

FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

A Driving Simulation Study For Communication And Awareness Enhancement of ADAS Algorithms

Pedro Henrique Delgadinho Simões



Mestrado em Engenharia Informática e Computação

Supervisor: Prof. Alexandre Valle

Second Supervisor: João Palmeira

Third Supervisor: Rúben Silva

July 25, 2023

A Driving Simulation Study For Communication And Awareness Enhancement of ADAS Algorithms

Pedro Henrique Delgadinho Simões

Mestrado em Engenharia Informática e Computação

Approved in oral examination by the committee:

President: Prof. Daniel Mendes

Referee: Prof. Alexandre Valle

Referee: Prof. Paulo Dias

July 25, 2023

Resumo

O desenvolvimento e aperfeiçoamento dos Sistemas Avançados de Assistência à Condução (ADAS), para atenuação dos acidentes causados por erro humano, têm sido um dos grandes focos da indústria automóvel. Estes sistemas consistem em algoritmos que utilizam sensores para determinar as reações adequadas perante cenários perigosos, com o objetivo de aumentar a segurança rodoviária. No entanto, as condições meteorológicas e determinados cenários críticos, têm um grande impacto no desempenho dos ADAS. Por conseguinte, os simuladores virtuais 3D desempenham um papel fulcral na resolução de alguns destes problemas, uma vez que fornecem os cenários necessários, garantem condições de segurança ao eliminar efeitos nefastos da experimentação no mundo real e reduzem as despesas associadas a esta última.

No entanto, os condutores de veículos apresentam geralmente alguma dificuldade na compreensão do funcionamento destes algoritmos ADAS, sobretudo na forma como estes algoritmos possibilitam a redução de situações indesejáveis causadas por erro humano.

Assim, esta dissertação visa melhorar a forma como o condutor comum percebe os ADAS, proporcionando uma experiência interativa que comunica eficazmente os benefícios destes algoritmos, utilizando um simulador de condução 3D. Após uma pesquisa cuidada sobre os simuladores existentes, o CARLA Simulator foi a plataforma escolhida para realizar esta experiência, uma vez que proporciona uma elevada qualidade de renderização, realismo e controlo extensivo sobre a simulação. Nesta dissertação, a solução proposta passa pela submissão de condutores comuns a dois cenários diferentes de situações críticas (sinais de *stop*, semáforos e peões e ciclistas a atravessar perigosamente a estrada), através do simulador, e pela posterior comparação do desempenho do participante com a performance de um algoritmo de Veículo Autónomo (AV) com funcionalidades ADAS, nesses mesmos cenários. O objetivo desta solução é demonstrar a disparidade entre a conduta de um condutor comum face aos algoritmos ADAS, através de um *dashboard* que, de forma concisa e clara, comunica os benefícios destes sistemas.

Para testar esta solução, foi desenvolvido um protótipo dividido em três componentes principais - Interface, Simulação e *Dashboard*. Foi também criado um protocolo de teste com o método da experiência, requisitos de equipamento, critérios de seleção dos participantes e métricas de avaliação. Após a implementação deste protocolo, a experiência foi testada utilizando o protótipo desenvolvido, nomeadamente numa “feira de emprego” com a Continental Engineering Services.

Os resultados obtidos, a partir de uma amostra de 20 participantes, revelaram que o protótipo não só comunicou eficazmente os benefícios dos algoritmos ADAS, como também aumentou a confiança dos participantes nos sistemas, melhorando a compreensão das suas capacidades em situações críticas. Além disso, foi revelado que o algoritmo InterFuser teve um desempenho superior ao da maioria dos condutores.

Assim, tendo em conta os resultados e reflexões decorrentes, esta dissertação cumpriu o seu objetivo de estudar e propor uma solução para aumentar a sensibilização relativa às capacidades dos algoritmos ADAS e comunicar os benefícios destes sistemas, utilizando um simulador de condução.

Abstract

The Automotive Industry has placed significant emphasis on the development and enhancement of Advanced Driving Assistance Systems (ADAS) to mitigate accidents caused by human error. These systems consist of algorithms that use sensors to determine appropriate actions in dangerous scenarios, aiming to enhance road safety. However, weather conditions and actual critical scenarios have a major impact on the performance of ADAS. Therefore, 3D virtual simulators play an important role in solving some of these problems as they provide the scenarios needed, ensure safe conditions by eliminating the consequences of real-world experimentation, and reduce the expenses associated with the latter.

However, common vehicle drivers struggle to comprehend how these ADAS algorithms operate, particularly the extent to which they successfully address specific situations that are prone to human errors.

Therefore, this dissertation aims to improve the way common drivers perceive ADAS by providing an interactive experience that efficiently communicates the benefits of ADAS algorithms using a 3D virtual driving simulator.

After conducting in-depth research on the existing simulators, CARLA Simulator was the chosen platform for this experiment since it provides high render quality, realism, and extensive control over the simulation.

In this dissertation, the proposed solution involves subjecting common drivers to two different scenarios that included critical situations (stop signs, traffic lights, and pedestrians and cyclists crossing the road dangerously) and subsequently comparing the participant's performance with an Autonomous Vehicle (AV) algorithm with ADAS features' performance, in those same scenarios.

The purpose of this solution is to demonstrate the difference in people's behavior compared to ADAS algorithms, using a dashboard that communicates, in an effective and concise manner, the benefits of these systems.

To test this solution, a prototype was developed and divided into three main components – Interface, Simulation, and Dashboard. A test protocol was also created with the experiment's process, equipment requirements, participant selection criteria, and evaluation metrics. After implementing a well-structured test protocol, the experiment was tested using the developed prototype in a career fair with the sponsorship of Continental Engineering Services.

The findings obtained from a sample of 20 participants revealed that the prototype not only, effectively communicated the benefits of ADAS algorithms, but also increased the participants' trust in the systems, and improved their awareness of the algorithms' capabilities in critical situations. Additionally, it demonstrated that the InterFuser algorithm outperformed the majority of drivers. With the results and derived considerations in mind, this dissertation accomplished its goal of studying and proposing a solution to enhance awareness towards ADAS algorithms capabilities and communicating the benefits of these systems using a 3D virtual driving simulator.

Agradecimentos

Este esforço e dedicação é dedicado à minha família, que me mostrou o que é verdadeira união. À minha mãe, que me ensinou e continua a ensinar o que é ter força de vontade e nunca desistir. Que me mostrou o que é relativizar dificuldades e disfrutar das pequenas coisas que tomamos como garantidas. Tenho o maior orgulho em ti e como questionas quando te o digo, fica escrito para nunca duvidares. Ao meu pai, que sempre me mostrou o que é ser um Superhomem, que me ensinou o que é ter integridade e que sempre me fez sentir que consigo fazer tudo aquilo que meto na cabeça. Ao meu irmão, que me mostrou desde pequeno o que é a alegria da música através do brilho dos seus olhos e como podemos usá-la como refúgio dos momentos mais complicados. À minha prima Sarito, a minha segunda irmã, que nunca falhou em acompanhar-me nas minhas maluquices em época de exames, mas também em apoiar-me nos momentos sérios.

À Catarina, a minha paz, que me acompanhou desde o início e mudou o meu percurso académico com a sua dedicação inagualável, mas sobretudo a minha vida com o seu apoio constante, com o seu amor interminável e com a sua atenção ao pormenor que ainda hoje me surpreende.

Aos meus amigos pelo companheirismo.

Aos meus co-orientadores, João e Rúben, por me terem acolhido de forma tão natural e por terem tornado este percurso um bocado mais simples, com a sua leveza e disponibilidade constante. À Continental Engineering Services pela experiência e ao *Team Leader* Pedro, pela preocupação durante todo este processo.

E por último, ao meu orientador, Professor Alexandre, que me ajudou do início ao fim a descomplicar esta fase tão complicada, guiando-me e defendendo os meus interesses. Um agradecimento especial pela sua preocupação e cuidado terem superado aquilo que podemos considerar os comuns "deveres de um orientador".

Pedro Simões

*“You can be happy and also going through something really hard at the same time.
You don’t have to pick one or the other. . . .
You can’t wait until life isn’t hard anymore, before you decide to be happy.”*

Jane Marczewski

Contents

1	Introduction	1
1.1	Context	1
1.2	Problem	2
1.3	Hypothesis	2
1.4	Goals	2
1.5	Outline	3
2	ADAS Algorithm Simulation and Validation	4
2.1	Introduction to Automotive Industry	4
2.1.1	Concepts	5
2.1.2	ADAS Algorithms	8
2.1.3	ADAS Common Problems and Challenges	10
2.1.4	How ADAS Algorithms are Tested	11
2.2	Simulation	12
2.2.1	Virtual 3D Simulators	13
2.3	Simulation in Automotive Industry	13
2.3.1	Virtual Simulation for ADAS	14
2.3.2	Simulator Technology	14
2.3.3	Comparison Between Simulators	18
2.3.4	CARLA Simulator	20
2.4	HCI Methodology Evaluation	22
2.5	Summary	24
3	Human Driver VS ADAS Simulation Framework	25
3.1	Context and Problem	25
3.2	Solution Outlook	26
3.2.1	Requirements	27
3.3	Prototype	31
3.3.1	Architecture	32
3.3.2	Interface Component	33
3.3.3	Simulation Component	34
3.3.4	Dashboard Component	34
3.4	Summary	38
4	Experiment	40
4.1	Test Protocol	40
4.1.1	Goals	40
4.1.2	General Process	41

4.1.3	Participant Population	42
4.1.4	Equipment	43
4.1.5	Informed Consent Form	43
4.1.6	Pre-Test Questionnaire	43
4.1.7	Participant Tasks	44
4.1.8	Metrics	44
4.1.9	Post-Test Questionnaire	45
4.2	Test Environment	45
4.3	User Feedback	47
4.4	Summary	48
5	Results and Discussion	49
5.1	Results	49
5.2	Discussion	57
5.3	Summary	61
6	Conclusions and Future Work	63
6.1	General Conclusions	63
6.2	Future Work	64
	References	66
A	Dashboard	70
A.1	Scenario One Page	70
A.2	Scenario Two Page	73
A.3	Final Results Page	76
A.4	Google Forms Page	77
B	Forms	78
B.1	Informed Consent Form	78
B.2	Pre-Test Questionnaire	79
B.3	Pos-Test Questionnaire	79

List of Figures

2.1	ADAS Sensors in a modern vehicle (based on [13]).	5
2.2	Bird's-eye View for parking.	6
2.3	Adaptive Cruise Control architecture [37].	9
2.4	Lane Departure Warning System diagram (based on [14]).	10
2.5	Examples of complex Lane Detection and Tracking conditions [19].	10
2.6	Euro NCAP ADAS Tests (based on [21]).	11
2.7	Steering to Avoid Obstacle.	11
2.8	Examples of ACC critical events.	12
2.9	CarSim Simulator.	15
2.10	Gazebo Simulator.	15
2.11	MATLAB/Simulink Simulator.	16
2.12	CARLA Simulator.	17
2.13	LGSVL Simulator.	17
2.14	Different weather conditions and time of day [5].	20
2.15	Town5 – Example of preset town [33].	21
2.16	Example of some sensors' data [34].	22
2.17	A/B Tests.	23
3.1	Scenario 1 Overview and Path.	27
3.2	Scenario 2 Overview and Path.	28
3.3	Comprehensive Scene Understanding by Sensor Fusion [32].	29
3.4	Algorithm's Views and Sensors.	29
3.5	Dashboard Mock-up.	30
3.6	Prototype Main Components and Relation.	31
3.7	Prototype Architecture.	32
3.8	Developer Commands Interface.	33
3.9	Logitech G29 Steering Wheel and Pedals.	34
3.10	Critical Card Example.	35
3.11	Score Card Example.	36
3.12	Infraction Counter Card Example.	36
3.13	Fraction Card Example.	37
3.14	Collision Bar Chart.	38
3.15	Route Doughnut Chart.	38
4.1	Traffic Light Intersection In United States of America.	42
4.2	Black Dots Indicating the Path that the Participant Must Follow On the Scenarios.	44
4.3	Continental Engineering Services Exhibitor.	46
4.4	Experiment Setup.	46

4.5	Participant Experimenting the First Scenario of the Prototype.	47
5.1	Distribution of Participants Per Age.	49
5.2	Driver's License Years.	50
5.3	Distribution of Participants Per Gender.	50
5.4	Highest Level of Education Distribution.	50
5.5	Knowledge of ADAS.	51
5.6	Level of ADAS Trust.	51
5.7	InterFuser Algorithm Mistake Due To Big Shadow.	53
5.8	Level of Realism in Critical Situations.	54
5.9	Critical Situations In Real Life.	55
5.10	Participants Outperformed Algorithm In Scenario 1.	55
5.11	Participants Outperformed Algorithm In Scenario 2.	55
5.12	Participants Outperformed Algorithm In Final Results.	56
5.13	Quality of Dashboard.	56
5.14	Classification of Experiment ADAS Benefit Communication.	57
5.15	Level of ADAS Trust.	57
5.16	Age and Performance Correlation.	58
5.17	Years of Driver's License and Critical Situations' Realism Opinion Correlation. .	59
5.18	Level of ADAS Trust Before and After Experiment Correlation.	60

List of Tables

2.1	Sensors' Comparison (based on [28]).	7
2.2	SAE Levels of Automation (based on [27]).	8
2.3	Simulators comparison (based on [10]).	19
3.1	Critical Card prop description and type.	35
3.2	Score Card prop description and type.	36
3.3	Infractions Counter Card prop description and type.	36
3.4	Fraction Card prop description and type.	37
3.5	Graphs Card prop description and type.	37
3.6	Collision Bar Chart prop description and type.	37
3.7	Route Completion Doughnut Chart prop description and type.	38
5.1	Simulation Results.	52

Abreviaturas e Símbolos

ADAS	Advanced Driver Assistance Systems
AB	Automatic Braking
ACC	Adaptive Cruise Control
AV	Autonomous Vehicle
API	Application Programming Interface
DART	Dynamic Animation and Robotics Toolkit
GUI	Graphical User Interface
HCI	Human-Computer Interaction
LDWS	Lane Departure Warning System
LiDAR	Light Detection and Ranging
OEM	Original Equipment Manufacturer
Radar	Radio Detection and Ranging
ROS	Robot Operating System
UE4	Unreal Engine 4

Chapter 1

Introduction

1.1 Context

Safety is one of the biggest concerns in the automotive industry. Therefore, Advanced Driving Assistance Systems (ADAS) are being developed and improved to prevent accidents that can cause injuries to people inside and outside vehicles [13]. Some of these systems already exist in modern cars, such as Adaptive Cruise Control (ACC) which makes the vehicle slow down or speed up on the highway according to the situation, Blind Spot Detection, that detects blind spots and warns the driver and Automatic Braking (AB) which brakes automatically according to the objects detected and traveling speed.

ADAS are systems that use algorithms to increase the level of automation and provide better safety by eliminating human error [17]. As these systems evolve, Autonomous Vehicles (AV) become more of a reality. By the year 2050, according to Montgomery and Mudge *et al.* [20], there will be approximately \$796 billion in annual benefits from AVs. Congestion Mitigation prevention will generate \$71 billion, Accident Reduction (economic impact and quality of life improvements) \$503 billion, Reduced Oil Consumption \$58 billion, Value of time \$153 billion, and Reduction in taxi services cost \$10 billion.

However, to achieve this level of automation, ADAS has a long way to go. Weather conditions and tricky situations (for example, construction work on roads that affect the expected traffic behavior) significantly impact its performance. Therefore, developers test these systems both in-field and through simulation.

Even though real-world testing is essential to developing these systems, mainly because of the robustness of components and hardware [30], tests are costly, time-consuming, and some of them are impossible to perform for safety reasons.

Given this context, 3D Virtual Simulators play an essential role as a solution to some of these problems since they provide the scenarios needed (for instance, any weather condition), ensure safety by eliminating physical consequences, and reduce costs by cutting expenses that would go to infrastructures in the real world.

1.2 Problem

ADAS are designed to help the driver by providing comfort and, most importantly, active safety. However, there is a certain skepticism about trusting these systems. Understanding better how these systems work, what is actually happening, and the advantages of these systems improves trust [11].

Currently, common car drivers do not understand how these ADAS algorithms perform, especially the difference they make in certain human error-prone situations [8].

Do modern car drivers know what happens behind the scenes and how these systems impact their day-to-day life? Are the drivers aware of how ADAS can assist them in certain situations? And would their performance be the same in identical scenarios without the help of ADAS?

1.3 Hypothesis

Although real-world ADAS testing is crucial to its development and validation, simulation has become an essential tool for the automotive industry. It presents the possibility of recreating complex real-life scenarios. This capacity to build personalized situations with realism in a short amount of time with very few costs captivates the automotive industry.

After identifying that the average driver is unaware of how these ADAS algorithms work, let alone the difference they can make in specific scenarios when human error is likely to occur, it raises the need to develop a framework to solve this issue.

Therefore, the 3D driving simulator usage, in the Automotive Industry context, can enhance the communication of the ADAS algorithms benefits and how these systems act in the driving process.

1.4 Goals

The fundamental goal of this dissertation is to improve the way common drivers perceive ADAS and to highlight the role of these systems in potentially dangerous situations.

In order to achieve the designated main objective of developing a drivable experience in a 3D driving simulator and to test, validate and conclude the experiment's strengths and weaknesses, two other goals were defined.

For the first goal, there needs to be a simulator survey to determine which of the simulators available is the best option for what is intended. Understanding what kind of simulations are made in this field is also essential to get the most out of the experiment. Regarding the second goal, it is indispensable to construct a structured experiment with a primary objective, how many participants, the type of participants, and the type of tests.

1.5 Outline

In addition to the introduction, this dissertation contains five further chapters. In Chapter 2, the state-of-the-art in ADAS algorithm simulation and validation are explored in detail. Furthermore, Chapter 3 presents a solution outlook for the problem this dissertation is trying to solve. Moreover, Chapter 4 describes how the experiment will be conducted, and Chapter 5 goes through the obtained results. Finally, Chapter 6 analyzes the conclusions and future work.

Chapter 2

ADAS Algorithm Simulation and Validation

This chapter discusses state-of-the-art fundamental topics for this dissertation in detail. In Section 2.1, an introduction to the automotive industry is made to establish context and essential facts that affect the work to be done. Section 2.1.1 introduces automotive industry concepts critical to this dissertation, explaining the idea of automation, automation levels, and ADAS.

Section 2.1.2 addresses and explores some ADAS algorithms, and Section 2.1.3 presents ADAS common challenges to better grasp ADAS current problems and why they exist. Section 2.1.4 displays how these algorithms are tested in field and simulation.

Section 2.2 introduces simulation in general terms, while Section 2.2.1 explains virtual 3D simulators and how they are used. Furthermore, Section 2.3 presents simulation from an automotive industry perspective, covering some use cases, evaluating some simulators, and going into detail on the most indicated simulator for the goal of this dissertation. Section 2.4 addresses Human-Computer Interaction (HCI) methodology and how the proposed solution, being a human-computer interaction, should be evaluated.

Finally, Section 2.5 summarizes this chapter and highlights its key points.

2.1 Introduction to Automotive Industry

According to the European Commission, more than 90% of accidents are caused by human error, and there are still more than 40 000 deaths per year on the road [3].

In the Automotive Industry, there is a lot of focus on reducing this statistic by developing ADAS. As this technology advances, it is expected to help prevent accidents and potentially save many lives. Furthermore, it can be beneficial in other aspects:

- Environmental, due to its capacity to reduce emissions by optimizing driving behavior and decreasing traffic congestion [7].

- Mobility, due to the possibility of transporting people that do not have a license to drive or are incapable of doing so.
- Health, due to its capacity to reduce driving-related stress that weather conditions and heavy traffic can cause.

If widespread adoption is successful, the predicted yearly social benefits of ADAS will reach about \$800 billion by 2050 [20].

ADAS aids in the development of the automotive sector. The advantages it provides make it a must to continue to develop these systems.

Additionally, there are going to arise new opportunities related to productivity by reallocating the time spent driving. Thus, Mobility as a Service (MaaS) will be a new market to explore [9].

2.1.1 Concepts

ADAS

ADAS has become prominent in modern vehicles as an active safety system, intending to prevent accidents while also being crucial to AV development. These systems use sensors, such as cameras, LiDAR, Radar, and ultrasonic sensors, to understand the vehicle's surroundings and act accordingly.

According to Figure 2.1, modern vehicles have several sensors, including cameras, LiDAR, short and medium-range Radar, long-range Radar, and ultrasonic sensors. Each sensor provides unique characteristics that serve ADAS in different ways.

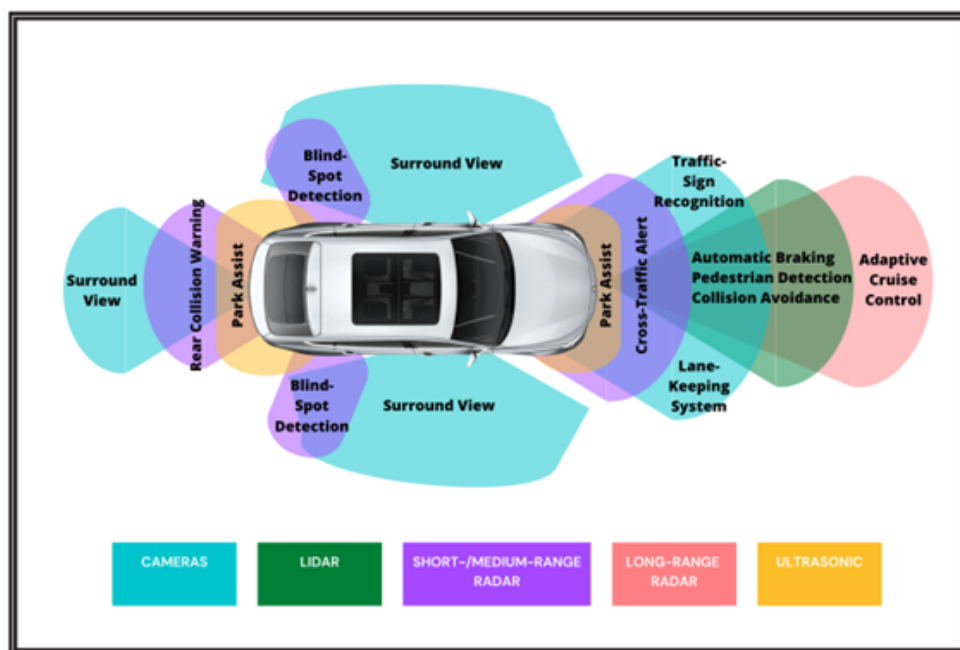


Figure 2.1: ADAS Sensors in a modern vehicle (based on [13]).

