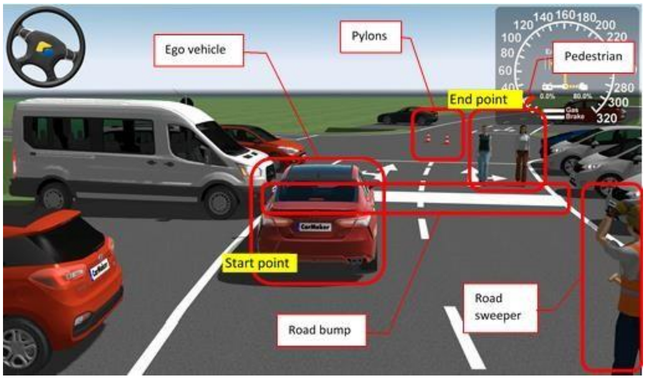


Figure 23. Road object avoidance test scenario developed using IPG CarMaker

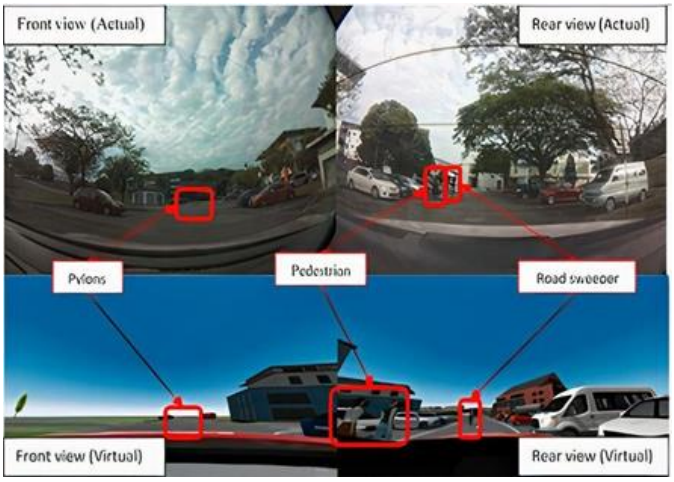


Figure 24. Onboard camera comparison for road object avoidance scenario

4.3.1 Pedestrian road crossing

In this scenario, the ego vehicle is following a red car in front with a group of pedestrians crossing the road in front. The red car in front applied brake and stopped to let the pedestrians cross the road. Therefore, the ego vehicle needs to apply brake to slow down while maintaining distance with the vehicle in front and stop the vehicle to wait for the pedestrians to cross. Once the pedestrians crossed the road, the red car in front will start moving and the ego vehicle will follow the red car. Figure 25 shows the pedestrian road crossing virtual scenario developed and Figure 26 shows the onboard camera comparison between real-world and virtual simulation.