Cameras capture the environment, considering color, texture, and contrast, whereas other sensors cannot. There are often two types of cameras, monocular and stereo cameras. Monocular cameras have a single lens, with low image-processing requirements and low-depth information. This type of camera aims to detect obstacles, lanes, traffic signs, and pedestrians. On the other hand, stereo cameras, which use two or more lenses, are used for 3D information extraction and accurate depth estimation.

A modern car has cameras in the front, rear, and in the two side mirrors. With a combination of images gathered by each camera, it is possible to have a 360-degree surrounding view of the vehicle. With those images, the system recreates a bird's-eye view, helping the driver better understand the surroundings. This feature helps the driver park as shown in Figure 2.2.

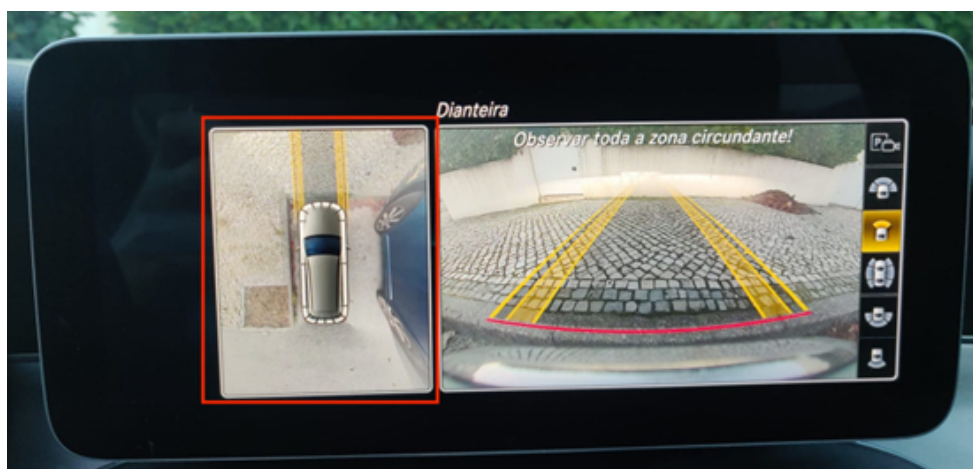


Figure 2.2: Bird's-eye View for parking.

LiDAR uses a laser beam to measure distances by calculating the time it takes for the laser to hit the object and come back to the sensor, and it is located on the front part of a car. The primary benefit of LiDAR, compared to cameras, is that it can reach longer ranges and be more precise in measuring distances. These features are useful for systems such as automatic braking, pedestrian detection, and collision avoidance [28].

Radar emits microwaves, and accordingly to the changes in the frequency of the reflected wave, it can determine the distance and speed of an object. As microwaves can travel further than optical light used in LiDAR, radar can detect objects at longer distances. Therefore, long-range radars are used for ACC, while short and medium-range radar is used for blind spot detection, collision warning, and cross-traffic alert. One significant advantage of radar over LiDAR is that rainy or foggy weather conditions do not affect the sensor's performance while also being less expensive [18].

Ultrasonic sensors are located in the front and rear of the vehicle and emit sound waves to detect the distance to a very close object accurately. It is mainly used for parking assist and automatic parking [13].

Table 2.1 compares the sensors' properties, strengths, and weaknesses in a modern car. Sensors can be active, meaning they do not require any external condition to function, such as radar,

LiDAR, and ultrasonic, while cameras are passive because they need external conditions like decent illumination to work correctly. Each sensor acts using different methods. Cameras operate by capturing images; radars emit microwaves; LiDAR uses a laser beam; and ultrasonic uses sound waves. Every sensor adds value differently since their individual strengths complement each other's weaknesses.

Table 2.1: Sensors' Comparison (based on [28]).

	Camera	Radar	LiDAR	Ultrasonic
Sensor Type	Passive	Active	Active	Active
Method	Vision	Microwaves	Laser beam	Sound waves
Strengths	Object classification	Works well in rainy and foggy conditions	Very accurate distance measurements	Very close distance measurements
Weaknesses	Works badly in rainy and foggy conditions	Object classification	Affected by rainy and foggy conditions	Only for small distances

Therefore, sensor fusion is essential for ADAS reliability [28], since it combines information from different types of sensors in order to accurately determine the state of the vehicle's environment [13].

Levels of Automation

According to Schöner [31], AVs are not just a continuation of assistance driving systems. Assistance systems are made to help the driver avoid human error accidents. However, to fully achieve AVs, the dependency on human input must end. According to the Society of Automotive Engineers (SAE) [27], the difference between driver support features and automated driving features is that the first requires that a human be actively driving, whereas the latter does not. SAE is the "leader in connecting and educating mobility professionals to enable safe, clean, and accessible mobility solutions" [26]. AVs are vehicles where ADAS are capable of performing all driving tasks in any condition. That is not possible yet. For that reason, SAE defined the levels of automation, presented in Table 2.2.

Table 2.2: SAE Levels of Automation (based on [27]).

LEVELS	DESCRIPTION	EXAMPLES
Level 0	No automation. There are only warnings.	Blind Spot Detection
Level 1	Assistance in steering or accelerating and braking.	Lane Centering or Adaptive Cruise Control
Level 2	Provide steering and braking/acceleration at the same time	Lane Centering with Adaptive Cruise Control
Level 3	Vehicle drives itself in certain specific conditions	Traffic Jam Assist
Level 4	Vehicles do not need any human input to drive themselves with certain conditions.	Driverless Taxis
Level 5	Full Automation. Vehicles can drive themselves in any condition.	Autonomous Vehicles communicate between them

In level 0, the driver handles everything manually, and there is no assistance. In level 1, ADAS assists the driver with either steering or accelerating/braking under certain situations with human input. In level 2, ADAS assists the driver with both steering and accelerating/braking under certain conditions with human input. In level 3, the vehicle drives itself in specific environments like the highway, but the driver may need to intervene. In level 4, the vehicle drives itself in certain environments, but there is no need for human input. Level 5 is full automation, the goal of autonomous navigation, where vehicles drive themselves in any condition without human input on any occasion. However, to achieve this level of automation there is still a long way to go.

Mercedes recently achieved Level 4 status, becoming the first automaker in the world which the German Federal Motor Transport Authority has authorized for public use. However, it may only be used for parking and under highly narrow circumstances [2]. At this moment, level 5 is not possible in any production vehicle. Nevertheless, level 3 is already present in some modern cars today. In level 3 vehicles, the driver must take control manually in certain situations. Therefore, there is still room for improvement concerning ADAS.

2.1.2 ADAS Algorithms

ADAS are comprised of various sensors, and each sensor is used for data collection to feed ADAS algorithms. These algorithms provide driver assistance functionalities. There are several algorithms, such as ACC, Automatic Braking, Pedestrian Detection, Lane Keeping System, and Traffic Sign Recognition (Figure 2.1). In this section, some of those algorithms are explored in depth.

Adaptive Cruise Control (ACC)

ACC controls the throttle and brake in order to adjust the vehicle's velocity and distance to vehicles ahead. Therefore, this algorithm relies on sensors like LiDAR that measure the distances and velocity of the vehicle in front. This system works with a velocity set by the driver between 40km/h up to 160 km/h. If the evaluation of distance and relative velocity of the car in front is determined safe by the algorithm, the vehicle maintains its velocity. Otherwise, it adjusts the throttle and brake to keep a safe distance. In all scenarios, ACC is overridable by the user at any time by pressing the brake or throttle.

Figure 2.3 demonstrates the ACC's architecture. According to Xiao and Gao [37], this architecture is split into four stages: Signal collecting (SC), Signal processing (SP), Signal actuating (SA), and Signal displaying (SD).

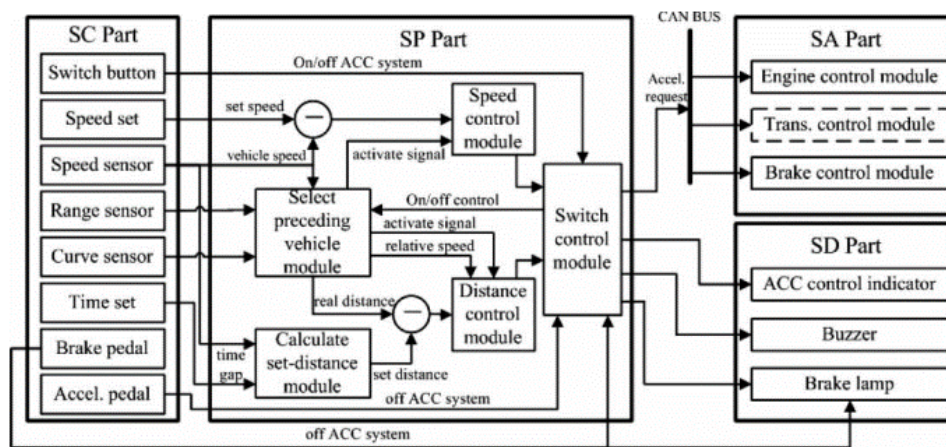


Figure 2.3: Adaptive Cruise Control architecture [37].

The driver can turn On and Off ACC with a button, which the switch control module detects. When the system is On, the range sensor starts to detect objects. If it does not see any object close enough, it activates the Speed control module, which later asks permission from the Engine control module to accelerate. If the Range sensor detects a close-by object or an object at a low speed, it activates Distance control modules, which triggers a request to the Brake control module in order to brake the vehicle. If the ACC system cannot brake sufficiently to avoid an object, it activates a buzzer in the display to let the driver know that he needs to take action and brake manually.

Consequently, if the driver presses the switch button to Off, presses the brake or throttle pedal, ACC is turned off immediately, and complete control is returned to the driver.

Lane Departure Warning System (LDWS)

A camera is mounted behind the vehicle's windshield in a vision-based LDWS to take pictures of the road. The lines on the road are interpreted, and lanes are recognized. A warning is sent to the driver each time the car inadvertently drifts out of its lane. This system is disabled if the turn signal is on.

Figure 2.4 represents a vision-based LWDS diagram, where the process starts with capturing images, pre-processing those. Following that, the system begins to line detect in order to identify lane markers.

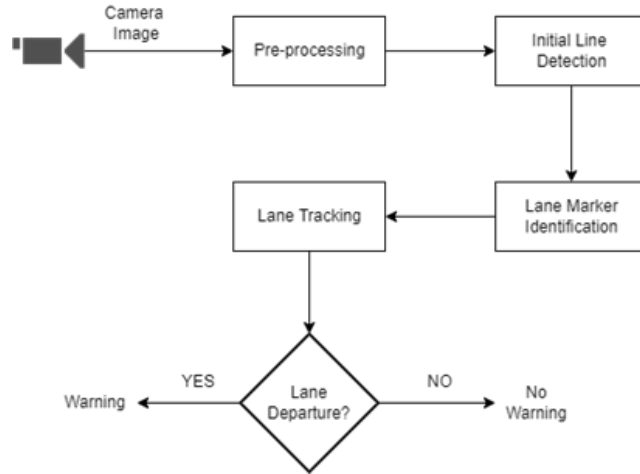


Figure 2.4: Lane Departure Warning System diagram (based on [14]).

Subsequently, those lanes are tracked to determine whether the vehicle is departing the lane. If the vehicle is, in fact, departing lane, then LWDS warns the driver.

This system is composed of Lane Detection and Tracking algorithms [14] and faces many challenges due to the variety of lanes and visibility issues seen in Figure 2.5. Weather conditions also affect the performance of these algorithms.



Figure 2.5: Examples of complex Lane Detection and Tracking conditions [19].

2.1.3 ADAS Common Problems and Challenges

There are several challenges regarding ADAS and autonomous navigation. One of the major problems in ADAS today is weather conditions' impact on the algorithms' performance. As mentioned in sub-section 2.1.1, cameras have difficulty in foggy and rainy conditions due to low visibility and inadequate contrast.

Another current issue in ADAS is the high power it consumes to run complex algorithms. Therefore, there is a need to minimize the power consumption of the hardware used for ADAS [29].

One more problem arises with the increased connectivity in modern vehicles, where the car receives and sends various information. Restricting access to hackers is a big security issue [12].

Recreating ADAS testing scenarios is a challenge, as they are costly, sometimes impossible to recreate in real life or too dangerous to replicate. For those reasons, simulators are critical to bypass these problems—the why and how are addressed in the following sections.

2.1.4 How ADAS Algorithms are Tested

European New Car Assessment Programme (Euro NCAP) is a program set to evaluate the performance of cars regarding safety by performing in-field tests. Euro NCAP defined essential test scenarios in order to rate cars' safety, taking several parameters into account.

With the growth of ADAS and its features, a stand-alone rating system for assisted driving systems was introduced by Euro NCAP in 2020 [22]. This new rating system focuses on two critical areas displayed in Figure 2.6: Assistance Competence and Safety Backup.

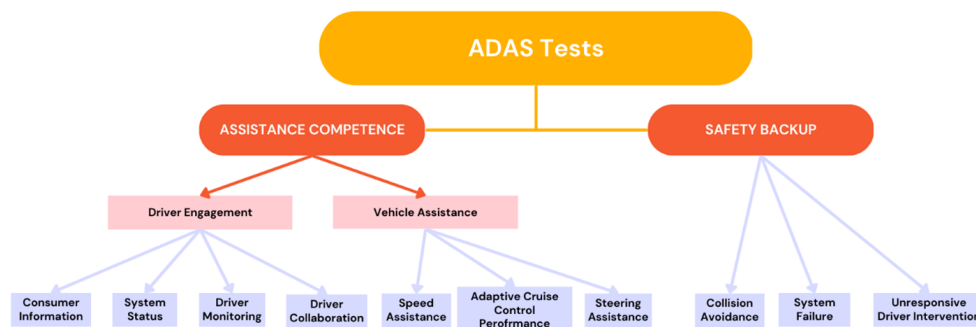


Figure 2.6: Euro NCAP ADAS Tests (based on [21]).

Assistance Competence is composed of Driver engagement, which are protocols that test how well the driver understands the system, its limitations, and if the system can cooperate with the driver, and Vehicle Assistance, which assesses how ADAS react to different scenarios.

Within Driver Engagement, there are four protocols tested: Consumer Information – takes into account the systems' names, making sure it is not misleading for the consumer –; System Status – tests systems status feedback –; Driver Monitoring – tests how well the system makes sure the driver is engaged –; and Driver Collaboration – evaluates the systems' cooperation with the driver in situations where he needs to correct a system's action, such as steering to avoid an obstacle (Figure 2.7).



Figure 2.7: Steering to Avoid Obstacle.

As for Vehicle Assistance, there are three protocols. One of those protocols is Speed Assistance, which evaluates if ACC speed adapts to conditions and traffic signs or allows the driver to adjust this speed manually. Another protocol is Adaptive Cruise Control Performance, which assesses if the vehicle's ACC adapts to critical events such as a vehicle cutting-in in front (Figure 2.8 on the left side) or approaching a stationary car in a curve (Figure 2.8 in center). Steering Assistance is the last protocol tested in Vehicle Assistance, which tests if the systems help the driver to keep the vehicle in lane (Figure 2.8 on the right side).

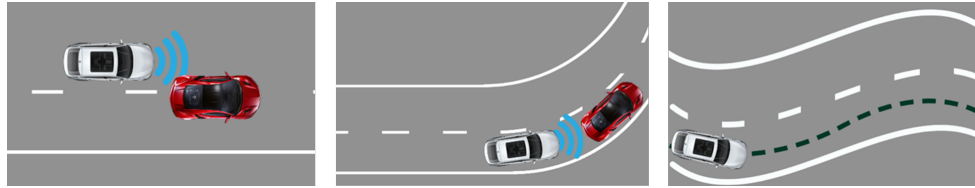


Figure 2.8: Examples of ACC critical events.

Safety Backup focuses on emergency situations and how well the system can respond. Therefore, Collision Avoidance is examined to determine if the system is capable of intervening in order to avoid a collision. System Failure is a protocol that tests occasions where the systems are disengaged and if it is perceptible to the driver. Unresponsive Driver Intervention tests how the vehicle reacts to an unresponsive driver that received a warning and has not acted on it.

Euro NCAP performs these tests in real life, which are indispensable for these systems. However, virtual simulation plays a significant role in refining these systems before being tested on the field.

The following sections will examine in-depth why simulations are made in the automotive industry and how ADAS benefit from virtual driving simulations.

2.2 Simulation

Simulation is an imitation of real-world processes supported by models [16]. These models can be reconfigured in order to suit a desired outcome better. A model is a depiction of the design and operation of a particular system of interest.

Simulations are used to support engineers' decisions, allowing them to evaluate performances [1]. Before making changes to a current system or creating a new one, simulation is used to lower the likelihood that performance will fall short of expectations, remove unforeseen bottlenecks, prevent resource under- or over-utilization, and improve system performance [1].

According to Law [16], you can classify simulation models in three different categories:

- Static or Dynamic – where Static simulation models occur in a particular time, so time is not an important factor, whereas Dynamic simulation models evolve over time, and time is a crucial factor.

- **Deterministic or Stochastic** – Deterministic models do not have probabilistic inputs such as random values. On the contrary, Stochastic models are models that deal with random inputs.
- **Continuous or Discrete** – Continuous models are models where the variables continuously change over time. Discrete simulation models have variables change at a particular moment in time instantaneously. An example of a continuous model would be simulating a car race, where the variables continuously change over the race. In contrast, if the simulation focuses on the moments when the car is stopped during a race, there would be only a few moments in the race where this occurs.

There is a diverse use for simulation, and it is applicable in many areas, such as analyzing supply chains and manufacturing systems [24], assessing organizations' performances (hospitals, restaurants, shopping centers) [35], testing and evaluating software and hardware's performance for transportation systems [4]. This dissertation focuses on the latter.

2.2.1 Virtual 3D Simulators

In order to improve the safety performance of ADAS, simulation is considered a good compliment.

Due to time, cost, and safety issues, simulating specific and unpredictable scenarios in real life is not viable. Virtual 3D Simulators are indispensable, considering they eliminate almost all expenses and safety issues and diminish tremendously the time factor [4].

Virtual 3D Simulators can provide validation of driver and vehicle interaction scenarios [31], mimicking the behaviors of the traffic participants, such as unpredictable behaviors like changes in speed, lane keeping of human drivers but also pedestrian behaviors like crossing the road on unexpected places.

These virtual environments replicate the real world, with weather conditions, shadows, buildings, and other objects obstructing the driver's view.

Simulators do not replace field tests in the real world as they contribute to more validation insight. According to Fremont and Kim *et al.* [6], simulation was used to predict whether a certain scenario would be safe or unsafe to go to track testing. 93.3% of safe simulated test cases led to safe behavior on the track, while 62.5% of dangerous test cases resulted in unsafe conduct on the track.

However, using this software tool is essential because it allows all kinds of scenarios, system configurations, driver characteristics, and environments. Furthermore, it is possible to run extensive tests in a short time with almost zero costs, contrary to real-world testing [4].

Thus, Virtual 3D Simulators are a crucial tool for improving ADAS.

2.3 Simulation in Automotive Industry

In the Automotive Industry, simulation is used in all sorts of tasks. These tasks include manufacturing, supply chains, and autonomous navigation.

Manufacturing has several stages: press, weld, paint, engine, and trim. All these manufacturing stages involve problems that are sensitive to analysis. An analysis that is achieved by simulation [36].

The simulation of supply chains is fundamental to the Automotive Industry due to its complexity. Understanding reactive behaviors of network plants to particular situations, such as increases or decreases of demand and technical problems, is one of the main perks of simulation to this sector in the industry [24].

This dissertation will focus on the usage of simulation for ADAS.

2.3.1 Virtual Simulation for ADAS

As new technologies in the automobile sector emerge, there is a growing need for active safety and driver-assistance features that improve driver comfort, convenience, and safety. These systems are leading the way to autonomous navigation.

Considering this, testing these technologies becomes more complex and sometimes impossible in real-life. Therefore, virtual simulation of vehicles under different weather and traffic conditions is mandatory. Virtual simulation does not replace physical on-road testing but is a vital part of the process. Since testing vehicles in the real world is very costly and time-consuming, it occurs at the latter end of the development phase when virtual testing has already occurred. Virtual simulation is beneficial because it provides a safe environment to test dangerous driving scenarios, where developers can test scenarios without the damage of system failures or faulty evaluations [4].

2.3.2 Simulator Technology

There are several 3D Driving simulators on the market, and choosing the right one for the tasks proposed is essential. For that reason, in this sub-section, the advantages and disadvantages of CarSim, Gazebo, MATLAB/Simulink, CARLA, and LGSVL simulators are analyzed.