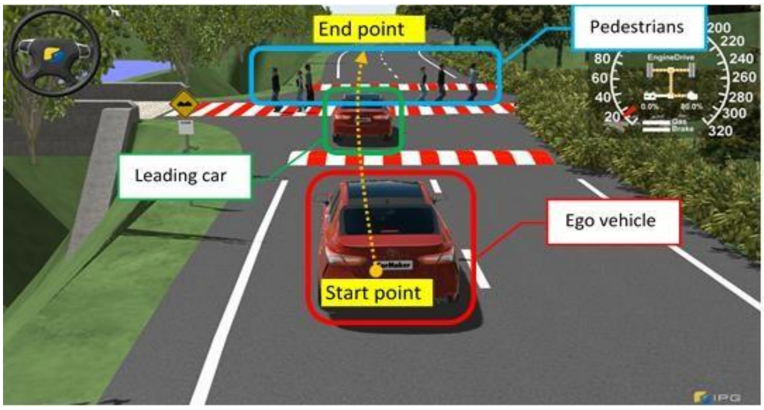


Figure 25. Pedestrian road crossing test scenario developed using IPG Car Maker

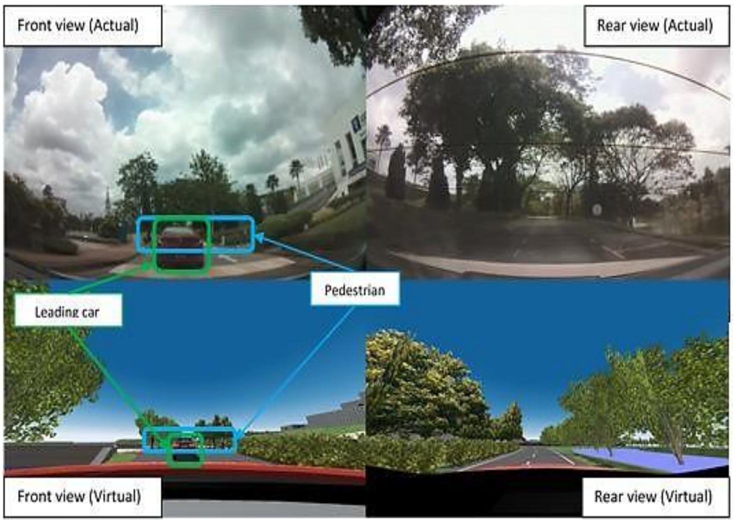


Figure 26. Onboard camera comparison for pedestrian crossing scenario

4.4 Scenario-Based Testing Simulation

Virtual scenarios are developed in the IPG CarMaker based on actual scenarios captured from the instrumented vehicle. In order to verify the driving data, the recorded data from the instrumented vehicle is compared with the data simulated from the IPG CarMaker. In this experiment, the driver in the instrumented vehicle is a human driver whereas the driver in the IPG CarMaker virtual vehicle is the built-in IPGDriver model which can reproduce driving behaviors of average driver.

4.4.1 Road obstacle avoidance

Figure 27 shows the screenshots of virtual ego vehicle manoeuvring to avoid the road obstacle in front by sliding to the left side of the road. Figures 28(a-e) show the vehicle response obtained from the instrumented vehicle and the IPG CarMaker simulation. The vehicle responses and the driving inputs that are investigated are the steering angle, gas pedal input, brake pedal input, pitch rate and yaw rate of the vehicle.