CarSim

It is a simulator (Figure 2.9) commonly used by Original Equipment Manufacturer (OEM) companies¹. The main advantage of this simulator is that it does not require any other type of software to run a simulation. At the same time, it allows the use of external models with a standard interface for MATLAB/Simulink. It also has an extended documentation². The simulator includes up to 200 different locations and motion-independent objects such as vehicles, pedestrians, animals, buildings, and trees. Furthermore, it supports up to 99 ADAS sensors that detect moving objects, but for this use, it is necessary an extended license³.

¹<https://www.carsim.com/company/customers/carsim.php>,(accessed Nov.29,2022)

²<https://www.carsim.com/products/carsim/index.php>,(accessed Nov.29,2022)

³https://www.carsim.com/products/supporting/objects_sensors/index.php, (accessed Nov.29,2022)



Figure 2.9: CarSim Simulator.

Gazebo

Gazebo is an open-source, cross-platform simulator (Figure 2.10) as it supports MacOS, Linux, and Windows operating systems⁴. It provides a range of sensors necessary for ADAS algorithm development and testing. It has Dynamic Animation and Robotics Toolkit (DART) as a default physics engine, allowing a better level of accuracy than game engines. This level of accuracy is due to the full access this tool gives to internal kinetic and dynamic quantities, unlike other physics engines that view the simulator as a black box⁵. The fact that it is an open-source simulator is an advantage: a wide community support is available, and sharing and using models by other developers is an option. It also provides integration with Robot Operating System (ROS), an open-source robotics middleware suite [38].

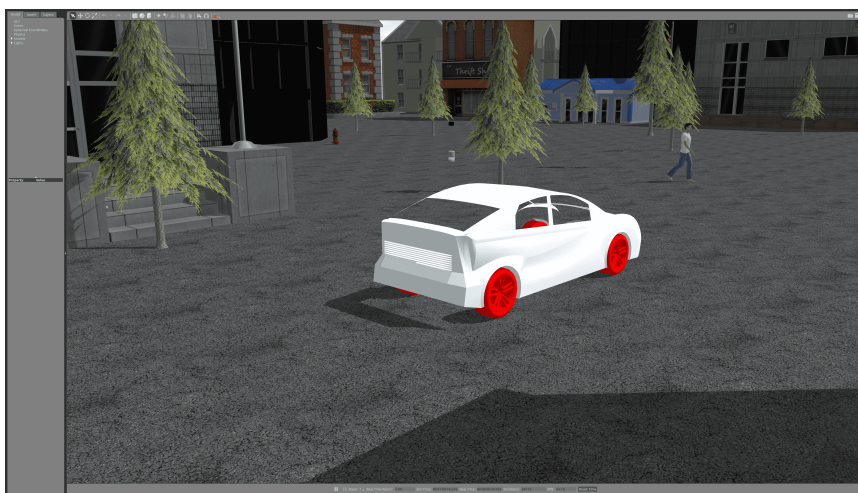


Figure 2.10: Gazebo Simulator.

⁴<https://gazebo.org/features>, (accessed Nov.30,2022)

⁵<https://dartsim.github.io/>, (accessed Nov.29,2022)

MATLAB/Simulink

MATLAB with Simulink (Figure 2.11) has an interface for ROS to access live and recorded driving data ⁶. To use geographic map data, it provides compatibility with HERE HD Live Maps, which offers precise HD map data ⁷ and OpenStreetMap that is an open-source map data provider ⁸. It offers two types of environments, cuboid and Unreal Engine, a game engine that allows the creation of realistic worlds ⁹. The cuboid environment uses simple graphics and probabilistic sensor models. Unreal Engine is used to develop perception algorithms. MATLAB also supplies an Automated Driving Toolbox to help developers with designing, simulating, and testing ADAS ¹⁰.

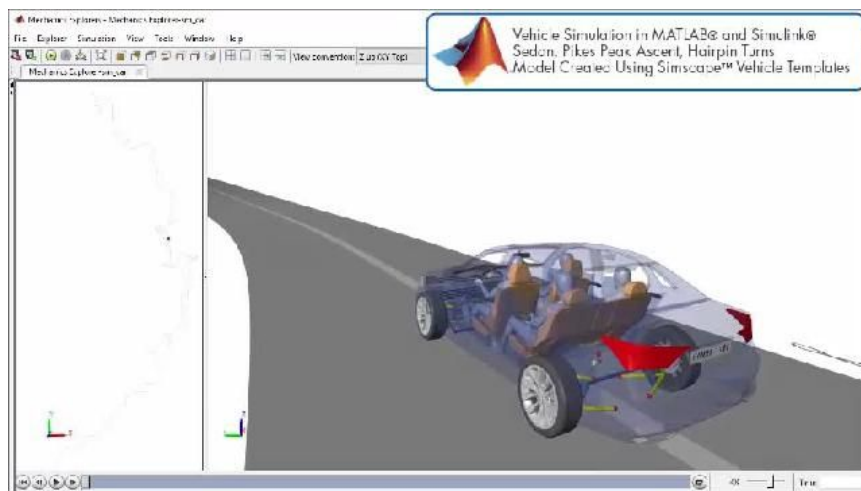


Figure 2.11: MATLAB/Simulink Simulator.

CARLA

CARLA is an open-source simulator (Figure 2.12) developed specifically for autonomous driving research and development. It is built over Unreal Engine 4 (UE4), a realistic game engine that delivers rendering quality, NPC logic, and realistic physics [5]. It is a modular tool with a robust Application Programming Interface (API) that fulfills ADAS requirements for tests and validation. It offers open assets, including buildings, vehicles, and towns, the ability to change weather conditions, and supports several sensors necessary for ADAS algorithms. In addition, CARLA is compatible with MacOS, Linux, and Windows and integrates with other software such as ROS and Autoware.

⁶<https://www.mathworks.com/solutions/automated-driving.html>, (accessed Nov.29,2022)

⁷<https://www.here.com/platform/HD-live-map>, (accessed Nov.29,2022)

⁸<https://www.openstreetmap.org/about>, (accessed Nov.29,2022)

⁹<https://www.unrealengine.com/en-US/unreal-engine-5>, (accessed Nov.30,2022)

¹⁰<https://www.mathworks.com/products/automated-driving.html#refap>,(accessed Nov.29,2022)

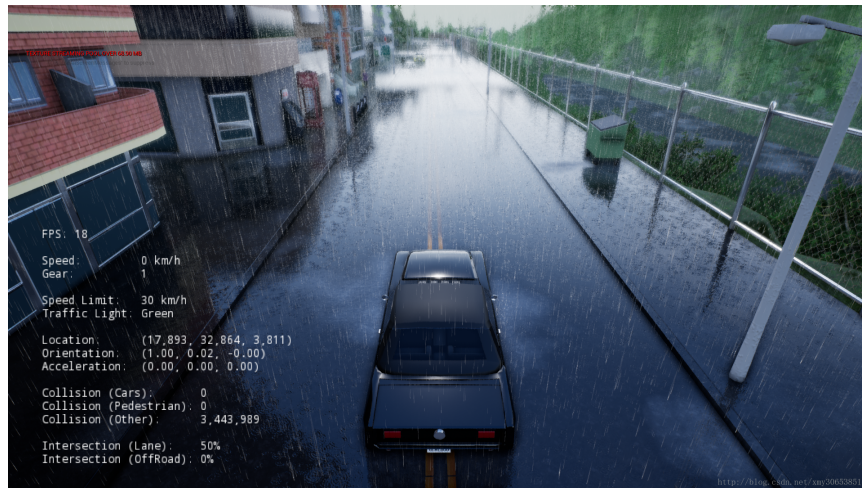


Figure 2.12: CARLA Simulator.

LGSVL

LGSVL is an open-source simulator (Figure 2.13) by LG that provides full-stack simulation, integration with ROS, Autoware, and Apollo, and easy sensor customization [25]. It runs on Unity Engine, a game engine that promotes real-time 3D development ¹¹. Although this simulator presents an amount of quality features to the development and testing of ADAS algorithms, LG decided to suspend active development of the simulator as of January 2022 ¹². The fact that there will not exist any bug fixes and support is a liability. It provided a website with models made by other users and the simulator documentation, but it will eventually be terminated. For that reason, this simulator is not an option.



Figure 2.13: LGSVL Simulator.

¹¹<https://unity.com/>, (accessed Nov.29,2022)

¹²<https://www.svl simulator.com/news/2022-01-20-svl-simulator-sunset/>,(accessed Nov.29,2022)

2.3.3 Comparison Between Simulators

To identify the appropriate simulator for the goal of this dissertation, certain requirements were defined as crucial. These conditions cover the architecture of autonomous vehicles and infrastructure to drive the simulated car, and were based on the studies of Kaur and Taghavi *et al* [10].

Perception

AVs depend on their ability to see and how they perceive the surrounding environment. That is why ADAS algorithms make use of a variety of sensors. One challenge in the perception of these vehicles is how they act in different weather conditions. To test that perception, simulators must have support to recreate these real-world scenarios and weather conditions.

Camera Calibration

Camera Calibration possibility is a positive feature because several algorithms depend on cameras, and their configuration is important to achieve the best performance.

Vehicle Control

ADAS algorithms aim to assist the driver and diminish human error in a time where level 5 automation is not yet possible. Therefore, the algorithms act on human input. For that reason, the chosen simulator must provide vehicle control, where the user can throttle, brake, and steer.

Scenario Simulation

Recreating realistic scenarios is mandatory in a simulator that aims to develop and test algorithms that will be applied in the real world. Creating scenarios that are hard in real life is fundamental to avoid these situations in the future. Moreover, trustworthy traffic simulations depend on various static and dynamic traffic agents and their behavior. Simulators with a flexible API that allows the user to configure behaviors of dynamic objects are imperative to create complex scenarios.

Integration with other software

The integration of the simulator with other software like ROS and Autoware is advantageous since these tools add value to the simulator by providing sensor data and object data.

Open-Source

Open-source simulators are favorable since they are free, have a big community that keeps adding value to the tool, and enable collaboration.

Portability

The ability to run the simulator in multiple operating systems is a plus. It decreases the limits to the user because the user is not required to have one specific operating system and increases the portability since there are fewer restrictions on where the simulator can be used.

According to Table 2.3, CARLA Simulator is the best choice for what is intended to develop. It is the most complete Simulator that was reviewed and presents all the desired features.

Table 2.3: Simulators comparison (based on [10]).

Requirements	CarSim	Gazebo	MATLAB/ Simulink	CARLA	LGSVL
Perception	N	N	N	Y	Y
Camera Calibration	N	N	Y	Y	N
Vehicle Control	Y	Y	Y	Y	Y
Scenario Simulation	Y	N	Y	Y	Y
Integration with other software	Y MATLAB/ Simulink	Y ROS	Y CarSim, ROS	Y Autoware, ROS	Y Autoware, ROS, Apollo
Open-Source	N	Y	N	Y	Y
Portability	Y	Y MacOS, Windows, Linux	Y	Y MacOS, Windows, Linux	Y Windows, Linux

From a perception perspective, it is able to support several types of realistic weather conditions. It offers camera calibration for ADAS algorithms that require that kind of control while offering vehicle control with realistic vehicle dynamics and physics. Moreover, it gives the possibility to

create specific traffic scenarios with a large amount of dynamic objects like pedestrians and vehicles. Besides that, it also provides integration with external software, such as ROS and Autoware, increasing the tool's possibilities. Additionally, it offers all the mentioned features freely in an open-source environment with compatibility with operating systems like MacOS, Windows, and Linux.

Hence, CARLA Simulator is the best option of the simulators evaluated, having all the required features and support with an extended API and active community.

2.3.4 CARLA Simulator

CARLA (Car Learning to Act) is an open-source simulator for urban driving. This simulator was built from scratch with the purpose of assisting in the training, development, and validation of autonomous driving models that cover both perception and control. CARLA provides urban environments created by design artists, including numerous vehicle types, buildings, pedestrians, and street signs. The simulation platform allows for a configurable sensor suite. It offers signals like GPS coordinates, speed, acceleration, and detailed data on collisions and other infractions that may be utilized to develop, train, and validate ADAS [5].

CARLA has the ability to specify the weather condition and time of day, as Figure 2.14 shows, which is a vital feature for ADAS development and testing. The simulator is implemented over Unreal Engine 4 (UE4), which provides render quality and realistic physics.



Figure 2.14: Different weather conditions and time of day [5].

Moreover, CARLA is a server-client system where the server runs the simulation, and the client API is implemented in Python and is in charge of coordinating the communication between the server and the autonomous agent via sockets [5].

In CARLA, there is a world and a client. The world is an object that represents the simulation. It serves as an abstract layer that houses the main methods to spawn actors and change the weather. The client is the module that makes requests to change anything happening in the simulation ¹³.

Actors are everything that has a role in the simulation, such as vehicles, pedestrians, sensors, and more.

CARLA maps contain a town's 3D model and its road definition. The road definition for a map is based on an OpenDRIVE file, a standard, annotated road definition format. There are several preset Town layouts, like Town 5 seen in Figure 2.15 with different characteristics, and there is also the possibility to customize these preset maps or even create a new one according to the needs of the simulation [33].

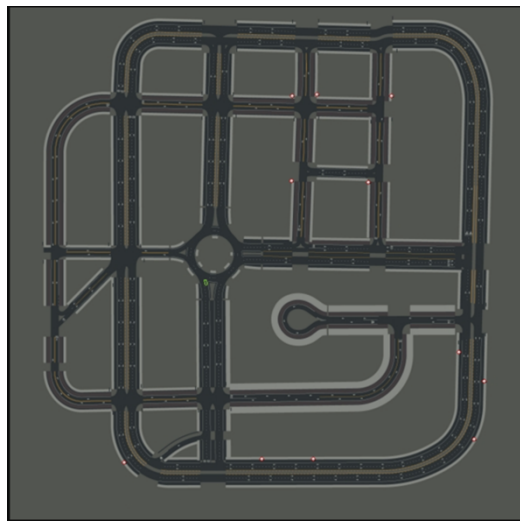


Figure 2.15: Town5 – Example of preset town [33].

Sensors are actors that wait for events to happen to gather data and help ADAS make decisions according to the data collected. CARLA provides a variety of sensors such as cameras (top left in Figure 2.16), radar (top right in Figure 2.16), and LiDAR (bottom in Figure 2.16) [34].

¹³https://carla.readthedocs.io/en/latest/core_concepts/, (accessed Dec.21,2022)



Figure 2.16: Example of some sensors' data [34].

Integration with ROS is another asset of CARLA, having two-way communication. ROS topics are translated for the CARLA server, and communications are passed between nodes in ROS and then converted into commands that can be used in CARLA. ROS integration adds value to the simulator because it provides sensor data, object data, control over vehicles through the steering, accelerating, and braking, and control over aspects of simulation such as pausing and playing simulation ¹⁴.

2.4 HCI Methodology Evaluation

Human-Computer Interaction (HCI) needs to be evaluated in order to provide the most significant experience to the user possible [23]. To do so, there are a few aspects to have in mind.

Firstly, it is imperative to define the goal of the tests to shape the experience according to what is intended to understand. Then, it is necessary to establish what resources are needed to make the tests possible. Resources such as location, availability, and duration are required to perform the tests, and equipment is indispensable for the experience, such as software and hardware.

Consequently, to conduct an effective experiment, the existence of an experiment protocol is recommended in order to guide the participant. This protocol usually comprises an introduction and the objectives of the tests. After clarifying the participant, providing an informed consent form is important. Subsequently, asking the participant to complete a pre-test questionnaire is a positive way to understand expectations and preconceived ideas. Afterward, several tasks are presented to the participant with the aim of retrieving data. In the end, the participant is asked to complete a post-test questionnaire and to answer a few questions in an interview [15]. Therefore, pilot tests

¹⁴<https://carla.readthedocs.io/projects/ros-bridge/en/latest/>, (accessed Dec.21,2022)

are suggested before collecting data in an effort to test the procedure. This gives an opportunity to tweak the duration, instructions, tasks, and questionnaires.

Furthermore, it is essential to choose the participants wisely for better data extraction. Participants should have different genders and ages if the product has no restrictions, and they should also represent target users. Ten to twenty people should participate to meet the need for a large enough sample for a richer data collection.

Tests can either assess usability problems with interviews, questionnaires, and direct observation or consider performance measures with A/B tests [39] for example. A/B tests can be either between participants (left in Figure 2.17) or within-participants (right in Figure 2.17).

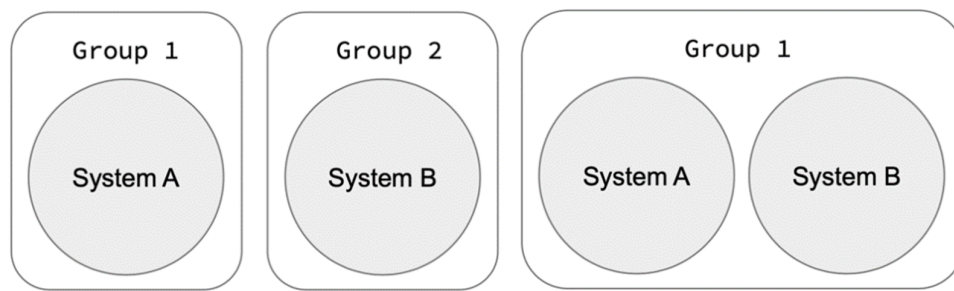


Figure 2.17: A/B Tests.

The problem with A/B tests, where the participant experiences both systems A and B, is the fact that it becomes a biased experience on system B because of what was previously experienced on system A. Therefore, participants should experience systems in a random order to prevent this.

After determining the goals of the experience, the resources needed, the type and number of participants, tasks must be adequately defined. Tasks must be specific, impartial, and equal for every participant. An example of an incorrect task would be to ask the participant to drive from point A to point B in the simulator, avoiding pedestrians and being aware of weather conditions. This task would alert and affect the participant's behavior. The correct task would be just asking the participant to drive from point A to point B. This way, it leaves it up to the participant the "how to do", providing them with just the "what to do".

When it comes to data collection, there are several data types. Objective data is not biased by the participant and it includes the time taken to perform a task and the errors committed. In contrast, subjective data depends on the user's preferences and opinions. Besides that, quantitative data is measurable and specific - an example would be asking "How many errors were committed?" - while qualitative data focuses more on the participant's personal feedback.

Usability can be measured in many different ways, depending on the goal of the experience. Some usability measures are the time it takes to complete a specific task, the number of mistakes made, the number of clicks, or user satisfaction. There are standard usability questionnaires available to provide both after a test, such as System Usability Scale (SUS) and User Experience Questionnaire (UEQ), and after a task, like Single Ease Question (SEQ) and After-Scenario Questionnaire (ASQ).

2.5 Summary

The automotive industry is currently developing SAE levels 3 and 4 of automation with the help of ADAS. Algorithms such as ACC and LDWS keep improving active safety in vehicles. However, developing these systems requires a lot of testing and validation. The process of real-life testing involves high expenses, amount of time, and safety measures. Hence the use of simulators to solve these challenges.

With the rise of ADAS, Euro NCAP recently created specific test scenarios for these systems to guarantee user safety and ensure that these features do not give wrong impressions to the driver (claiming to be autonomous and enticing the driver not to pay attention to what is happening).

Virtual simulators are essential for developing, testing, and validating ADAS, due to their capacity to recreate scenarios that would be expensive, dangerous, and sometimes impossible in real life. Driving simulators are now indispensable to the automobile industry, although they do not completely replace on-the-ground simulations.

There are many simulators available on the market that can carry out this task. However, the CARLA simulator is the best simulator for the purposes of this dissertation as it provides render quality, realism, and a lot of control over the simulation.

Moreover, to achieve the best result using this simulator, the experiences to be done should be planned ahead of time. The test's goals, tasks, and questionnaires need to be defined. Participants should represent the target user by having a range of ages and genders and should number at least ten people.

Chapter 3

Human Driver VS ADAS Simulation Framework

In this chapter, the solution is addressed. Section 3.1 explains the context of the possible solution, while Section 3.2 explores the details of the solution and its requirements. Additionally, Section 3.3 presents the prototype's structure and a full explanation of the functionalities and how it works. Finally, Section 3.4 covers the conclusions from this chapter.

3.1 Context and Problem

ADAS algorithms and AVs are being developed and improved to take the transportation sector toward a new level of excellence and innovation. The primary goal is to eliminate human error from the equation by relying on advanced technologies and algorithms to handle critical driving tasks. Reducing human involvement can significantly minimize the risk of errors caused by factors such as fatigue, distraction, or misjudgment.

Furthermore, the emphasis is placed on providing enhanced comfort and safety for passengers and road users. By implementing sophisticated sensors, intelligent control systems, and real-time data analysis, ADAS, and AV technologies aim to create a safer and more enjoyable travel experience.

In addition to safety and comfort, improving efficiency is a key aspect of ADAS algorithms and AV development. These technologies strive to optimize traffic flow, reduce congestion, and enhance fuel efficiency by leveraging precise navigation and intelligent routing. By making transportation more efficient, resources can be utilized more effectively, reducing overall travel time, fuel consumption, and environmental impact.

With this in mind, ADAS algorithms still have their flaws and a long way to go to be fully autonomous, but their current state is already an incredible help to human driving in critical situations. There is already automatic parking, emergency braking, lane changing detector, and also systems that can understand if the driver is too tired to drive or not, with the goal to warn the driver

of the risk and advise some rest. All of these features are made to coexist with the human driver and improve their driving, but the final objective is to free the human from the driving activity.

Therefore, even though modern cars already come with these ADAS systems to improve road safety, many drivers still need to learn about these systems' existence and potential benefits.

With this problem in mind came the idea to communicate these benefits in an interactive and personal way to answer the questions previously presented in Section 1.2.

It is intended to allow the common driver to understand what is happening in their vehicles and the impact these algorithms have on their day-to-day life.

Therefore, in order to demonstrate it, the goal is to design and develop a framework that puts the user to the test against a driving algorithm. By making the user drive the same scenarios as the algorithm, it is possible to compare the performances and present those results to the user. This chapter explains how this can be achieved.

3.2 Solution Outlook

The solution starts by providing a simulation that allows the user to drive (with steering wheel and pedals) in some unpredictable scenarios with day-to-day situations such as pedestrians crossing the road in unexpected places with low visibility, cyclists crossing the road wrongfully, red lights, intersections, stop signs, and traffic interactions.

Therefore, choosing or creating appropriate scenarios that challenge both the participants and the AV algorithm is necessary. It was decided that two different scenarios would be ideal. One easier scenario for a fast acquaintance of the driving style and environment for the participant and a more complex scenario with different critical situations. Even though more scenarios would bring an even wider variety of events and results, it would prolong the necessary time for a single experiment and diminish the number of participants. Therefore, a compromise was made that allowed a balance between different scenarios and experiment time (number of different participants).

This solution also involves a demonstration of what an AV algorithm with ADAS features would do in those same scenarios and critical situations. This demonstration, after the participant's driving, allows not only a direct comparison between behaviors during the simulation but also a better understanding of these algorithms. To allow this better understanding of the algorithm, there needs to be a visual explanation of the algorithm's sensors during the demonstration.

The goal is not to have a perfect algorithm that completes the scenarios with zero mistakes, as this could induce the participants in error by making it look that these algorithms are flawless. For that reason, the chosen algorithm must have a good performance in the chosen scenarios but also have some typical flaws (for example, perception limitation due to weather conditions, shadows, nighttime, and occlusions).

After the participant's driving and algorithm demonstration, it is intended to provide a clear dashboard that allows a straightforward interpretation of the results and provides a visual comparison between the user and the algorithm performances, a final score, and committed infractions graphs.

3.2.1 Requirements

In order to achieve the proposed solution, it is necessary to have a few scenarios that put the user and also the chosen algorithm to the test. For that reason, these scenarios present multiple traffic situations, such as lane merging, lane changing, negotiations at traffic intersections, handling traffic lights and traffic signs, and coping with pedestrians, cyclists, and other elements.

The first scenario (Scenario 1) (Figure 3.1) is in CARLA's Town1, and it presents a path where inner-city traffic plays a significant role, with several traffic lights, pedestrians, and cyclists crossing.



Figure 3.1: Scenario 1 Overview and Path.

Whereas Scenario 2 (Figure 3.2) is in CARLA's Town3, focusing more on traffic intersections with traffic lights, stop signs, and lane merging, but also with pedestrians and cyclists to cope with.



Figure 3.2: Scenario 2 Overview and Path.

Another crucial element of the experiment is the chosen algorithm because it must have a good performance and showcase a typical flaw. Therefore, for this experiment, it was chosen an AV algorithm that was used in CARLA Autonomous Driving Leaderboard, a competition that is meant to evaluate the driving proficiency of autonomous agents in realistic traffic scenarios.

Thus, the algorithms created for this competition guaranteed that they would perform in these scenarios. Therefore, the Interpretable Sensor Fusion Transformer (InterFuser) algorithm by the InterFuser team was chosen. It came second in the competition¹, but it was the best algorithm with the code available². Therefore, this algorithm fulfilled the requirements of having an outstanding performance in the chosen scenarios but not perfect. It was open-source, well-documented, and implemented to be used in the CARLA simulator, allowing seamless integration into the solution.

According to the InterFuser team [32], the InterFuser algorithm is an algorithm where "information from multi-modal multi-view sensors is fused, and driving safety is enhanced by providing intermediate interpretable features as safety constraint heuristics".

This algorithm uses multiple sensors, such as LiDAR and cameras with different views (left, front, right, and focus) for scene comprehension. Figure 3.3 visually demonstrates the combination of sensor fusion.

¹<https://leaderboard.carla.org/leaderboard/>,(accessed Jan.12,2023)

²<https://github.com/pendilab/InterFuser>,(accessed Jan.12,2023)

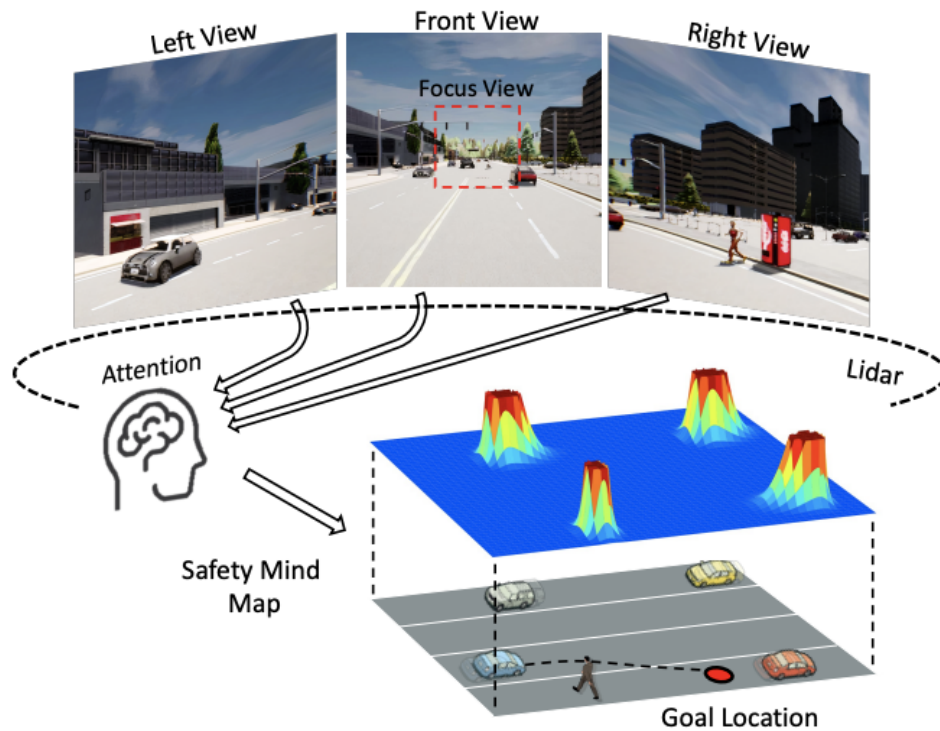


Figure 3.3: Comprehensive Scene Understanding by Sensor Fusion [32].

Regarding the InterFuser algorithm, one of the reasons it was chosen (other than the previously mentioned) was due to the ability to provide a visualization of the sensors being used that enable the performance of the algorithm, like the three extra camera views (right, focus, left) on top of the figure 3.4, the probability of finding a stop sign or a red light in blue in the bottom of the figure, object detection in vehicle's path and future prediction on the right side of the figure.



Figure 3.4: Algorithm's Views and Sensors.

The presentation of the results and KPIs of the simulations is another essential aspect of the solution. In order to build a dashboard that compares the human and algorithm performances visually, the goal is to record the frames where the critical situations occur (red lights, stop signs,

pedestrian and cyclist road crossings), generate GIFs with those frames, and visually compare, one by one, the specific situation side by side with a brief description of the situation below. Providing this visual comparison side-by-side allows the participants to immediately identify the differences between their own behaviors and the InterFuser algorithm. This visual comparison is important because it rules out the possibility of the participants forgetting what they have done and confirms the statistical data presented.

Data graphs will also be provided to compare both performances focusing on the evaluation metrics. Figure 3.5 presents a mock-up of that dashboard and the next section will explain in detail the prototype for this experiment.

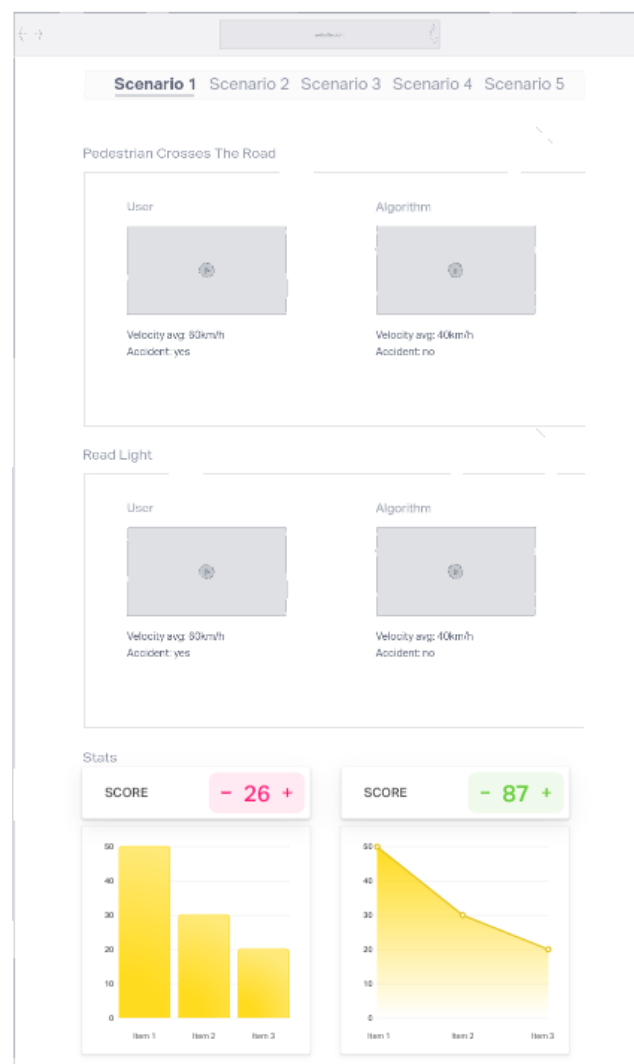


Figure 3.5: Dashboard Mock-up.

3.3 Prototype

This prototype was built using CARLA 0.9.10. This version was chosen because it was already used in Continental Engineering Services Porto.

The prototype consists of three main components represented in Figure 3.6:

- **Interface** (Python)
- **Simulation** (CARLA Simulator)
- **Dashboard** (React.js)

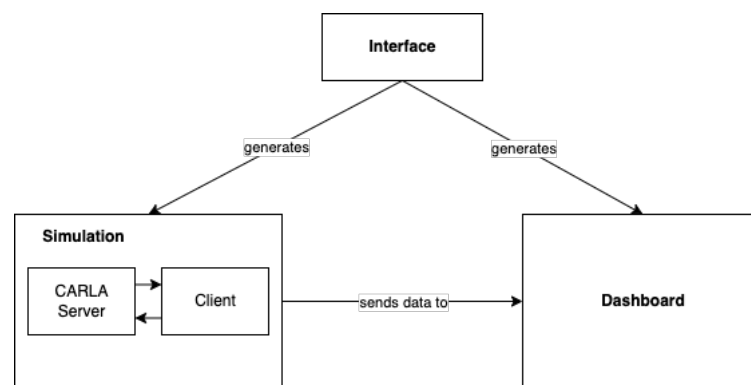


Figure 3.6: Prototype Main Components and Relation.

A component was necessary to streamline and automate the processes involved in controlling the experiment. Therefore, it was developed an Interface, using tkinter³, a python package that allows creating fast GUIs. This interface generates the simulations that involved running the CARLA server and client.

The CARLA server, which hosts the simulation environment, needs to be launched to run the simulations. Then the CARLA Leaderboard module must be launched, which is responsible for starting and monitoring the simulation, running the defined scenarios, and collecting evaluation metrics. Within this module, a Python script acts as the client, connects to the server, and allows manual vehicle control with a Logitech G29 Steering Wheel and Pedals in the defined scenarios. This module is also in charge of storing the simulation data in a JSON file.

After running the simulations, the interface component is used again to generate the dashboard. This dashboard, a web app developed in React.js using Material UI⁴ library components, receives JSON data about the KPIs and presents those KPIs clearly.

The following subsection presents and explains a detailed architecture of the prototype.

³<https://docs.python.org/3/library/tkinter.html>, (accessed Feb.10,2023)

⁴<https://mui.com/>, (accessed Feb.10,2023)

3.3.1 Architecture

The components mentioned above and their interactions result in the final architecture for the prototype presented in Figure 3.7.

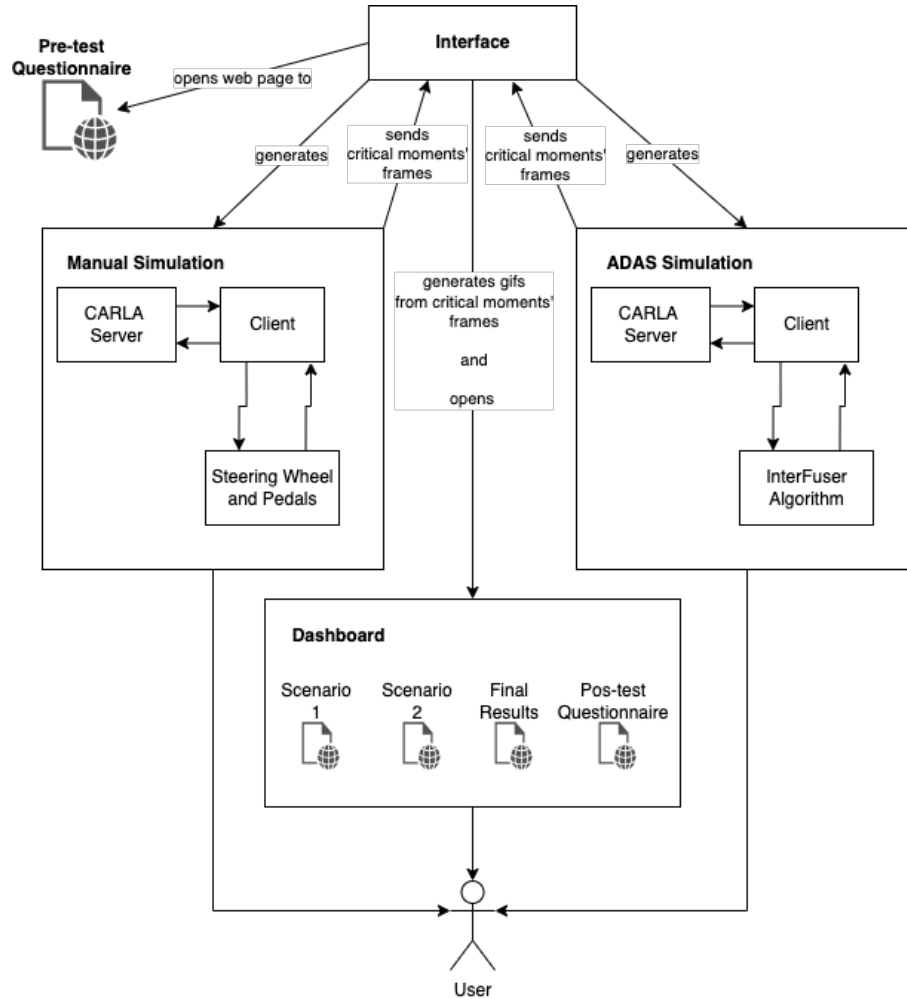


Figure 3.7: Prototype Architecture.

The final architecture puts the Interface Component in control of the whole experiment since it automates every part of it. It is the experiment control center for the supervisor.

From the Interface Component, it is possible to open the web page directly for the pre-test questionnaire, which is the first thing the participant must do. Subsequently, this interface can run the manual simulation for the participant to drive, where it opens the CARLA Server and runs the client with the Logitech G29 Steering wheel and pedals integration. While the participant is driving, frames are captured in critical moments, such as pedestrian crossings and traffic lights, so those frames can be converted into GIFs to be displayed later on the dashboard.

The same thing occurs for the InterFuser algorithm simulation, although this was only run once and converted into a video format due to performance issues. The algorithm required a series

of sensors that slowed the CARLA simulator, and it was not viable to show this simulation every time in real-time.

Subsequently, after the simulations, it is possible to generate the dashboard via the interface, which is composed of four web pages: two scenario pages with the GIFs of the critical moments and with the score performances and graphs, the final results page that provided a final evaluation of the whole experiment and a page with a google form integrated page for the pos-test questionnaire.

The following sections provide detailed explanations of these components.

3.3.2 Interface Component

A python interface (Figure 3.8) was developed to quickly run scripts that automate processes like running the CARLA server, running the user's simulation, running the algorithm's simulation, and opening the dashboard with the simulations' results.

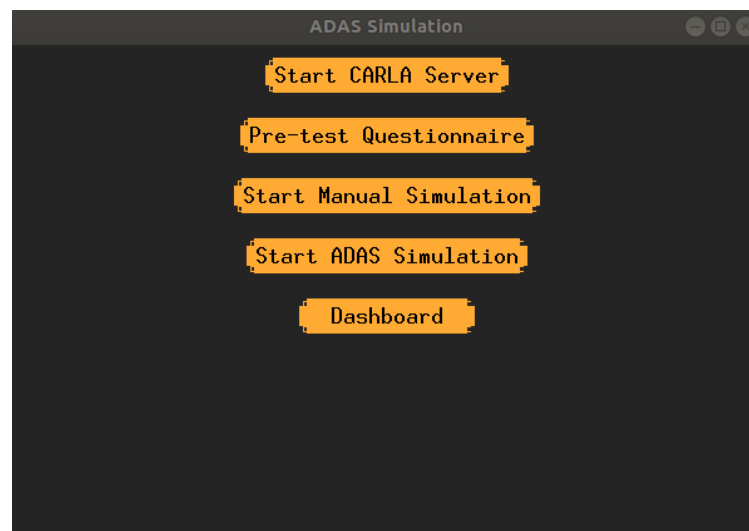


Figure 3.8: Developer Commands Interface.

While a button for the “CARLA server” was helpful for development, both buttons for running “manual simulation” and “ADAS simulation”, run the CARLA server and the simulation itself, so there is no need to press both buttons to get the simulations going.

The "Pre-test Questionnaire" button opens a Google form to question the users about demographic data before the experiment.

The Dashboard button is to be pressed by the end of the simulations to generate the web page with dashboard results. But before it generates the page, it searches for frames stored by the simulation of critical moments and generates GIFs with those frames to be presented in the Critical Cards of the dashboard as media.

3.3.3 Simulation Component

The Simulation component contains the CARLA Leaderboard package. Inside that package, there is a scenario runner folder, which is a CARLA simulator module that enables the development and execution of traffic scenarios, and a leaderboard folder that manages the human agent and the autonomous agent and assesses its performances in various traffic scenarios over a variety of routes. After each simulation, a JSON file is generated with the agent's performance in the scenarios.

The CARLA Leaderboard package did not have an integration with the steering wheel and pedals. Therefore, being a crucial part of the experiment for reliable results, a script was developed for the Logitech G29 Steering wheel and pedals integration (Figure 3.9).



Figure 3.9: Logitech G29 Steering Wheel and Pedals.

The data captured by the RGB camera sensor of the vehicle during simulations was saved in order to display those images on the dashboard later. This sensor sends frames to pygame, a Python gaming library – that is how the user sees the simulation in real time. Even though this sensor could get all the frames from the simulation, it was only necessary to save the frames from the defined critical moments. Time fragments could not determine these critical moments because that would change from simulation to simulation. Therefore, it had to be from waypoints of the vehicle's location in the scenario.

Consequently, an interval of waypoints was defined for each critical moment, and as long as the vehicle is between this interval, the simulation saves the frames.

3.3.4 Dashboard Component

The Dashboard component is a web app developed in React.js, and the goal is to show the results of the user's performance versus the autonomous vehicle's algorithm performance, providing statistics with clean charts using a chart's library Canvasjs and also visual comparisons of crucial moments. The entire dashboard is displayed in Appendix A.

The dashboard was designed in modules to be reusable with different algorithms and expandable in case new metrics are developed in future works. Therefore, this dashboard is composed of several components.

- **Tabs** – It is possible to add as many tabs necessary for scenarios (one tab per scenario). In this experiment's case, there are four tabs. Tab One and Two display the performance of their respective scenarios, while the third tab provides an overall view of the final results, with the average score of both scenarios and all of the infractions committed. The fourth tab is dedicated to a Google Forms questionnaire embedded in the web app for post-experiment feedback.
- **CriticalCard(title,description,media)** – Displays a media comparison between the user and the algorithm side by side from a single critical moment in the scenario, where the title, description, and media can be changed according to Table 3.1.

Table 3.1: Critical Card prop description and type.

Prop	Description	Type
title	The title of the critical card.	string
descriptionUser	Critical moment user's description.	string
descriptionAlgo	Critical moment algorithm's description.	string
mediaUser	Media to be presented in User card. Media components: video, audio, picture, iframe, img	string
mediaAlgo	Media to be presented in Alogrithm card. Media components: video, audio, picture, iframe, img	string

In this example (Figure 3.10), the critical moment is a pedestrian that crosses the road unexpectedly behind an object, so the reaction time is short. In this case, the user hit the pedestrian while the algorithm avoided it.

2 - Pedestrian Crosses The Road



Figure 3.10: Critical Card Example.

- **ScoreCard**(score) – Presents a percentage score comparison between the user and the algorithm side by side, where the score can be changed according to Table 3.2.

Table 3.2: Score Card prop description and type.

Prop	Description	Type
scoreUser	The score for the user.	number
scoreAlgo	The score for the algorithm.	number

The score card (Figure 3.11) has the lowest score in red color, while the highest score is in green. The left side belongs to the user, and the right to the algorithm.



Figure 3.11: Score Card Example.

- **InfractionCounterCard**(title, infractions) – Counts the number of infractions and displays it, where the title and the number of infractions can be changed according to Table 3.3.

Table 3.3: Infractions Counter Card prop description and type.

Prop	Description	Type
title	The title of the infractions card.	string
infractionsUser	Number of user's infractions.	number
infractionsAlgo	Number of algorithm's infractions.	number

The infraction counter card (Figure 3.12) presents the number of infractions in green color if they are zero, while if the infractions are bigger than zero, they are presented in red.