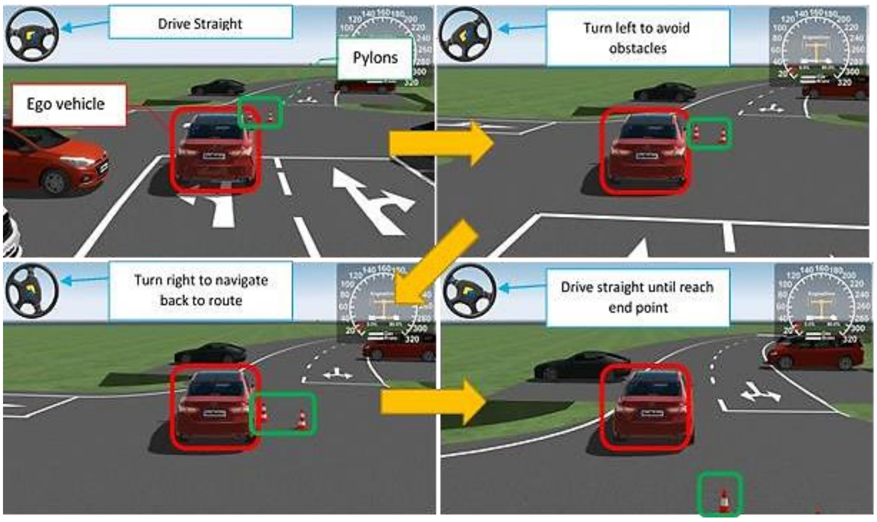
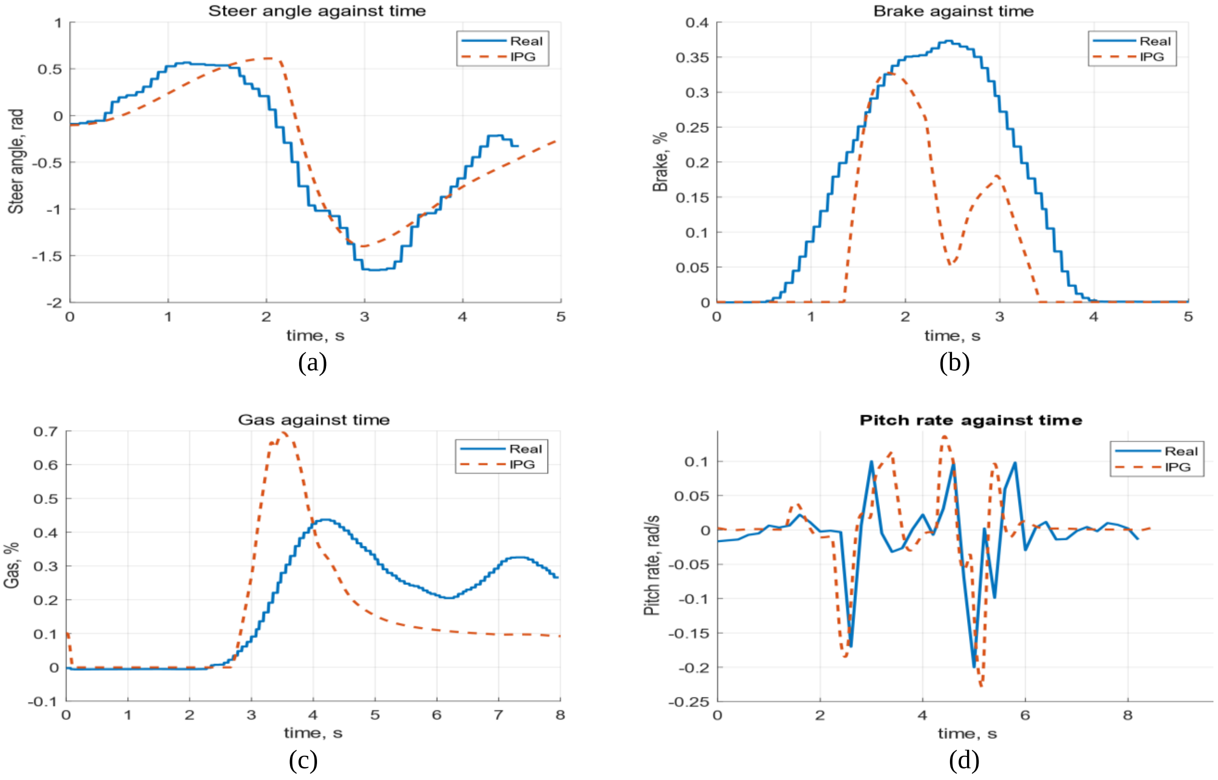

Drive Straight
Pylons
Ego vehicle
Turn left to avoid obstacles
Turn right to navigate back to route
Drive straight until reach end point

Figure 27. Road obstacle avoidance scenario manoeuver



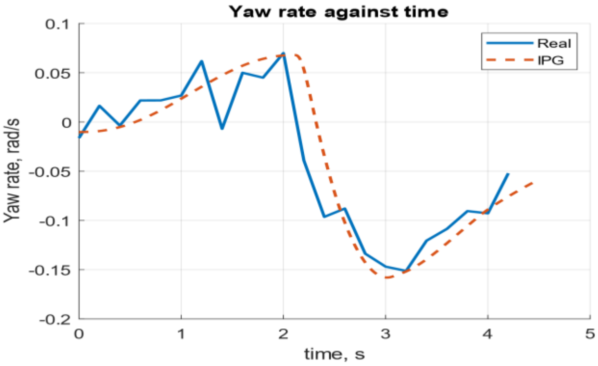
(a)

(b)

(c)

(d)

Figure 28. (a) Steering angle input, (b) Brake pedal input, (c) Gas pedal input, (d) Pitch rate response



(e)

Figure 28. (cont.) (e) Yaw rate response for road obstacle avoidance

Table 7. Percentage of RMS error for road obstacle avoidance test

Testcase	Observation data	Root Mean Square (RMS)		Percentage of error (%)
		IPG	Real	
Road obstacle avoidance	Steering angle, rad	0.64810	0.72350	10.420
	Brake input, %	0.13060	0.14510	9.993
	Gas input, %	0.15900	0.15410	3.180
	Pitch rate, rad/s	0.05752	0.05440	5.699
	Yaw rate, rad/s	0.07661	0.07261	5.509

From the data plots shown above, it can be observed that the driving input produced by the IPGDriver and the vehicle response from the IPG CarMaker is similar to the actual driving input and vehicle response. From Table 7, it can be observed that the steer angle, brake input, gas input, pitch rate and yaw rate show the percentage difference of RMS errors about 10.42%, 9.993%, 3.18%, 5.699% and 5.509%, respectively. Figures 28(b) and 28(c) show noticeable differences in yaw rate and pitch rate between the real-world and the virtual scenario. These differences are most likely due to the slight variation between the virtual road model developed and the real-world such as the condition of the road, the slope of the road environment, height of the bump and the parameters of the actual vehicle and the virtual vehicle as well as the driving style of the human driver and the virtual IPGDriver. From Figure 28(b), it can be observed that at around 1.5 seconds, the IPSDriver pressed the brake pedal later than the human driver. Therefore, the IPGDriver needs to apply braking force to the vehicle at a shorter time as shown by the gradient of the pedal inputs. This caused the vehicle to stop at a higher deceleration. As a result, the vehicle pitched down at a higher rate due to higher inertia. Similar to the braking scenario, the IPGDriver also applied gas input to the vehicle at a higher rate as shown in Figure 28(c). Therefore, the vehicle pitched up at a higher rate. Meanwhile, the large yaw rate difference at around 2 seconds of the experiment is due to the higher steer angle provided by the IPGDriver at 2 seconds as shown in Figure 28(a). However, it can still be observed that the pattern of the vehicle response and driving data obtained from the virtual environment followed closely to the data captured from real world environment. Furthermore, the large RMS errors also show that the driving behaviour between a human driver and the default virtual driver are different. Therefore, the IPGDriver is not suitable to be used as driver model for evaluation of safety system of autonomous vehicles in Malaysia. This is mainly because the IPGDriver from IPG CarMaker is the generic model and it does not represent exactly the driving pattern for Asian driving pattern.

4.4.2 Pedestrian road crossing

For pedestrian road crossing scenario, the ego vehicle will follow the car in front and stop behind the red car while waiting for the pedestrians to cross the road. After the pedestrians crossed the road, the leading car start moving. Therefore, the ego vehicle will then start moving following the leading car. Figure 29 shows the screenshots of the manoeuvres for the pedestrian road crossing scenario, while Figures 30(a-c) shows the vehicle response obtained from the instrumented vehicle and the IPG CarMaker simulation. The vehicle responses and the driving inputs that are investigated are the steering angle, gas pedal input, brake pedal input, pitch rate and yaw rate of the vehicle. From the data plots shown in Figures 30(a-c), it can be observed that the driving input applied by the driver and the vehicle response from the IPG CarMaker simulation is having different fromthe actual driving input and vehicle response but overall they are having similar trend. From Figure 30(a), both the human driver and IPGDriver applied up to 46.57% and 42.53% of the brake pedal input respectively during the start of the experiment. However, the human only applied the large braking force at the first 2 seconds and the braking input is reduced to around 30% after the car stopped moving, whereas the IPGDriver maintained a higher braking force until the distance from the leading car is more than 2 m which is the default safe distance specified. From Figure 30(b), it can be observed that once there was enough gap from the leading car, both the human driver and IPGDriver applied throttle input in following the leading car. However, it can be observed that the

IPGDriver behaves more aggressively compared to the human driver as shown in the Figures as the IPGDriver's inputs have higher gradient. As a result, the brake input, gas input and pitch rate show the percentage difference of RMS errors about 0.1388%, 12.83% and 14.75% respectively. The large RMS errors show that the IPGDriver is unable to drive like a real human driver in these cases. Therefore, a driver model that can reproduce these humans' driving behaviours is needed. Meanwhile, the pitch rate of the vehicle obtained from the real world and the IPG CarMaker has noticeable differences. This is due to the difference in road environment between the real-world and the virtual road model created. In the real environment, the slope profile at the second bump is higher which results in a higher pitch rate value at 2.5 to 4 seconds of Figure 30(c). Therefore, a more precise measurement of the real environment is needed for development of the environment model to make sure the vehicle response in the real and virtual environment is identical. The summary of RMS percentage error for pedestrian crossing test is shown in Table 8.