Figure 3.12: Infraction Counter Card Example.

- **FractionCard**(title, fraction) – Displays a fraction to present how many specific infractions were committed in the total possible infractions to be committed, where the title and fraction can be changed according to Table 3.4.

Table 3.4: Fraction Card prop description and type.

Prop	Description	Type
title	The title of the critical card.	string
fractionUserNumerator	Numerator user's fraction.	number
fractionUserDenominator	Denominator user's fraction.	number
fractionAlgoNumerator	Numerator algo's fraction.	number
fractionAlgoDenominator	Denominator algo's fraction.	number

For example, in a scenario where there are two stop signs, and the user runs one of them, it will display 1/2 in red color. If the user stops at all stop signs, it will display 2/2 in green color. This example is represented in Figure 3.13



Figure 3.13: Fraction Card Example.

- **GraphsCard**(graph) – Displays any type of graphs side by side, where the graph can be changed according to Table 3.5.

Table 3.5: Graphs Card prop description and type.

Prop	Description	Type
graphUser	The type of graph for the user.	json
graphAlgo	The type of graph for the algorithm.	json

The GraphsCard component was developed with two different graph components, using Canvasjs for the graphs, that feed of JSON files:

- **CollisionBarChart**(title,pedestrians,vehicles,obstacles) – A Bar Chart for all types of collisions that occurred in the simulation, where the title, pedestrians, vehicles, and obstacles can be changed according to Table 3.6.

Table 3.6: Collision Bar Chart prop description and type.

Prop	Description	Type
title	The title of the bar chart.	string
pedestrians	Number of pedestrian collisions.	number
vehicles	Number of vehicles collisions.	number
obstacles	Number of obstacles collisions.	number

In the example of Figure 3.14, the user collided with 2 pedestrians, 7 vehicles, and 0 obstacles, while the algorithms did not collide.

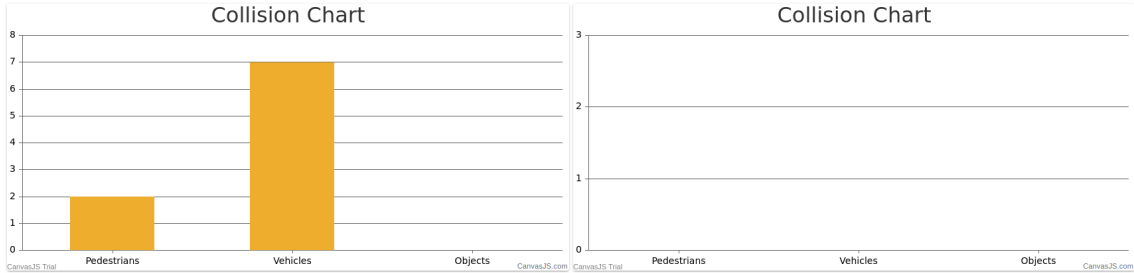


Figure 3.14: Collision Bar Chart.

- **RouteDoughnutChart**(title,completed) – A Doughnut Chart for route completion, where the title and percentage completed can be changed according to Table 3.7.

Table 3.7: Route Completion Doughnut Chart prop description and type.

Prop	Description	Type
title	The title of the doughnut chart.	string
completed	The percentage of the route completion.	number

In the example of Figure 3.15, the user completed 100% of the route. Therefore, it shows the doughnut completely green, while the algorithm completed only 51% of the route, showing the rest of the uncompleted route percentage in red.

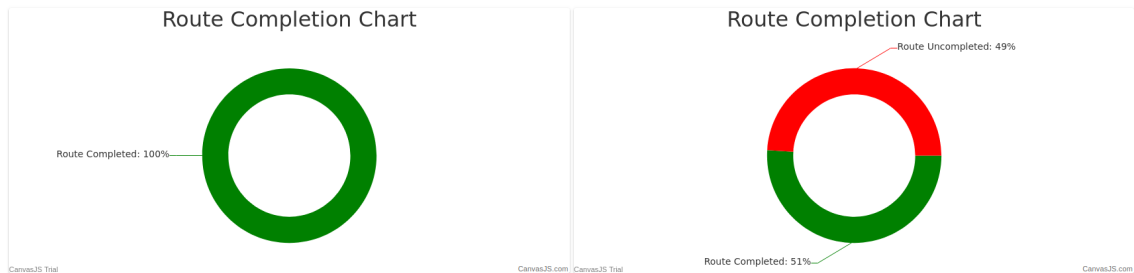


Figure 3.15: Route Doughnut Chart.

3.4 Summary

This chapter introduces a solution outlook for a prototype that aims to give common drivers a better look at ADAS systems and their impact on their day-to-day life.

Furthermore, it explains that the experiment will have two driving scenarios with critical moments, where the user will drive with a steering wheel and pedals. The same scenarios will be driven by the InterFuser Algorithm from CARLA Autonomous Driving Leaderboard, a competition that assesses these algorithms' performances and efficiency.

In this chapter, the scenarios used are presented and explained, as well as the InterFuser algorithm. It also details the prototype architecture required by the experiment.

Moreover, this chapter provides a detailed explanation of the prototype and its components: Interface, Simulation, and Dashboard, providing a detailed architecture. The Interface component deals with the automation of processes such as opening the pre-test questionnaire, generating the simulations, and opening the dashboard, all with a click of a button. The Simulation component is in charge of providing the simulations, both for the user and the algorithm, in the same scenarios. The Dashboard component presents the results of each simulation side by side to compare the performances of the user and algorithm.

Chapter 4

Experiment

This chapter focuses on the experiment protocol. Section 4.1.1 highlights the goals, while Section 4.1.2 explains the general process. In Section 4.1.3, the participant population is determined, and Section 4.1.4 determines the equipment and system requirements to conduct this experiment. Section 4.1.7 details the participant tasks during the experiment. The metrics used in the experiment are presented in Section 4.1.8 while Section 4.2 explains the test environment and Section 4.3 presents user feedback. Finally, Section 4.4 summarizes the whole chapter.

4.1 Test Protocol

A test protocol is a complete set of rules that detail how to control the experiment, to have the experiment recreated and tested. The test protocol includes a clear explanation of the experiment's purpose, instructions on how to recreate it, the equipment necessary, and the analysis and exhibition of the data collected.

Therefore, test protocols are essential for experiment tests as they establish the norms and procedures to achieve reliable results.

4.1.1 Goals

The goal of this test is to assess a solution to improve the communication of ADAS' algorithms benefits, in our day-to-day vehicles, by providing a clear comparison between the performance of a driver without the aid of these algorithms and the performance of an AV with ADAS features.

This experiment allows participants to witness the advantages of ADAS systems firsthand by observing the difference between their mistakes and how ADAS prevents them through the simulation. Rather than relying on generic examples, participants can directly experience how ADAS would have averted their errors and understand the benefits more effectively.

4.1.2 General Process

The experiment is introduced to the participants, providing them with the context and goals of the experiment. Then, the participants fill out an Informed Consent Form to get to know their rights as a participant, to know what kind of data is collected from them, who they can contact if they have any doubts, and to confirm that they consent to participate in the experiment.

The person leading this experiment commands it via the interface (explained in Section 3.3.2), where it opens the pre-test questionnaire for the participant, generates the simulations for the participant's tasks, opens the dashboard after the simulations for the participant to observe and to fill the post-test questionnaire integrated into the dashboard.

Before the simulation starts, there are a few rules to be explained to the participants (some of which can be considered limitations as compared to a real-life car interface):

- The car driven is automatic, so gears are not a factor.
- The participant must accelerate and brake only with the right foot.
- The right pedal is the accelerator, and the middle pedal is the brake.
- The speed limit is not an evaluated factor as there is no speedometer.
- The throttle pedal is not as sensible as a real one, so the user must put the foot all the way down to accelerate (the car accelerates progressively) and all the way up if the participant wants to decrease acceleration.
- The scenarios tested are in the United States of America:
 - Stop Signs do not need a physical sign; they can only be signaled on the road.
 - Traffic lights are on the other side of the intersection, and for an easier understanding, a picture should be shown explaining this (Figure 4.1). In this example, the car needs to go left, and it can go because the traffic light in front of him is green. The traffic light on the left side is to be disregarded.

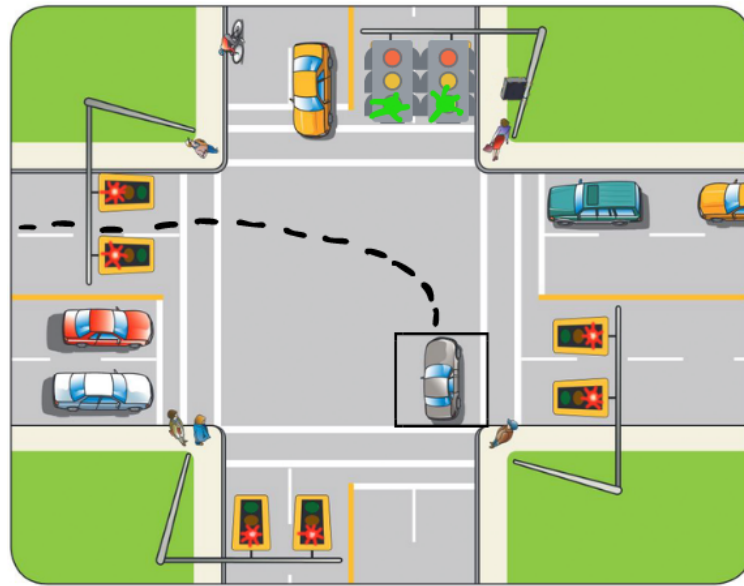


Figure 4.1: Traffic Light Intersection In United States of America.

The moderator must be present during the tasks to ensure there aren't any doubts and everything goes as defined. This person does not interfere in the driving, but it can remind details to the participant. For instance, if they are stuck, the reverse gear is used by pressing the letter Q on the keyboard. If the normal route is disturbed by an unpredictable event, the responsible must instruct the participant to act as they would in real life and continue their path accordingly.

The pre-test questionnaire should last at most 2-3 minutes. The task regarding the simulations should last for approximately 20 minutes. The participant should have around 8 minutes to observe the dashboard and complete the post-test questionnaire.

Moreover, after driving and watching the autonomous driving, the participants are presented with a dashboard with their simulation results and the algorithm results. This dashboard compares the results extracted from the participant and the algorithm but also visually compares the critical moments, such as stopping at a red light or not. During this time, the participants can see the final results.

When the participants are done seeing the final results, they are presented with a post-test questionnaire regarding the experiment.

4.1.3 Participant Population

The participant target of this experiment is anyone over 18 years old with a driver's license. The driver's license is important because it proves that every participant knows the traffic rules and how to act accordingly, like in real life, and has experienced some critical situations represented in the simulation and can relate to them.

4.1.4 Equipment

To perform this test, a laptop is required with CARLA Simulator installed to run the simulations, python installed to run the scripts that automate the configuration and start of the simulations, and nodejs/npm to run the dashboard that presents the results of the simulation locally. The Logitech G29 Steering Wheel and Pedals are also needed.

This framework was developed in Linux. The specs used to build this framework, having in mind the system requirements recommended by the CARLA documentation ¹, are the following:

- **Operative System and Version** – "CARLA provides support for previous Ubuntu versions up to 16.04. However, proper compilers are needed for Unreal Engine to work properly. Dependencies for Ubuntu 18.04 and previous versions are listed separately below." Ubuntu 18.04 was used for this experiment.
- **130 GB disk space** – "Carla will take around 31 GB and Unreal Engine will take around 91 GB so have about 130 GB free to account for both of these plus additional minor software installations".
- **An adequate GPU** – It was used NVIDIA QUADRO T1000, with 896 CUDA cores and with 4096 MB of memory.
- **Two TCP ports and good internet connection** – "2000 and 2001 by default. Make sure that these ports are not blocked by firewalls or any other applications".

4.1.5 Informed Consent Form

In this form, the participant is informed about what the experiment consists on. It gives a brief explanation of what is going to happen and what the participant is asked to do.

It also acknowledges the participant for their time, specifies the duration of the experiment, and ensures that the completed forms remain anonymous.

Moreover, it lets the participant know the requirements to be eligible for the experiment and provides an email where the participant can reach out for any doubts or concerns.

In the end, it contains a phrase that states that the participant consents to the processing of the data for this experiment, and the participant is required to fill a yes or no answer.

The full informed Consent Form is in Appendix **B.1**.

4.1.6 Pre-Test Questionnaire

The Pre-Test Questionnaire is intended to be filled out before the experiment because it collects demographic data such as age, gender, years of driver's license, and level of education. This data is important to correlate with the post-test questionnaire answers later.

¹https://carla.readthedocs.io/en/0.9.12/build_linux/#system-requirements, (accessed Dec.12,2022)

It also questions if the participant knows what ADAS are, and if so, what is their level of trust in it, evaluated on a scale from 1 to 10.

The Pre-Test Questionnaire is presented in Appendix B.2.

4.1.7 Participant Tasks

To start the experiment, the first task is to fill the pre-questionnaire form with demographic data.

After this, the participant drives two different scenarios, already mentioned in Section 3.2.1, with critical situations using a steering wheel and pedals. The first scenario is composed of 8 critical moments: 3 pedestrian crossings, 3 red lights, 2 cyclists crossings, whereas the second scenario is composed of 10 critical moments: 3 pedestrian crossings, 3 red lights, 2 stop signs, 2 cyclists crossings.

The participant must drive those scenarios following the routes displayed with black dots (Figure 4.2).



Figure 4.2: Black Dots Indicating the Path that the Participant Must Follow On the Scenarios.

When both scenarios are completed, the participant is presented with the driving performance of the algorithm in those same scenarios. This demonstration includes a visual representation from the driver's point of view.

Subsequently, the participant is presented with a dashboard. The participant must go through the dashboard results to formulate his own conclusions.

In the end, there is a post-test questionnaire for the participant.

4.1.8 Metrics

The metrics used in this experience are the ones that CARLA Autonomous Driving Leaderboard defined as appropriate to evaluate the performance of autonomous vehicles². In order to be consistent, the same metrics were used for the user's manual control performance.

²<https://leaderboard.carla.org/>, (accessed Feb.25,2023)

$$\text{Driving Score} : R_i P_i \quad (4.1)$$

$$\text{Infraction Penalty} : \prod_j^{ped, \dots, stop} (p_i^j)^{\#infractions_j} \quad (4.2)$$

The Driving Score 4.1 is composed of the multiplication of the Route Completion, which is the percentage of the i -th route distance completed, and the Infraction Penalty 4.2 that has the base value of 1.0 and decreases with committed infractions of the i -th route.

The infractions have a specific penalty depending on the occasion severity:

- **Collisions with pedestrians** – 0.50.
- **Collisions with other vehicles** – 0.60.
- **Collisions with static elements** – 0.65.
- **Running a red light** – 0.70.
- **Running a stop sign** – 0.80.
- **Off-road driving** – If an agent drives off-road, that percentage of the route will not be considered in the computation of the route completion score.

This means that colliding with pedestrians is the most severe penalty because it drops the value of the Infraction Penalty to half, and running a stop sign is the least severe.

4.1.9 Post-Test Questionnaire

The Post-Test Questionnaire serves the purpose of gathering feedback for the experiment since the participant fills it after it. With the answers retrieved from this form's questions (see Appendix B.3) and with the pre-test questionnaire answers, it is possible to take conclusions addressed in the next chapter.

4.2 Test Environment

To collect diverse and interesting results, the prototype was tested in a career fair in the Science and Technology Faculty of Coimbra University, where Continental Engineering Services had an exhibitor, displayed in Figure 4.3.

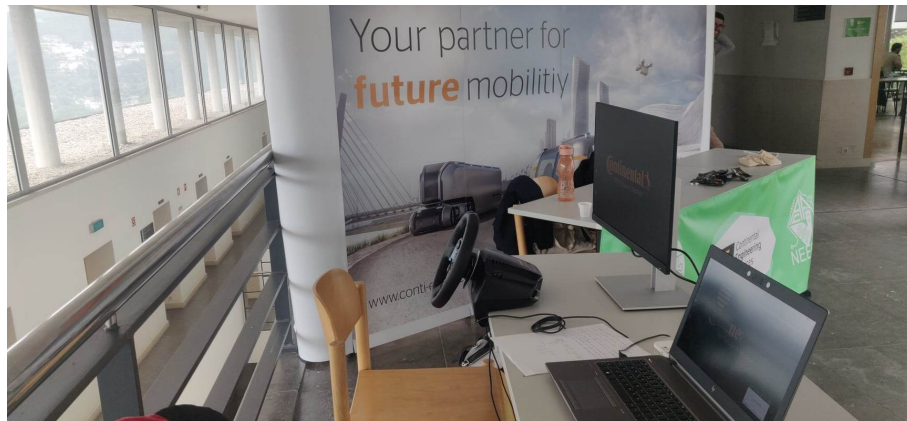


Figure 4.3: Continental Engineering Services Exhibitor.

Students and other members of the University had the possibility to approach this exhibitor in order to get to know more about the company and try out the experiment. The prototype generated a lot of interest, and for that reason, it had someone testing it from the beginning till the end of the day.

The working environment was purposely setup in a way where people could not see what was happening on the monitor in order to be a surprise to everyone that was going to be tested, since this element of surprise was crucial to the experiment.

Therefore, the Logitech Steering Wheel and Pedals were mounted to a table with the monitor in front of the steering wheel, as shown in Figure 4.4.

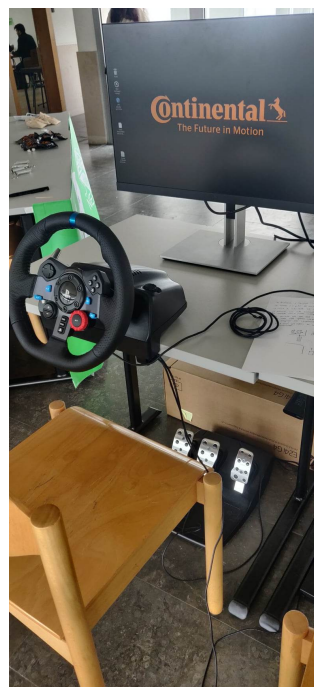


Figure 4.4: Experiment Setup.

4.3 User Feedback

The prototype functioned really well, working as planned. People that came by the Continental Engineering Services exhibitor were very interested in the experiment. Therefore, from the beginning until the end of the career fair, there was always a participant trying out the simulation.

During the tests, verbal feedback was collected while the participants drove on the scenarios like in Figure 4.5.

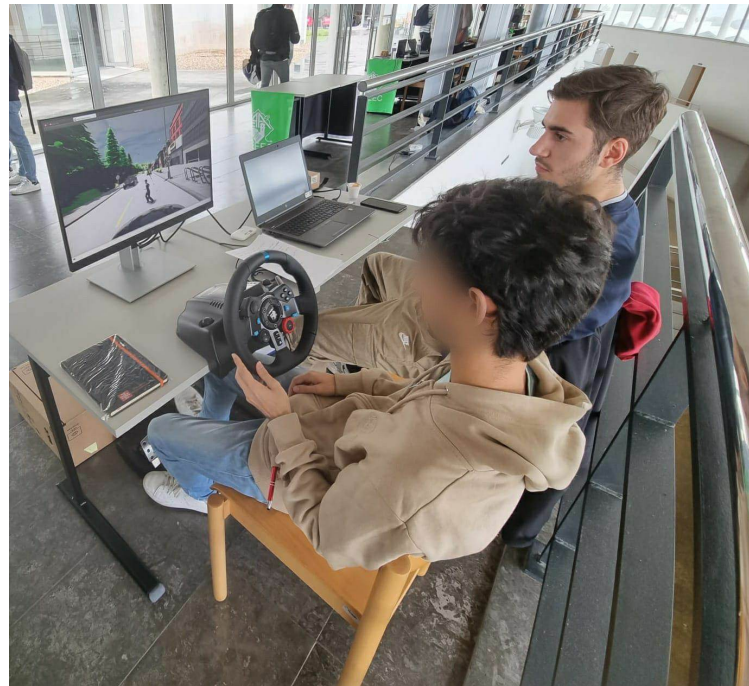


Figure 4.5: Participant Experimenting the First Scenario of the Prototype.

With the input of the participants, it was possible to compile a few remarks on the prototype. Users felt that it was a fun experiment and that they were immediately captivated to try it out. Users also felt that the experiment made them want to know more about ADAS and the experiment itself. Therefore, while they were driving, the tendency was to ask questions about what an AV would do in the same situation with ADAS algorithms as them, and later on, when watching the AV in those same situations, asking questions on how it functioned and paying close attention to the explanation.

Regarding the simulation and the driving tasks, users thought it was intuitive and easy to drive. It was pointed out that a higher frame rate would improve the smoothness of the experiment. Another aspect that the users mentioned were the waiting time at red lights, which could take a little less time, but when it was explained that the frustration of the waiting time, such as in real life, was a factor in the driving style, users understood and agreed that it did change the way they would proceed driving.

Since there was a competitive aspect to the experiment (scores were compared with the algorithm), there were a few users that acknowledged that they were being more careful than in real

life, hoping to beat the algorithm, but even in those cases, there were very few that actually were capable of doing so.

Regarding the time of the experiment, some users were reluctant to spend 30 minutes, but in the end, they were all pleased with it. The dashboard presented in the end received very positive feedback since every user was happy to be able to compare visually what it did versus the algorithm in certain situations and stated that the final score was very clear.

4.4 Summary

This chapter discusses the importance of test protocols in conducting experiments for ADAS algorithms. A test protocol guides the experiment, providing rules and procedures to ensure reliable and reproducible results. The goals of the test are to improve communication about the benefits of ADAS algorithms and to compare the performance of drivers without ADAS assistance to that of AVs with ADAS features.