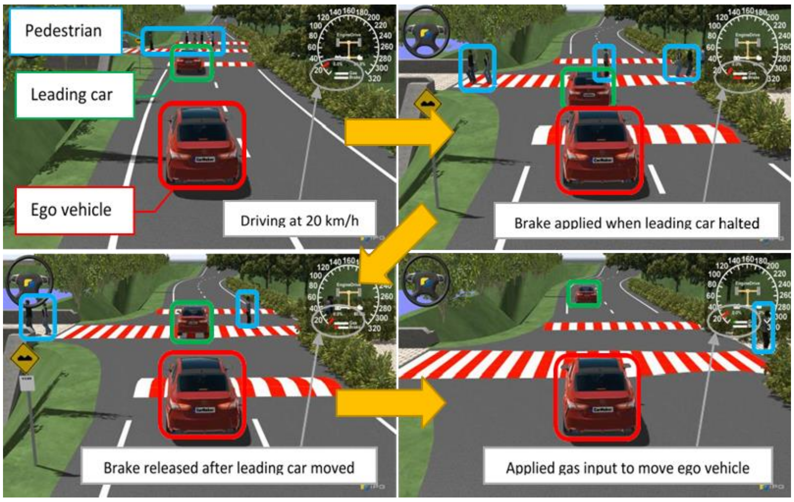
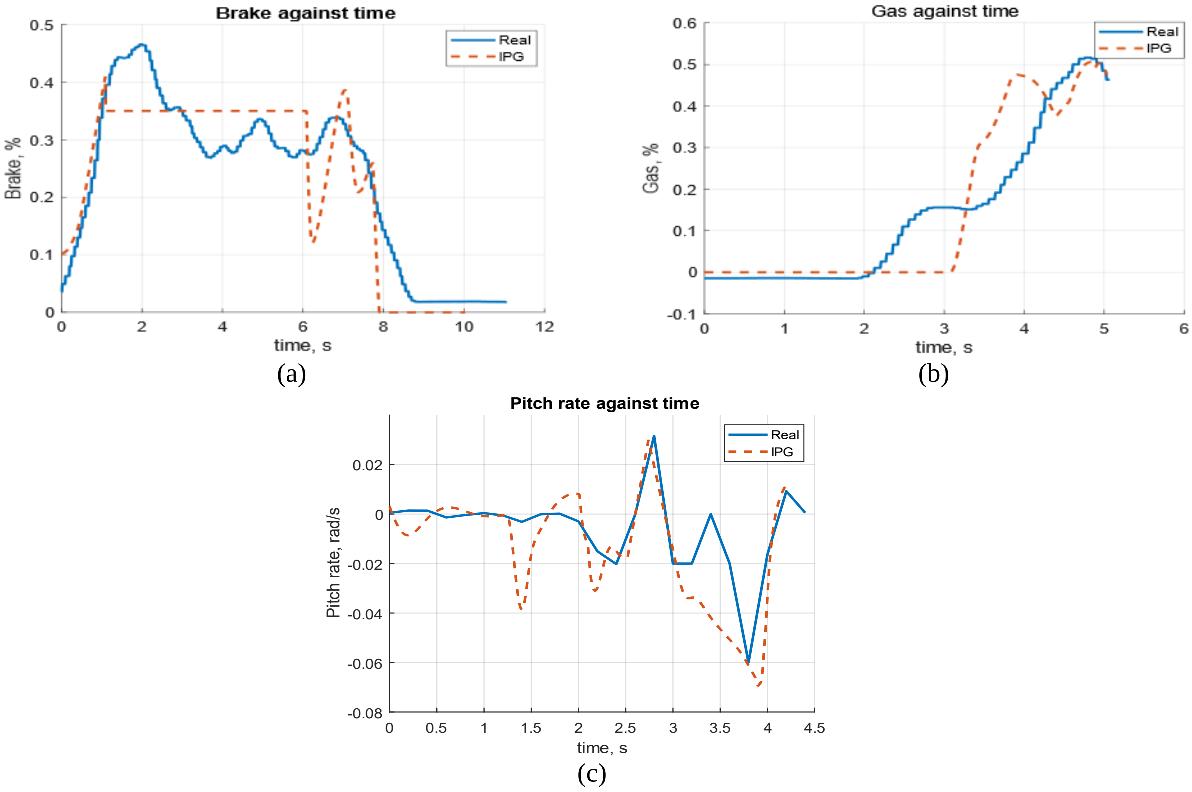

Pedestrian
Leading car
Ego vehicle
Driving at 20 km/h
Brake applied when leading car halted
Brake released after leading car moved
Applied gas input to move ego vehicle

Figure 29. Pedestrian crossing road scenario's manoeuvre



(a)

(b)

(c)

Figure 30. (a) Brake pedal input, (b) Gas pedal input and (c) Pitch rate response for pedestrian crossing scenario

Table 8. Percentage of RMS error for pedestrian crossing scenario

Testcase	Observation data	Root Mean Square (RMS)		Percentage of error (%)
		IPG	Real	
Pedestrian crossing	Brake input, %	0.1441	0.1439	0.1388
	Gas input, %	0.2022	0.1792	12.83
	Pitch rate, rad/s	0.01648	0.01405	14.75

4.5 Summary of Scenario Based Testing

The scenario-based testing is carried out by extracting a real world scneario and replicating it in the virtual environment. It is a novel approach using IPG CarMaker and overall, it is able to achieve RMS error between 3%-15%. The inconsistency of the result is mainly due to the driver model being applied to virtual testing. IPG Driver. Although it was shown that the input from IPG Driver managed to complete the scenario-based testing and drive through the specified route, the input is not similar to the human driver in this case. This finding suggested that the IPG Driver is not suitable to be used as the input driver as it cannot provide a consistent input similar to a common driver. In the future, it can be suggested to carry out the scenario-based testing with a driver in the loop simulator to enable a realistic input from a human driver. An alternative way is by considering a more reliable driver model that can closely follow the behaviour of the human.

5.0 CONCLUSIONS

In this study, a low-cost sensor-based configuration of an instrumented vehicle for scenario-based testing of autonomous vehicles has been developed. Low-cost, off-the-shelf commercial components were used to instrument the vehicle. The instrumented vehicle is validated using longitudinal and lateral tests based on SAE standards and the results show that the instrumented vehicle is able to capture the vehicle state with an average RMSE of less than 15%. The instrumented vehicle is used to collect driving data in the University of Nottingham Malaysia. YOLOv5 deep learning object detection algorithm is used to identify road users and road objects in the scene. Virtual scenarios which based on the scenarios identified from the video data recorded were developed. The results show that the virtual simulations were able to reproduce the actual scenarios. Therefore, from the experiment, it can be concluded that scenario-based testing can be done by using virtual environment with instrumented data recording vehicle at low cost. Furthermore, it can also be observed that the virtual driver from the simulator is unable to reproduce the driving style of a human driver. Therefore, for future work, a driver model which can reproduce average human driver will be developed to replace the IPGDriver and provide more realistic human driving inputs to the simulator. Besides, autonomous vehicle controllers such as Autonomous Emergency Braking (AEB) and Adaptive Cruise Control (ACC) systems which adapt to the local road environment can be designed and tested using the scenario-based simulation testing developed.

6.0 ACKNOWLEDGMENTS

This work is part of the research project entitled “Vehicle-in-the-loop Safety Testing Platform for Autonomous Vehicle using Malaysian Road and Traffic Environment.” This research is fully supported by Yayasan Tun Ismail Mohamed Ali Berdaftar (YTI) under Permodalan Nasional Berhad and the grant is led by Assistant Professor Ir. Ts. Dr. Vimal Rau Aparow. The team would like to thank Asia Pacific University of Technology and Innovation (APU) for the space provided to conduct field testing. The team also would like to thank University of Nottingham Malaysia for continuous support to conduct research in this area.

7.0 REFERENCES

[1] S. A. Bagloee, M. Tavana, M. Asadi, and T. Oliver, “Autonomous vehicles: challenges, opportunities, and future implications for transportation policies,” *Journal of Modern Transportation*, vol. 24, no. 4, pp. 284–303, 2016.