The general process of the experiment involves introducing the participants to the test, obtaining their informed consent, and guiding them through a series of tasks using a driving simulator. The participants drive through different scenarios with critical situations while following specific rules.

The test requires specific equipment, including a laptop with CARLA Simulator installed, Python for running scripts, Node.js/npm for running the dashboard, and a Logitech G29 Steering Wheel and Pedals. The experiment is conducted on a Linux system, following the system requirements recommended by CARLA.

The participant population for the experiment includes individuals over 18 years old with a driver's license. The driver's license ensures that participants are familiar with traffic rules and can relate to critical situations represented in the simulations.

The experiment adopts metrics defined by the CARLA Autonomous Driving Leaderboard to evaluate the participants' performance. The metrics include a driving score and an infraction penalty, with different penalties for various infractions such as collisions and running red lights or stop signs.

After completing the experiment, participants fill out a post-test questionnaire to provide feedback. This experiment was done at a career fair in the Science and Technology Faculty of Coimbra University with Continental Engineering Services. Users found the experiment captivating, generating curiosity about ADAS and the experiment itself. They found the simulation intuitive and easy to drive, although some suggested improvements in frame rate and waiting times at red lights. The dashboard presented at the end of the experiment received positive feedback, as participants appreciated the visual comparison between their performance and the algorithm's performance.

Chapter 5

Results and Discussion

This chapter presents an analysis of the results in Section 5.1, obtained in the experiment, and a following discussion in Section 5.2.

5.1 Results

Participants filled out a pre-test questionnaire for demographic data before the tests were conducted. In terms of age, it was gathered a wide variety ranging from the age of 18 up to 63 years old. The age of 30 was the most common with four users, the age of 20 was the second most common with three users, and two users with the age of 54, as its shown in Figure 5.1. There were also participants with 18, 19, 22, 23, 25, 26, 29, 40, 57, 62 and 63 years.

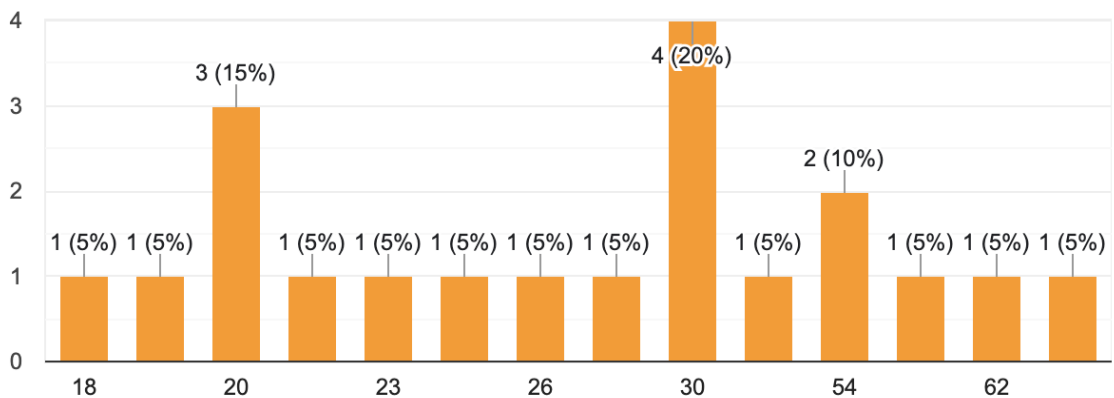


Figure 5.1: Distribution of Participants Per Age.

This age variety is important to the tests because it represents different types of experiences and stages in life. Therefore, there was also an extensive variety in years of experience driving, as there were users with approximately 1 year of driver's license up to 45 years. Four participants had 1 year of driver's license(Figure 5.2).

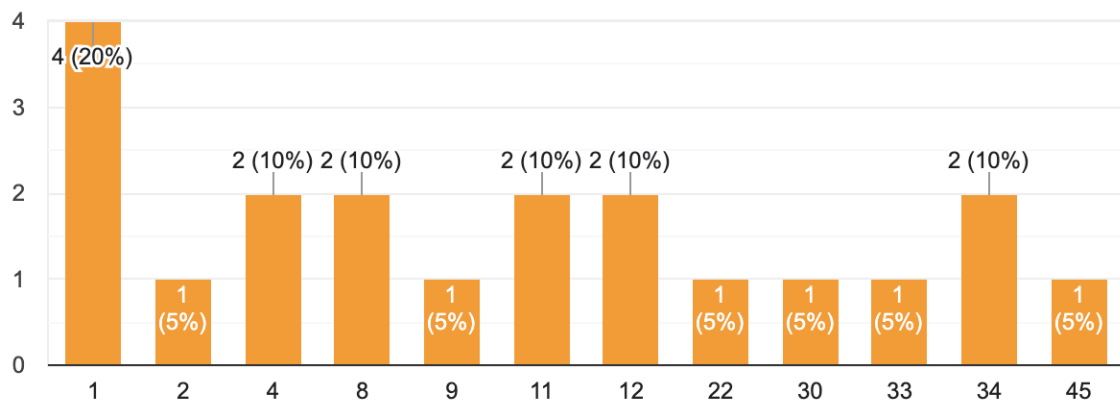


Figure 5.2: Driver's License Years.

The majority of the participants were male (14 out of 20 participants), representing, in this case, 70% of the participants, and 30% of the 20 participants were female (6 out of 20 participants). This percentage is presented in a pie chart in Figure 5.3.

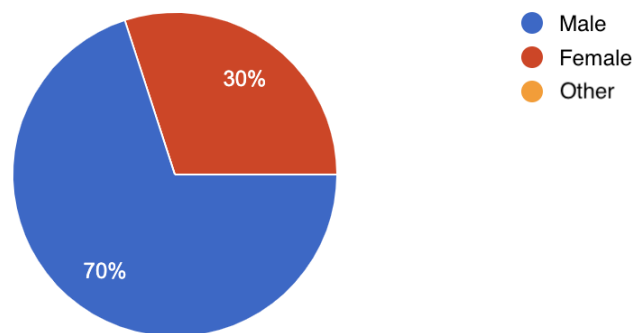


Figure 5.3: Distribution of Participants Per Gender.

All of the users finished at least high school (55%), while 15% had a Bachelor's degree and 30% had a master's degree (Figure 5.4).

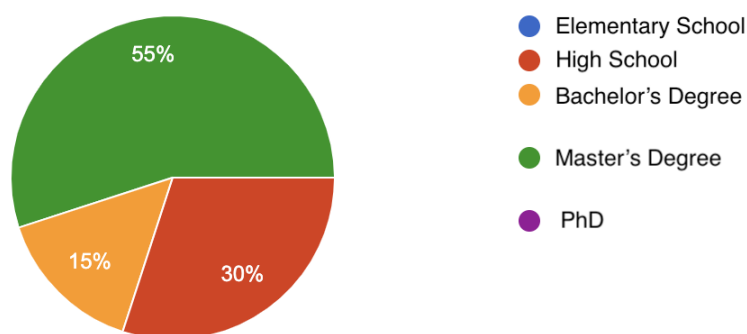


Figure 5.4: Highest Level of Education Distribution.

Of the 20 participants who took part in the experiment, 6 did not know about ADAS and what they are (Figure 5.5).

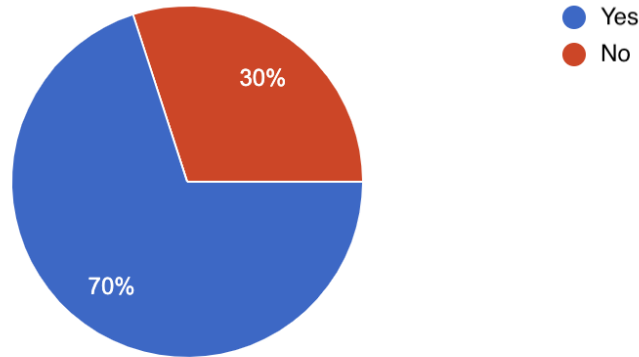


Figure 5.5: Knowledge of ADAS.

The rest of the participants who did know about ADAS and what they did, had different levels of trust in those systems. Figure 5.6 indicates that.

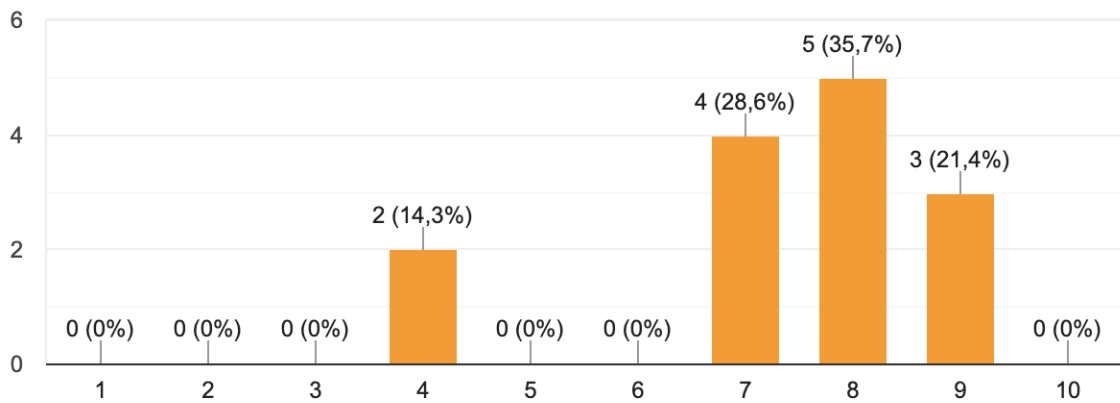


Figure 5.6: Level of ADAS Trust.

Two participants had pretty low trust in ADAS, as they chose 4 from 1 (big distrust in ADAS) to 10 (a lot of trust in ADAS). But most participants who already knew about ADAS had a pretty high level of trust in those systems, with 28.6% having level 7 of trust, 35.7% level 8 and 21.4% level 9.

After the Pre-test questionnaire, the simulation results were extracted to evaluate the participant's performance and compare it to the InterFuser algorithm. Table 5.1 presents the score of each scenario, the final score, which is the average score of both scenarios and the number of infractions committed (running a red light or a stop sign, collisions with pedestrians, vehicles or objects). It also displays the first (Q1) and third quartile (Q3), the median, the standard deviation (stddev), and the average (mean) results of all the participants. Lastly, it presents the results of the InterFuser algorithm.

Table 5.1: Simulation Results.

Participants	Scenario 1	Scenario 2	Final Result	Red Light	Stop Sign	Pedestrian	Vehicle	Object
1	50	63	56.5	0	2	1	0	0
2	100	63	81.5	0	2	0	0	0
3	100	28	64	1	1	1	0	0
4	24	31	27.5	4	2	1	0	0
5	15	12	13.5	1	1	2	4	0
6	100	29	64.5	2	0	0	1	0
7	30	47	38.5	0	1	1	2	0
8	100	70	85	1	0	0	0	0
9	50	19	34.5	0	2	2	1	0
10	50	21	35.5	1	0	2	1	0
11	50	55	52.5	1	1	1	0	0
12	100	98	99	0	0	0	0	0
13	50	29	39.5	2	0	1	1	0
14	25	7	16	3	0	2	3	0
15	50	34	42	3	0	1	0	0
16	30	24	27	2	1	1	2	0
17	27	14	20.5	1	1	1	3	1
18	8	34	21	1	0	4	1	0
19	100	80	90	0	1	0	0	0
20	100	33	66.5	1	1	0	1	0
Q1	29.25	23.25	27.38	0	0	0	0	0
MEDIAN	50	32	40.75	1	1	1	1	0
Q3	100	49	65	2	1	1.29	1.25	0
STDDEV	33.92	24.48	26.06	1.15	0.77	1.00	1.21	0.22
AVERAGE	57.95	39.55	48.75	1.20	0.80	1.05	1.00	0.05
InterFuser ALGORITHM	51	100	76	0	0	0	0	0

The InterFuser algorithm had 51% on Scenario 1, a score calculated by the multiplication of Route Completion and Infraction Penalties because it did not finish the route. Due to a big shadow on the road, the algorithm did not advance, having mistaken it for an object in front. Therefore, despite eventually being green at the traffic light, it stayed there (Figure 5.7). For this reason, it only completed 51% of the path without committing infractions, so it ended the scenario with that percentage.



Figure 5.7: InterFuser Algorithm Mistake Due To Big Shadow.

In Scenario 2, the InterFuser algorithm had 100%, where it finished the whole path without a single mistake. This resulted in a final evaluation of 76% with 0 infractions.

The experiment was tested on 20 participants, which had an average of 48.75% in the final results. This average is way lower than the final result of the InterFuser algorithm, meaning that most of the participants had a worse performance than the algorithm. It is also possible to see that the first quartile is closer to the second quartile (median) than the third quartile. This means that the lower 25% of the data points are relatively closer to the median, while the upper 25% of the data points are more spread out. It can also be determined that the average (mean) is higher than the median. This suggests that the presence of a few larger values are pulling the average towards higher values. The standard deviation suggests a moderate level of variability or dispersion within the dataset.

Very few participants beat the InterFuser algorithm, like the case of participants number 2, 8, 12, and 19. But number 12 is a particular case, having a 99% score. It avoided every critical situation but did 2% of the path off-road (that is why it had 98% on the second scenario). This participant was extremely slow and careful, trying to predict every situation and stopping when there was no actual sign to do so. Although this could be classified as "defensive driving", it was extreme, and its score could be easily harmed with a time limit. Despite this, no time limit was implemented because participants drive at their own pace in real life.

In the final results, it was evident that every participant who successfully defeated the algorithm achieved a perfect score of 100% in scenario 1, which was comparatively easier than scenario 2. Furthermore, the results in scenario 1 showed a wide range of variation, with scores ranging from 100% to 50%. This variation can be attributed to a specific challenging moment in the scenario: a child suddenly appeared crossing the road from behind an obstructing object. If the participant's vehicle collided with the child, their score would be penalized and reduced to 50%. As a result, participants who managed to avoid this collision (and did not commit any additional infractions) maintained a score of 100%, while those who experienced the collision (and did not commit any additional infraction) received a score of 50%.

Scenario 2 was more complex, and that is why there is a bigger variety in scores. It had more critical moments, and participants made all kinds of mistakes.

The average of the participants' results indicates that, in a day-to-day testing scenario, at least one red light is crossed, a pedestrian is hit if it appears from an unexpected place, and there is an accident with another vehicle. Occasionally a stop sign is also ignored, and there is rarely a collision with an object. This last metric (Collision with obstacles) had the expected low occurrence because the scenarios tested were traffic scenarios, but a scenario that would test parking situations would probably increase the chances of this type of collision.

After the simulations, the participants filled in a post-test questionnaire. In the first instance, it was asked about the realism of the critical situations that occurred in the simulation (Figure 5.8) from 1 (does not happen in real life) to 5 (happens in real life). After examining the findings from the questionnaire, it was determined that the response was highly positive, as 60% of the participants considered that the situations displayed in the simulation accurately portrayed dangerous real-life situations by selecting the maximum rating of 5, and 35% rating it 4 out of 5.

One participant voted 2 out of 5 on this question, stating that it was not that real because it doesn't happen exactly like in real life. For example, in the simulation, pedestrians crossed the road behind a vending machine that blocked the driver's view. The goal was to reproduce this event where pedestrians cross the road in unexpected moments with low visibility.

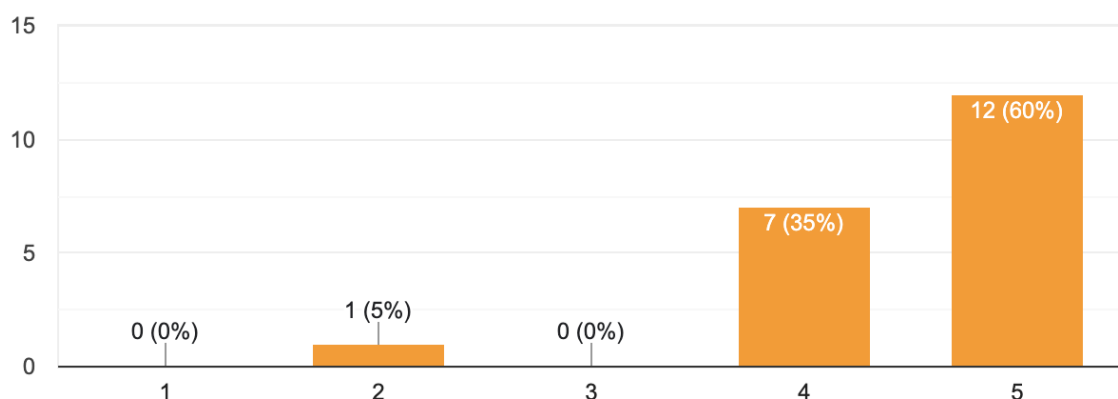


Figure 5.8: Level of Realism in Critical Situations.

In the simulations, there were several types of critical situations. Therefore, it was asked if the participants had already been in those situations or witnessed someone in those situations to confirm that these events actually happen in real life.

The results achieved in Figure 5.9 confirmed that these events were accurate. Every participant had already crossed a red light or seen a pedestrian crossing the road in an unexpected place with low visibility. 90% had already ignored a stop sign or been involved in a car accident with another vehicle, while 80% had seen a cyclist crossing the road wrongfully. Accidents with obstacles were less prominent, but 13 participants had already witnessed or been involved, while 5 participants had seen an accident with a pedestrian.

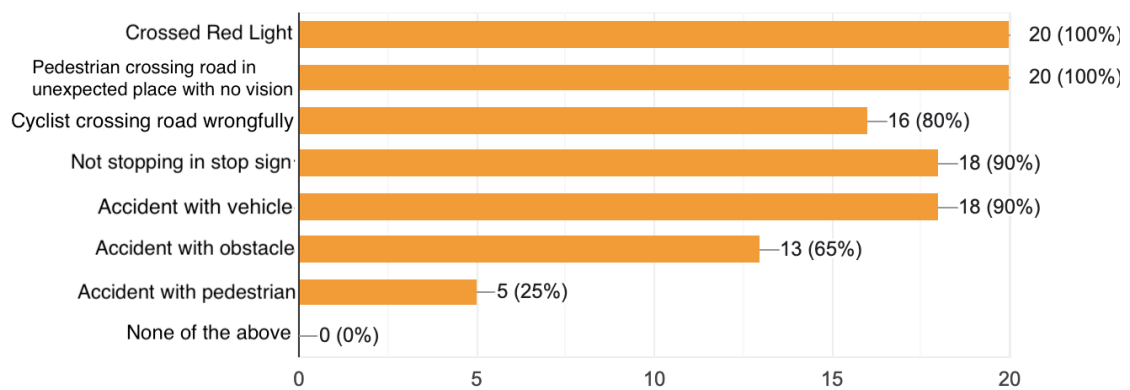


Figure 5.9: Critical Situations In Real Life.

Regarding the first scenario tested, where the InterFuser algorithm did not complete the whole route and had a 51% score, 13 participants (65%) did not score better than the algorithm. Only 7 participants (35%) had a better score in this scenario, as shown in Figure 5.10.

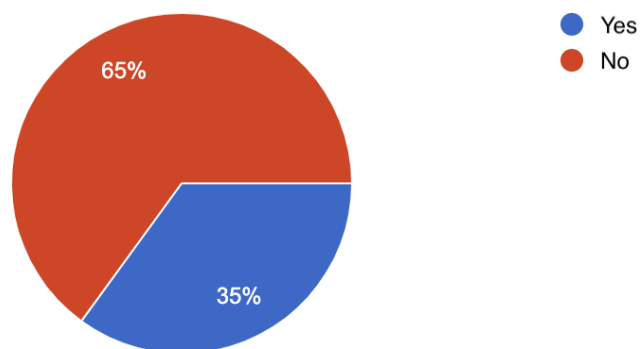


Figure 5.10: Participants Outperformed Algorithm In Scenario 1.

Regarding the second scenario, where the InterFuser algorithm drove it perfectly, having a score of 100%, no participant matched this score (Figure 5.11).

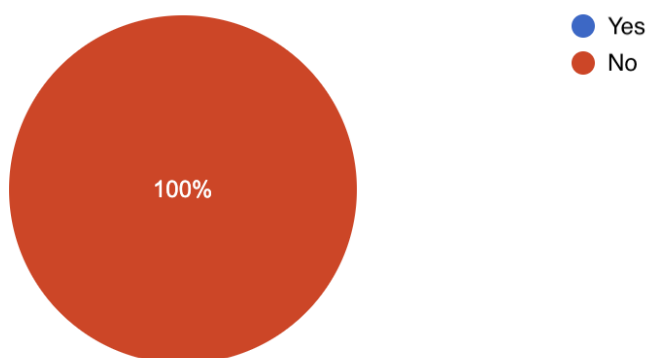


Figure 5.11: Participants Outperformed Algorithm In Scenario 2.

Therefore, of the 7 participants who had a better score in the first scenario, only 4 were capable of having a better final score combining the two scenarios. This means that only 20% of the participants were able to beat the algorithm (Figure 5.12).

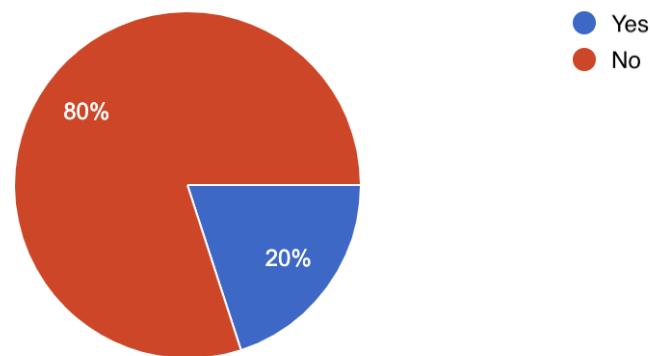


Figure 5.12: Participants Outperformed Algorithm In Final Results.