[2] F. Rosique, P. J. Navarro, C. Fernández, and A. Padilla, “A systematic review of perception system and simulators for autonomous vehicles research,” *Sensors (Switzerland)*, vol. 19, no. 3, pp. 1-29, 2019.

[3] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, “CARLA: An open urban driving simulator,” in *1st Conference on Robot Learning*, United State, 2017.

[4] H. Fan *et al.*, “Baidu apollo EM motion planner,” *ArXiv* [Online], 2018, Available: https://arxiv.org/pdf/1807.08048.pdf.

[5] G. Rong *et al.*, “LGSVL simulator: A high fidelity simulator for autonomous driving,” *ArXiv* [Online], 2020, Available: https://arxiv.org/pdf/2005.03778.pdf.

[6] S. Riedmaier, T. Ponn, D. Ludwig, B. Schick, and F. Diermeyer, “Survey on scenario-based safety assessment of automated vehicles,” *IEEE Access*, vol. 8, pp. 87456–87477, 2020.

[7] S. Riedmaier, D. Schneider, D. Watzenig, F. Diermeyer, and B. Schick, “Model validation and scenario selection for virtual-based homologation of automated vehicles,” *Applied Sciences (Switzerland)*, vol. 11, no. 1, pp. 1–24, 2021.

[8] J. D. Setiawan, M. Safarudin, and A. Singh, “Modeling, simulation and validation of 14 DOF full vehicle model,” in *International Conference on Instrumentation, Communication, Information Technology and Biomedical Engineering*, pp. 1-6, Bandung Indonesia, 2009.

[9] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, “Vision meets robotics: The KITTI dataset,” *International Journal of Robotics Research*, vol. 32, no. 11, pp. 1231–1237, 2013.

[10] X. Huang *et al.*, “The apolloscape dataset for autonomous driving,” *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, vol. 2018, pp. 1067–1073, 2018.

[11] P. V. Gopi Krishna Rao, M. V. Subramanyam, and K. Satyaprasad, “Study on PID controller design and performance based on tuning techniques,” in *International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT 2014)*, Kanyakumari, India, pp. 1411–1417, 2014.

[12] H. Caesar *et al.*, “nuScenes: A multimodal dataset for autonomous driving,” in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2020)*, Seattle, USA, pp. 11618–11628, 2020.

[13] J. Mao *et al.*, “One million scenes for autonomous driving: Once dataset,” *ArXiv* [Online], 2021, Available: https://arxiv.org/pdf/2106.11037.pdf.

[14] M. Sheeny, E. D. Pellegrin, S. Mukherjee, A. Ahrabian, S. Wang, and A. Wallace, “Radiate: A radar dataset for automotive perception in bad weather,” in *IEEE International Conference on Robotics and Automation (ICRA2021)*, Xian, China, pp. 1-7, 2021.

[15] K. Burnett *et al.*, “Boreas: A multi-season autonomous driving dataset,” *The International Journal of Robotics Research*, vol. 42, pp. 33-42, 2023.

[16] H.-H. Braess *et al.*, "Vieweg Handbuch Kraftfahrzeugtechnik," *Spinger Vieweg Wiesbaden*, Germany, 2013.

[17] M. Zhu, X. Wang, and Y. Wang, “Human-like autonomous car-following model with deep reinforcement learning,” *Transportation Research Part C: Emerging Technologie.*, vol. 97, pp. 348–368, 2018.

[18] I. Bae *et al.*, “Self-driving like a human driver instead of a robocar: Personalized comfortable driving experience for autonomous vehicles,” in *33rd Conference on Neutral Information Processing System (NeurIPS 2019)*, Vancouver, Canada, 2019.

[19] Y. Dong *et al.*, “Mcity data collection for automated vehicles study,” *ArXiv* [Online], 2019, Available: https://arxiv.org/pdf/1912.06258.pdf.

[20] J. Rehder and R. Siegwart, “Camera/IMU calibration revisited,” *IEEE Sensors Journal*, vol. 17, no. 11, pp. 3257–3268, 2017.

[21] M. Moussa, A. Moussa, and N. El-Sheimy, “Steering angle assisted vehicular navigation using portable devices in GNSS-denied environments,” *Sensors (Switzerland)*, vol. 19, no. 7, pp. 1-18, 2019.

[22] L. Fridman, D. E. Brown, W. Angell, I. Abdić, B. Reimer, and H. Y. Noh, “Automated synchronization of driving data using vibration and steering events,” *Pattern Recognition Letters*, vol. 75, pp. 9–15, 2016.

[23] W. Yao *et al.*, “On-road vehicle trajectory collection and scene-based lane change analysis: Part II,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 1, pp. 206–220, 2017.

[24] S. Feraco, A. Bonfitto, N. Amati, and A. Tonoli, “Redundant multi-object detection for autonomous vehicles in structured environments,” *Electrical Engineering in Transport*, vol. 24, no. 1, pp. 1–17, 2022.

[25] A. Agnoor, P. Atmakuri, and R. Sivanandan, “Analysis of driving behaviour through instrumented vehicles,” in *14th International Conference on COMmunication Systems and NETworkS (COMSNETS 2022)*, Bangalore, India, pp. 700–706, 2022.

[26] G. Li, Y. Yang, S. Li, X. Qu, N. Lyu, and S. E. Li, “Decision making of autonomous vehicles in lane change scenarios: Deep reinforcement learning approaches with risk awareness,” *Transportation Research Part C: Emerging Technologies*, vol. 134, p. 103452, 2022.

[27] D. Rempe, J. Philion, L. J. Guibas, S. Fidler, and O. Litany, “Generating useful accident-prone driving scenarios via a learned traffic prior,” in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2022)*, New Orleans, USA, pp. 17284–17294, 2022.

[28] J. Cheng, L. Zhang, Q. Chen, X. Hu, and J. Cai, “A review of visual SLAM methods for autonomous driving vehicles,” *Engineering Applications of Artificial Intelligence*, vol. 114, p. 104992, 2022.

[29] A. Amini *et al.*, “VISTA 2.0: An open, data-driven simulator for multimodal sensing and policy learning for autonomous vehicles,” in *IEEE International Conference on Robotics and Automation (ICRA 2022)*, Philadelphia, USA, pp. 2419–2426, 2022.

[30] Q. Li, Z. Peng, L. Feng, Q. Zhang, Z. Xue, and B. Zhou, "MetaDrive: Composing diverse driving scenarios for generalizable reinforcement learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 3, pp. 3461–3475, 2023.

[31] H. Yue, L. Zhang, H. Shan, H. Liu, and Y. Liu, "Estimation of the vehicle's centre of gravity based on a braking model," *Vehicle System Dynamics*, vol. 53, no. 10, pp. 1520–1533, 2015.

[32] J. Lv, A. A. Ravankar, Y. Kobayashi, and T. Emaru, "A method of low-cost IMU calibration and alignment," in *IEEE/SICE International Symposium on System Integration (SII 2016)*, Sapporo, Japan, pp. 373–378, 2016.

[33] M. Garrosa, E. Olmeda, S. F. Del Toro, and V. Díaz, "Holistic vehicle instrumentation for assessing driver driving styles," *Sensors*, vol. 21, no. 4, pp. 1–28, 2021.

[34] A. Parra, D. Cagigas, A. Zubizarreta, A. J. Rodriguez, and P. Prieto, "Modelling and validation of full vehicle model based on a novel multibody formulation," in 45th *Annual Conference of the IEEE Industrial Electronis Society*, Lisbon, Portugal, pp. 675–680, 2019.

[35] C. J. Hong and V. R. Aparow, "System configuration of human-in-the-loop simulation for level 3 autonomous vehicle using IPG CarMaker," in *IEEE International Conference on Internet of Things and Intelligence Systems (IoTaIS 2021)*, Bandung, Indonesia, pp. 215–221, 2021.

[36] V. R. Aparow *et al.*, "Scenario based simulation testing of autonomous vehicle using Malaysian road," in *5th International Conference on Vision, Image and Signal Processing (ICVISP 2021)*, Kuala Lumpur, Malaysia, pp. 33–38, 2021.