At the end of the experiment, it was presented a dashboard with the score comparison and statistics. This dashboard also included GIFs of the participant's performance and the algorithm in critical situations side by side.

Participants were asked about the quality of this dashboard and if the experiment results were clear. The response was extremely positive since all 20 participants voted 5 out of 5 (Figure 5.13). Participants felt that it was a very interesting way to show this comparison and that it was very intuitive and clean.

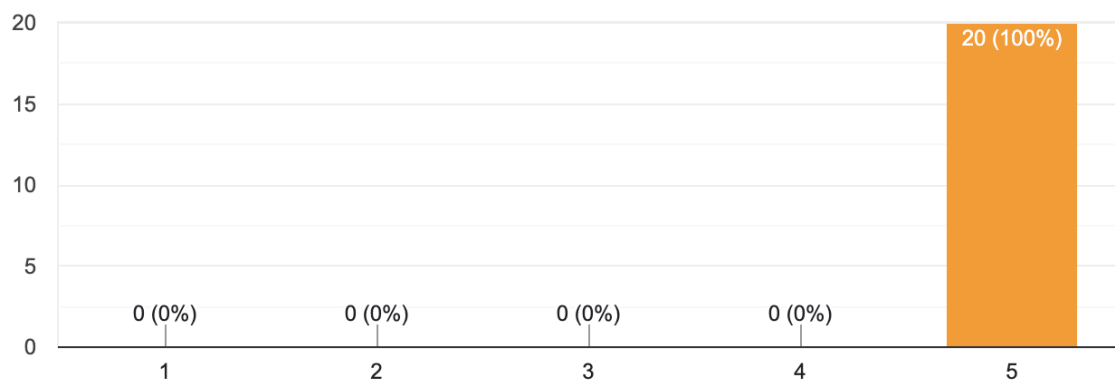


Figure 5.13: Quality of Dashboard.

The end goal of this dissertation was to do an interactive driving experience that allowed an effective communication of the benefits of these ADAS algorithms in our day-to-day.

Therefore, it was asked the question to the participants of how they felt about the experiment. From 1 (very poor communicative process) to 7 (very good communicative process), how well did this experiment communicate the benefits of ADAS algorithms?

Once again, the results were very positive since 30% voted 6 and the majority (70%) voted 7. This means that the participants felt it was an extremely effective experiment that enhanced their awareness of ADAS algorithms and the communication of its benefits. This opinion was unanimous among the participants who already knew about ADAS and who did not.

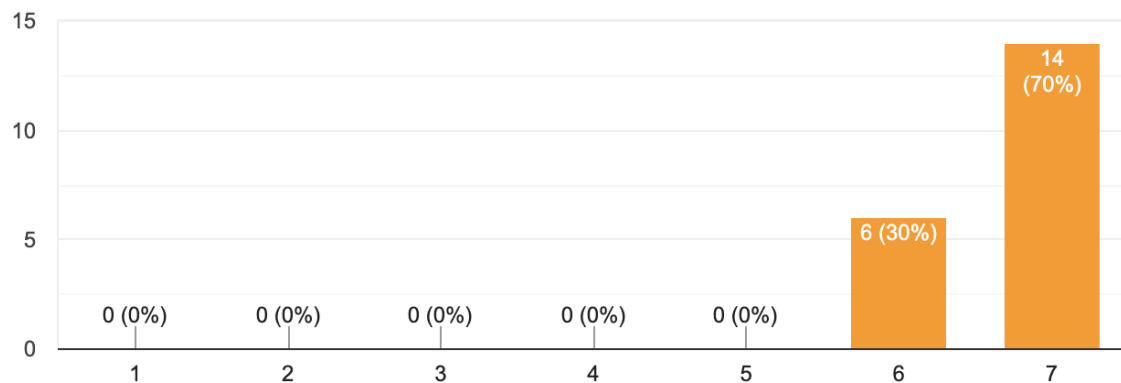


Figure 5.14: Classification of Experiment ADAS Benefit Communication.

With the chart in Figure 5.15, it is possible to say that this experiment also contributed to raising the level of trust in those systems, since 40% felt it really made them gain a much bigger trust than they had in those systems voting 7, 25% voting 6 and another 25% voting 5. There were also 2 participants who voted 4, meaning the level of trust was maintained but justified by the fact that their trust was already really high.

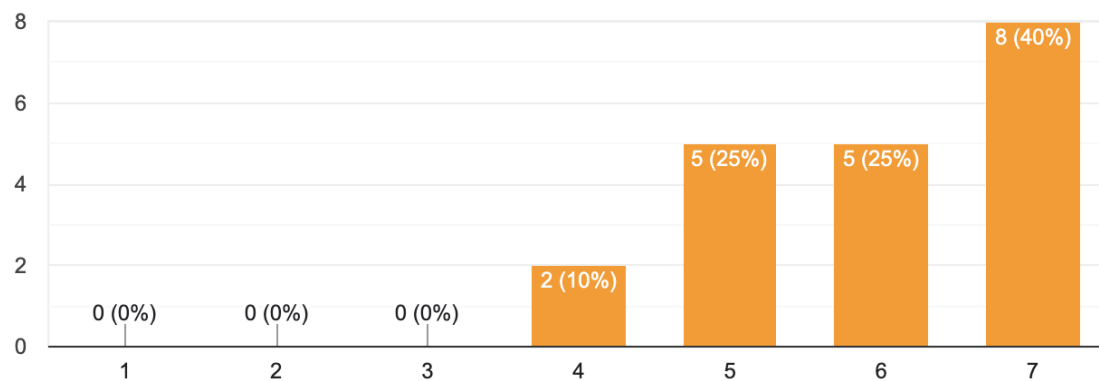


Figure 5.15: Level of ADAS Trust.

5.2 Discussion

In order to analyze the outcomes of the experiment, a set of questions were created that rely on combining the answers from different questions to obtain meaningful insights and conclusions:

- **Q1 + Q11:** Is age a factor in the participant's performance?

To evaluate this question, it was made a scatter plot graph (Figure 5.16) that correlates age (from 18 to 63) with performance (from 0% to 100%).

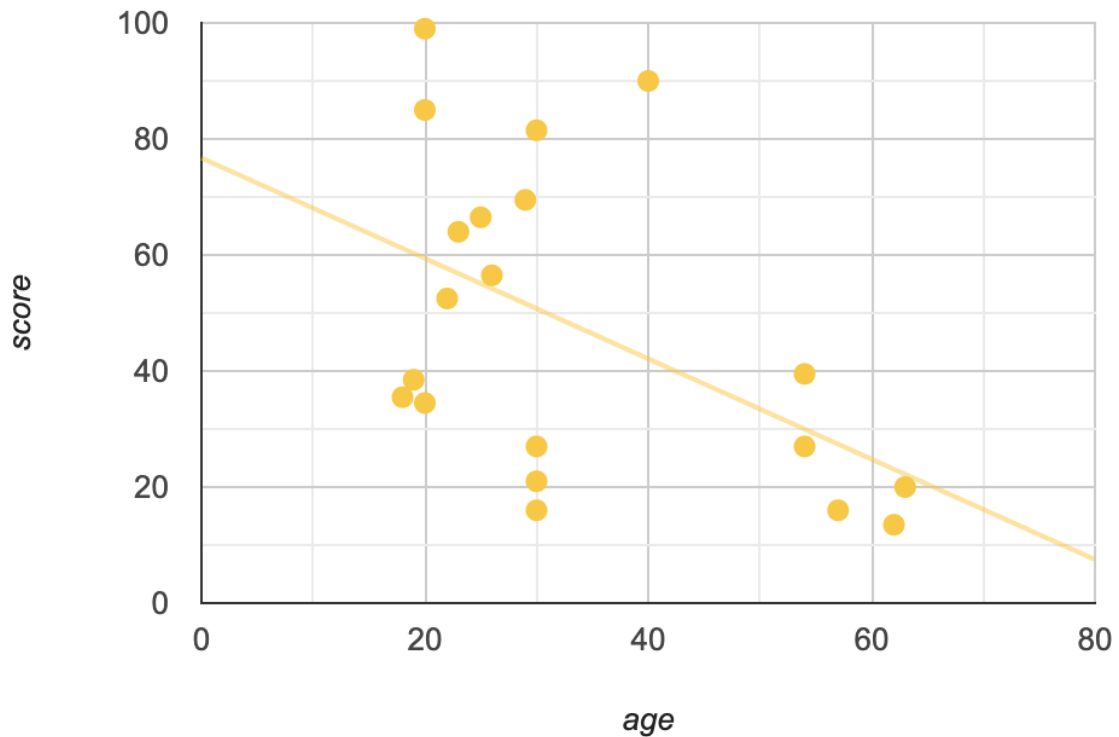


Figure 5.16: Age and Performance Correlation.

In this graph, it is possible to see that the trend line indicates that the final score goes down with age. Even though this could be attributed to a slower reaction time or worse vision, it was not the case. An observed trend was that participants of older age faced more difficulties in operating the steering wheel and pedals during the simulation.

Since the simulation had a game feel to it, the younger generation had no problems managing and adapting to the steering wheel and pedals and how to drive in the simulation. This was not so true for the older generation, and this justifies the worse results in general.

- **Q2 + Q11:** Are the years of experience helpful in understanding the realism of the scenarios' critical situations?

To understand this question, a graph was created correlating the opinion of the critical situations' realism (from 1 to 5) and the years of experience driving (from 1 to 45) (Figure 5.17).

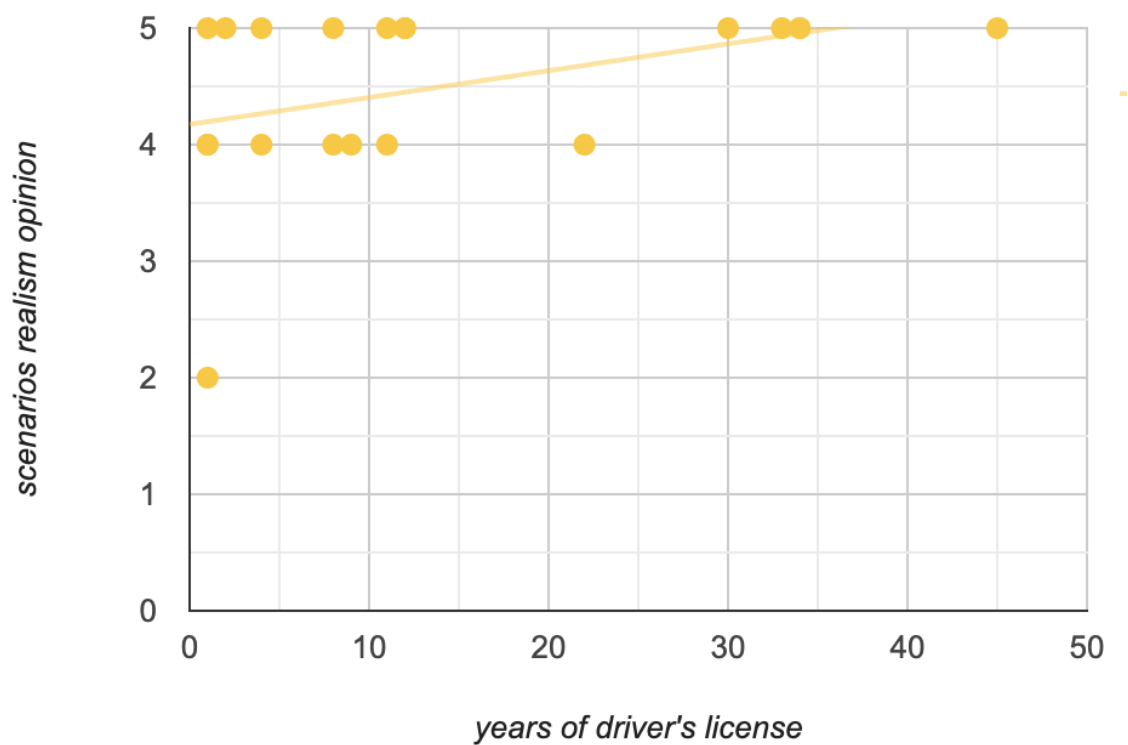


Figure 5.17: Years of Driver's License and Critical Situations' Realism Opinion Correlation.

From the correlation made in this graph, it can be said that the trend line indicates that the more experienced the participant, the more it found the critical situations realistic. Assuming that the more experienced drivers have been through more critical situations in real life than the less experienced drivers, it validates the scenarios chosen. Despite this, even the majority of the less experienced drivers also felt that it was realistic according to real-life experiences.

- **Q6 + Q14:** Did the experiment improve the overall trust in ADAS?

To analyse this, a graph was made that correlates the previous trust that the participant had with the trust it gained, maintained, or lost after the experiment (Figure 5.18).

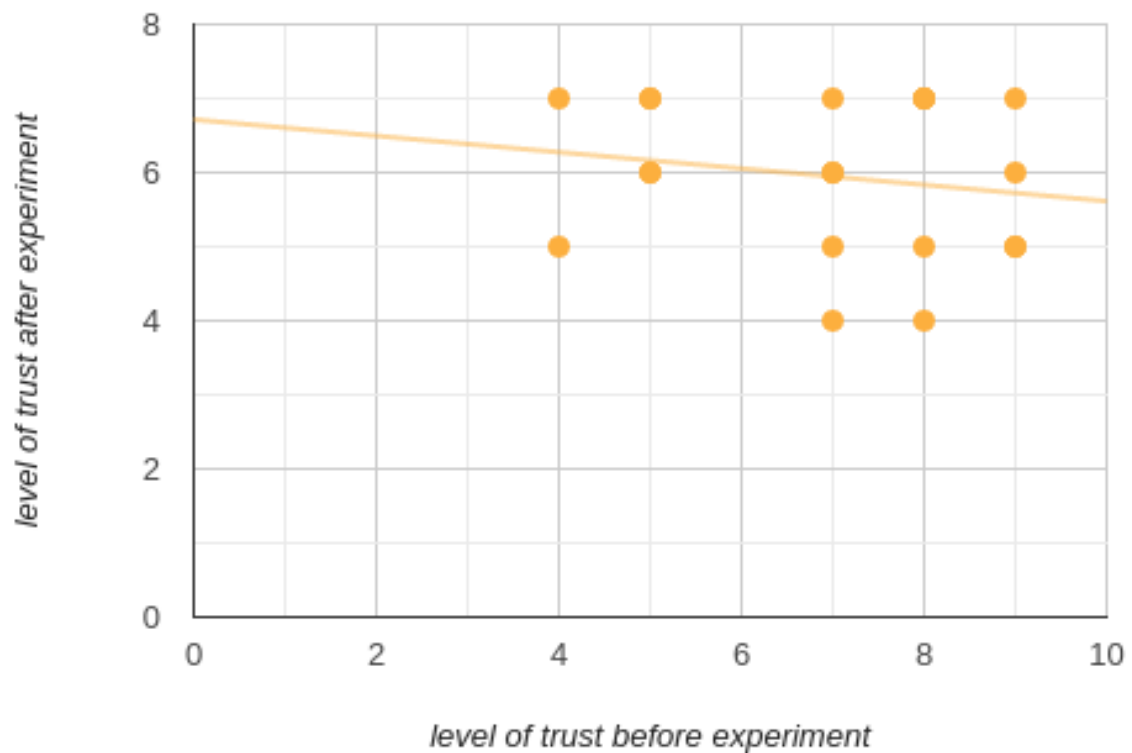


Figure 5.18: Level of ADAS Trust Before and After Experiment Correlation.

In this situation, a few participants did not know about ADAS, so they were given the value 5 from 1 to 10, meaning they did not trust nor distrust the systems.

Through the graph, it is possible to understand what the trend line indicates. Participants who had a lower initial level of trust experienced a bigger increase in their trust levels after the experiment. Despite this, it is clear that even participants with high trust in those systems felt that this experiment increased their level of trust.

For the participants who had a low level of trust before the experiment or did not know about ADAS, it also turned out to be a positive experiment since it raised their level of trust.

- **Q12 + Q13:** Is the dashboard well designed and contributed to a better understanding of the ADAS benefits?

The dashboard had an evaluation of 5 out of 5 by every participant, seen in Figure 5.13, and the experiment had an evaluation of mostly 7 out of 7 and a few 6 out of 7 in terms of communicating the benefits of ADAS algorithms effectively, seen in Figure 5.14.

This means that the whole experiment was successful in communicating the benefits and enhancing the awareness of ADAS algorithms in our day-to-day lives through the use of a driving simulator. Therefore, the initial hypothesis that the 3D driving simulator usage, in the Automotive Industry context, can enhance the communication of the ADAS algorithms benefits and how these systems act in the driving process was met.

Using a driving simulation where participants could drive with a steering wheel and pedals in the same scenarios as the algorithm turned out to be a very effective way of communicating the benefits of these algorithms, but also the presentation of a clear dashboard that showcased these final results and comparisons played a big role in this communication. Participants were able to see side-by-side their performance and compare it with the algorithm visually and statistically. This visual comparison enabled the participants to draw their own conclusions regarding the effectiveness of these algorithms and in what situations they can be a great help.

After analyzing the results, it is possible to answer the questions in the Problem Section. Are the drivers aware of how ADAS can assist them in certain situations? Some participants did not know about ADAS, others had very little trust in those systems, but every participant, by the end of the experiment, was aware of the benefits of ADAS and how these systems can assist drivers in critical situations.

And would their performance be the same in identical scenarios without the help of ADAS? This experiment also proved that the performance of most drivers was worse than the algorithm. The algorithm showcased its dominance, especially in situations where reaction time was key. And through this demonstration, it became clear to participants that ADAS algorithms are a tremendous help to drivers.

5.3 Summary

This chapter focuses on the analysis of the results obtained from the experiment to which the participants were submitted. The purpose of this test was to communicate the benefits of ADAS algorithms and to promote a greater understanding of their role in everyday driving.

In order to optimize the extraction of the obtained results, the participants were asked to fill out a post-test questionnaire that comprised several questions. When questioned about the realism of the critical scenarios presented in the simulation, the majority of the participants responded positively, agreeing that the scenarios accurately portrayed life-threatening situations found in real-life circumstances. The participants also verified that they had personally encountered or witnessed similar situations in their own personal experiences.

The experiment included two scenarios. In the first scenario, where the algorithm did not complete the entire route, only a small number of participants outperformed the algorithm. In the second scenario, where the algorithm demonstrated flawless execution, no participant scored equally. Overall, only a minority of participants were able to surpass the algorithm's performance when combining the results of the two scenarios.

At the end of the tests, the participants were presented with a dashboard showing the score comparison and statistics, along with GIFs comparing their performance with the algorithm in critical situations. When asked about the dashboard, all the participants remarked on its high quality and commented on how it effectively communicated the experiment results.

Additionally, the study assessed how participant-related variables, such as age and years of driving experience, influenced performance. It was determined that older participants faced more difficulties in operating the steering wheel and pedals, which affected their performance.

When queried about the levels of trust in the algorithms under study, the majority of the engagers reported higher levels of trust after the experiment, regardless of their initial level of trust or prior knowledge of ADAS.

After thorough deliberation of all the given answers, one can say that the use of a driving simulator and the clear presentation of results through the dashboard proved to be effective in communicating the benefits of ADAS algorithms. Participants were able to visually and efficiently compare their performance with the algorithm's, enabling them to draw their own conclusions about the effectiveness of these systems.

Overall, the experiment achieved its goal of enhancing awareness and communication of ADAS algorithm benefits. The driving simulation and the informative dashboard played crucial roles in transmitting the actual advantages of these algorithms. Hence, it can be stated that this study demonstrated that ADAS algorithms can significantly assist drivers in critical situations and that it effectively heightened the participants' awareness of these benefits.

Chapter 6

Conclusions and Future Work

6.1 General Conclusions

This dissertation was initiated with the aim of communicating and enhancing awareness of ADAS algorithms using a driving simulator. With that in mind, a few questions were raised: Do modern car drivers know what happens behind the scenes and how these systems impact their day-to-day life? Are the drivers aware of how ADAS can assist them in certain situations? And would their performance be the same in identical scenarios without the help of ADAS?

Therefore, the goal of developing an interactive experience in a simulator that effectively communicated these benefits was set. To do so, automotive industry concepts were studied.

After analysing these concepts, it was concluded that simulators are already essential in the automotive industry in order to develop, test, and validate these systems. That is because tests are often expensive, time-consuming, dangerous, and some times impossible to recreate. Despite not replacing real-life tests, simulators can eliminate those factors.

Therefore, extensive research was done on the simulators available in the market. Even though there are several powerful simulators, the CARLA simulator gathered the most positive attributes for the purpose of this dissertation. It is an open-source simulator that allows realistic driving simulations through the use of Unreal Engine 4. It is also known for having a very well-documented API that facilitates the manipulation of the simulations and scenarios. After selecting the most suitable simulator, research was made on how to develop a valuable experiment, but also on how to select the necessary requirements for having a well-structured study.

After the study of concepts, the prototype started to be designed and developed. After a lot of consideration of the prototype's architecture, it became clear that it should have three main components—Interface, Simulation, and Dashboard. Interface automatized the processes of the experiment, generating the simulations and dashboard with a click of a button. Simulation component included the integration with the steering wheel and pedals and the evaluation of the performances in the defined scenarios. The dashboard component gathered all the data collected from the simulation and presented the final results in an appealing way to demonstrate to the participants whether the algorithm could be better or not.

Hence, a test protocol was developed in order to perform the experiment consistently. This included the goals, general process, participant population, equipment necessary, consent form, the questionnaires, metrics evaluated, and a full explanation of participant tasks.

The prototype was taken to a career fair in which Continental Engineering Services participated to test and collect data and where participants with driver's licenses could take part in the experiment. User feedback was extremely positive, and there was significant enthusiasm for participating in the experiment. Participants enjoyed the possibility of driving the scenarios knowing that there was an algorithm score to beat and ended up impressed with the difference in results.

Thus, after collecting a considerable amount of data, the results were analyzed. These allowed to conclude that the experiment succeeded in communicating the benefits of ADAS algorithms and improving the trust in those systems. All participants felt that the dashboard displayed this comparison and information in an intriguing way that allowed an easy understanding of where the participants failed, and the algorithm did not and vice-versa.

In conclusion, this dissertation successfully validated the initial hypothesis that the usage of a 3D driving simulator, within the context of the Automotive Industry, can enhance the communication of ADAS algorithm benefits and their role in the driving process. Through the experiment conducted, participants became aware of the advantages of ADAS algorithms and how they can assist drivers in critical situations. The driving simulation, combined with the clear presentation of results through the dashboard, proved to be an effective means of conveying the benefits of these algorithms.

6.2 Future Work

The prototype developed in this dissertation accomplished its purpose by providing a comparison between the performance of a human driver and a driving algorithm in day-to-day scenarios.

That being said, there is room for improvement. In order to make the prototype more immersive and realistic, there are a few things that can be improved in the future. Optimizing the underlying code is crucial to improve the frames per second (FPS) of the simulation. By increasing the FPS, the gameplay will become smoother, enhancing the users' overall experience and providing a more realistic feel. The addition of a speedometer and side mirrors view would also improve this experiment by increasing the realistic feel.

To assess the human driver and algorithm's performance, it is important to explore new evaluation metrics. The integration of a speedometer would allow to explore the speed limits as a metric. It would also be interesting to have a metric that would evaluate the safety distance from the car ahead.

There are more critical situations that occur in real life that were not evaluated and can be designed to improve this experiment. Incorporating roundabouts and a few vehicles that do not follow its rules and implementing a parking lot scenario where stress, time, and patience are a big factor to the driver, are a few examples of new critical situations.

Furthermore, expanding the sample size of participants tested would yield valuable insights and contribute to more robust and accurate conclusions.

References

- [1] S Andradottir, K J Healy, D H Withers, and B L Nelson. Introduction to modeling and simulation.
- [2] Sebastien Bell. Mercedes becomes first brand approved for level 4 self-driving in germany (but only in one parking lot so far) | carscoops. <https://tinyurl.com/MercedesLevel4>. (accessed Dec.19,2022).
- [3] European Commission. European commission intelligent transport systems. https://transport.ec.europa.eu/transport-themes/intelligent-transport-systems/road_en. (accessed Dec.14,2022).
- [4] Weiwen Deng, Yong H Lee, and Annie Zhao. Hardware-in-the-loop simulation for autonomous driving.
- [5] Alexey Dosovitskiy, German Ros, Felipe Codevilla, Antonio Lopez, and Vladlen Koltun. CARLA: An open urban driving simulator. In *Proceedings of the 1st Annual Conference on Robot Learning*, pages 1–16, 2017.
- [6] Daniel J. Fremont, Edward Kim, Yash Vardhan Pant, Sanjit A. Seshia, Atul Acharya, Xantha Bruso, Paul Wells, Steve Lemke, Qiang Lu, and Shalin Mehta. Formal scenario-based testing of autonomous vehicles: From simulation to the real world. Institute of Electrical and Electronics Engineers Inc., 9 2020.
- [7] Hubert Igliński and Maciej Babiak. Analysis of the potential of autonomous vehicles in reducing the emissions of greenhouse gases in road transport. *Procedia Engineering*, 192:353–358, 2017.
- [8] Javier Llamazares Luis Montoro Ignacio Lijarcio, Sergio A. Useche. Cavailability, demand, perceived constraints and disuse of adas technologies in spain: Findings from a national study. *IEEE*, 2019.
- [9] Peraphan Jittrapirom, Valeria Caiati, Anna-Maria Feneri, Shima Ebrahimigharehbaghi, María J Alonso-González, Jishnu Narayan, M J Alonsogonzalez@tudelft NL, J N Sreekantannair@tudelft, (J N NL, Soora Rasouli, Harry Timmermans, and Dujuan Yang. Mobility as a service: A critical review of definitions, assessments of schemes, and key challenges. 2:13–25, 2017.
- [10] Prabhjot Kaur, Samira Taghavi, Zhaofeng Tian, and Weisong Shi. A survey on simulators for testing self-driving cars. *Proceedings - 2021 4th International Conference on Connected and Autonomous Driving, MetroCAD 2021*, pages 62–70, 1 2021.

- [11] Nandavar S. Yasmin S. Lewis I. Oviedo-Trespalcacios O. Kaye, S. Consumer knowledge and acceptance of advanced driver assistance systems. *Transportation Research Part F: Traffic Psychology and Behaviour*, 90, 300-311, 2022.
- [12] Karl Koscher, Alexei Czeskis, Franziska Roesner, Shwetak Patel, Tadayoshi Kohno, Stephen Checkoway, Damon McCoy, Brian Kantor, Danny Anderson, Hovav Snachám, and Stefan Savage. Experimental security analysis of a modern automobile. *Proceedings - IEEE Symposium on Security and Privacy*, pages 447–462, 2010.
- [13] Vipin Kumar Kukkala, Jordan Tunnell, Sudeep Pasricha, and Thomas Bradley. Advanced driver-assistance systems: A path toward autonomous vehicles. *IEEE Consumer Electronics Magazine*, 7:18–25, 9 2018.
- [14] Ammu M Kumar and Philomina Simon. Review of lane detection and tracking algorithms in advanced driver assistance system. *International Journal of Computer Science and Information Technology*, 7:65–78, 8 2015.
- [15] Ugur Kuter and Cemal Yilmaz. Choosing human-computer interaction (hci) appropriate research methods survey methods: Questionnaires and interviews. 2001.
- [16] Averill M Law. Simulation modeling and analysis, fifth edition. 2015.
- [17] Meng Lu, Kees Wevers, Rob Van Der Heijden, and Tom Heijer. Adas applications for improving traffic safety. *Conference Proceedings - IEEE International Conference on Systems, Man and Cybernetics*, 4:3995–4002, 2004.
- [18] Enrique Martí, Joshué Pérez, Miguel Ángel De Miguel, and Fernando García. A review of sensor technologies for perception in automated driving; a review of sensor technologies for perception in automated driving. *IEEE Intelligent Transportation Systems Magazine*, 11, 2019.
- [19] Joel C. McCall and Mohan M. Trivedi. Video-based lane estimation and tracking for driver assistance: Survey, system, and evaluation. *IEEE Transactions on Intelligent Transportation Systems*, 7:20–37, 3 2006.
- [20] W David Montgomery, Richard Mudge, Erica L Groshen, Susan Helper, John Paul MacDuffie, and Charles Carson. America’s workforce and the self-driving future realizing productivity gains and spurring economic growth. 2018.
- [21] Euro Ncap. Assisted driving highway assist systems.
- [22] Euro Ncap. Euro ncap vision 2030: a safer future for mobility.
- [23] Abiodun Ogunyemi and David Lamas. Interplay between human-computer interaction and software engineering. *Iberian Conference on Information Systems and Technologies, CISTI*, 2014.
- [24] Henri Pierreval, Romain Bruniaux, and Christophe Caux. A continuous simulation approach for supply chains in the automotive industry. *Simulation Modelling Practice and Theory*, 15:185–198, 2 2007.
- [25] Guodong Rong, Byung Hyun Shin, Hadi Tabatabaee, Qiang Lu, Steve Lemke, Martinš Možeiko, Eric Boise, Geehoon Uhm, Mark Gerow, Shalin Mehta, Eugene Agafonov,

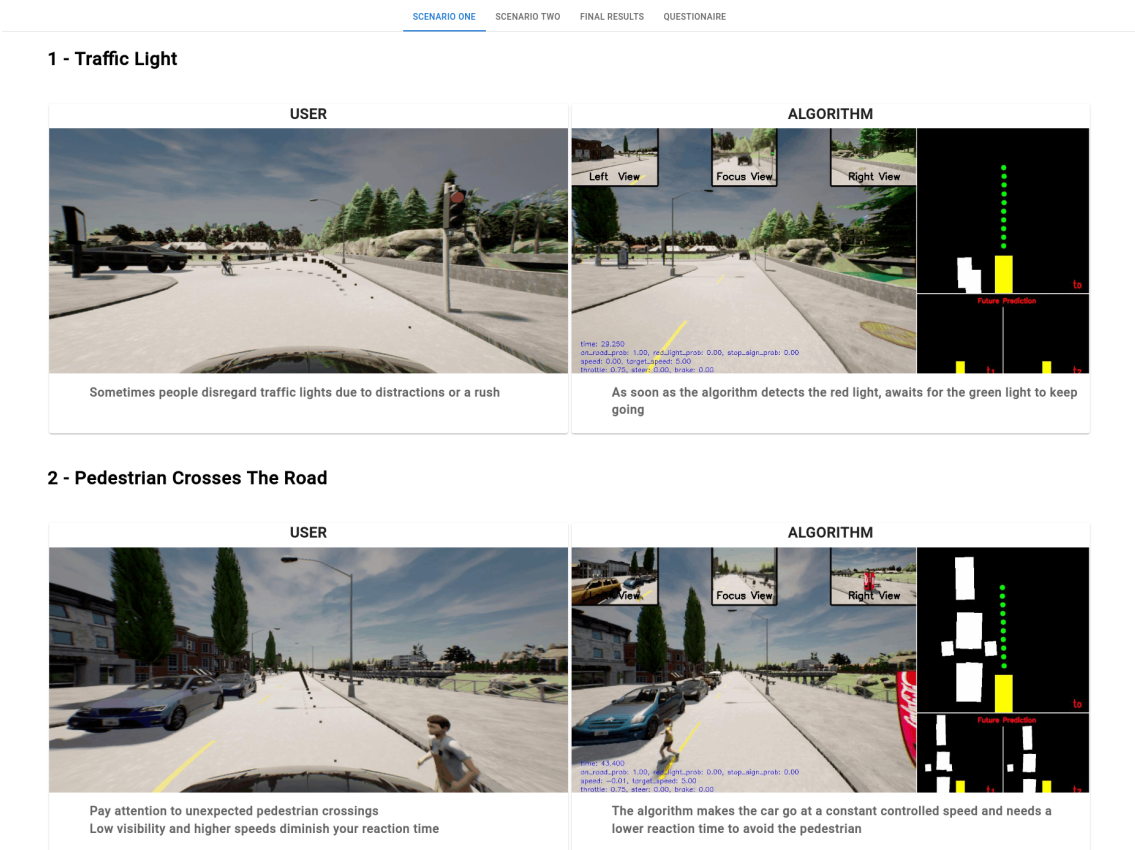
- Tae Hyung Kim, Eric Sterner, Keunhae Ushiroda, Michael Reyes, Dmitry Zelenkovsky, and Seonman Kim. Lgsvl simulator: A high fidelity simulator for autonomous driving. *2020 IEEE 23rd International Conference on Intelligent Transportation Systems, ITSC 2020*, 9 2020.
- [26] SAE. The mission of sae international is to advance mobility knowledge and solutions. <https://www.sae.org/>. (accessed Dec.29,2022).
- [27] SAE. Sae levels of driving automation™ refined for clarity and international audience. <https://www.sae.org/blog/sae-j3016-update>. (accessed Dec.30,2022).
- [28] Birgit Schlager, Stefan Muckenhuber, Simon Schmidt, Hannes Holzer, Relindis Rott, Franz Michael Maier, Kmeid Saad, Martin Kirchengast, Georg Stettinger, Daniel Watzenig, and Jonas Ruebsam. State-of-the-art sensor models for virtual testing of advanced driver assistance systems/autonomous driving functions. *SAE International Journal of Connected and Automated Vehicles*, 3:233–261, 10 2020.
- [29] Christopher Schwarzmüller, Fadi Al Machot, Alireza Fasih, and Kyandoghere Kyamakya. Adaptive contrast enhancement involving cnn-based processing for foggy weather conditions non-uniform lighting conditions. *Proceedings of the Joint 3rd International Workshop on Nonlinear Dynamics and Synchronization, INDS'11 and 16th International Symposium on Theoretical Electrical Engineering, ISTET'11*, pages 37–46, 2011.
- [30] Hans-Peter Schöner. *Challenges and Approaches for Testing of Highly Automated Vehicles*, pages 101–109. 2016.
- [31] Hans-Peter Schöner. *Simulation in development and testing of autonomous vehicles*, pages 1083–1095. 2018.
- [32] Hao Shao, Letian Wang, RuoBing Chen, Hongsheng Li, and Yu Liu. Safety-enhanced autonomous driving using interpretable sensor fusion transformer. *arXiv preprint arXiv:2207.14024*, 2022.
- [33] CARLA Simulator. Maps - carla simulator. https://carla.readthedocs.io/en/latest/core_map/. (accessed Dec.21,2022).
- [34] CARLA Simulator. Sensors reference - carla simulator. https://carla.readthedocs.io/en/latest/ref_sensors/#lidar-sensor. (accessed Dec.21,2022).
- [35] M M Acn Reference Format: G ¨ Unal and M Pidd. Dghpsim: Generic simulation of hospital performance. *ACM Trans. Model. Comput. Simul*, 21, 2011.
- [36] Edward J. Williams and Onur M. ¨lgen. Simulation applications in the automotive industry. *Use Cases of Discrete Event Simulation: Appliance and Research*, 9783642287770:45–58, 11 2013.
- [37] Lingyun Xiao and Feng Gao. A comprehensive review of the development of adaptive cruise control systems. *Vehicle System Dynamics*, 48:1167–1192, 10 2010.
- [38] Weijia Yao, Wei Dai, Junhao Xiao, Huimin Lu, and Zhiqiang Zheng. A simulation system based on ros and gazebo for robocup middle size league. *2015 IEEE International Conference on Robotics and Biomimetics, IEEE-ROBIO 2015*, pages 54–59, 2015.

- [39] Ítalo Fernandes, Simara Rocha, Carlos Portela, Geraldo Braz Junior, João Almeida, Aristofanes Silva, Davi Viana, Jacilane Rabelo, Anselmo Paiva, and Luis Rivero. Defining an a/b testing process for usability and user experience evaluation through the analysis of the results of a literature review. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 13516 LNCS:204–213, 2022.

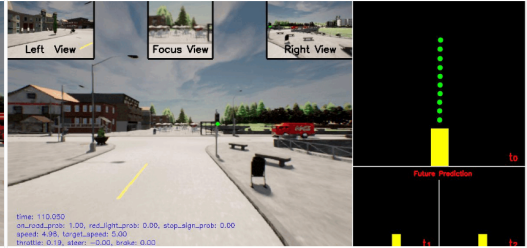
Appendix A

Dashboard

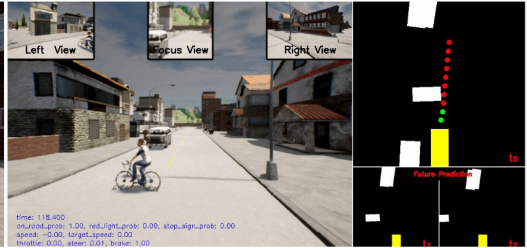
A.1 Scenario One Page



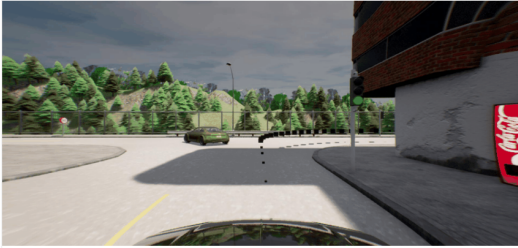
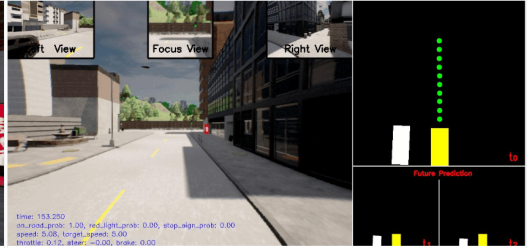
3 - Traffic Light

USER	ALGORITHM
 Sometimes people disregard traffic lights due to distractions or a rush	 As soon as the algorithm detects the red light, awaits for the green light to keep going

4 - Cyclist Crosses The Road

USER	ALGORITHM
 Pay attention to unexpected cyclist crossings Low visibility and higher speeds diminish your reaction time	 The algorithm makes the car go at a constant controlled speed and needs a lower reaction time to avoid the cyclist

5 - Traffic Light

USER	ALGORITHM
 Sometimes people disregard traffic lights due to distractions or a rush	 As soon as the algorithm detects the red light, awaits for the green light to keep going

6 - Cyclist Crosses The Road

USER	ALGORITHM
 Pay attention to unexpected cyclist crossings Low visibility and higher speeds diminish your reaction time	 Algorithm did not completed this step

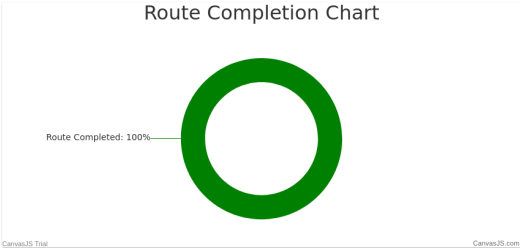
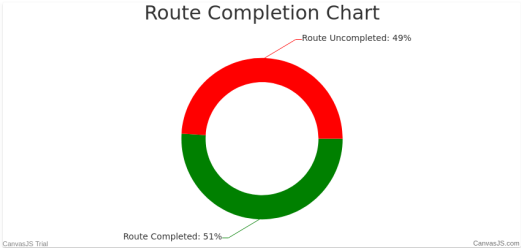
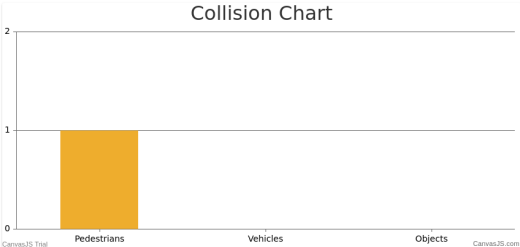
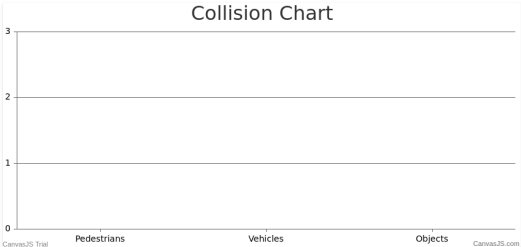
7 - Pedestrian Crosses The Road

USER	ALGORITHM
	
Pay attention to unexpected pedestrian crossings Low visibility and higher speeds diminish your reaction time	Algorithm did not completed this step

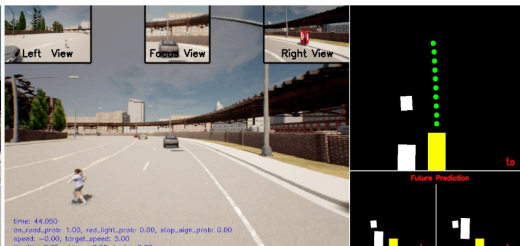
8 - Pedestrian Crosses The Road

USER	ALGORITHM
	
Pay attention to unexpected pedestrian crossings Low visibility and higher speeds diminish your reaction time	Algorithm did not completed this step

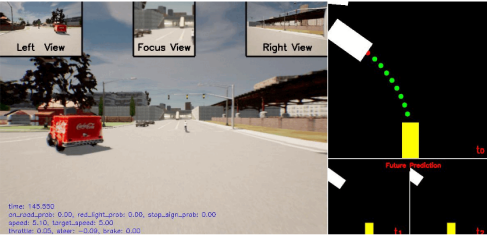
Stats

SCORE 50%	SCORE 51%
Route Completion Chart 	Route Completion Chart 
Collision Chart 	Collision Chart 
Crossed Red Lights 0	Crossed Red Lights 0

A.2 Scenario Two Page

SCENARIO ONE		SCENARIO TWO	FINAL RESULTS	QUESTIONAIRE
1 - Pedestrian Crosses The Road				
USER		ALGORITHM		
 Pay attention to unexpected pedestrian crossings Low visibility and higher speeds diminish your reaction time		 The algorithm makes the car go at a constant controlled speed and needs a lower reaction time to avoid the pedestrian		
2 - Pedestrian Crosses The Road				
USER		ALGORITHM		
 Pay attention to unexpected pedestrian crossings Low visibility and higher speeds diminish your reaction time		 The algorithm makes the car go at a constant controlled speed and needs a lower reaction time to avoid the pedestrian		
3 - Pedestrian Crosses The Road				
USER		ALGORITHM		
 Pay attention to unexpected pedestrian crossings Low visibility and higher speeds diminish your reaction time		 The algorithm makes the car go at a constant controlled speed and needs a lower reaction time to avoid the pedestrian		

4 - Traffic Light

USER	ALGORITHM
 Sometimes people disregard traffic lights due to distractions or a rush	 As soon as the algorithm detects the red light, awaits for the green light to keep going

5 - STOP

USER	ALGORITHM
 Stops signs can be easy to go unnoticed or ignored but they are essential to avoid dangerous situations	 The algorithm makes the car go at a controlled speed and makes it stop on the stop sign

6 - Cyclist Crosses The Road

USER	ALGORITHM
 Pay attention to unexpected cyclist crossings Low visibility and higher speeds diminish your reaction time	 The algorithm makes the car go at a constant controlled speed and needs a lower reaction time to avoid the cyclist

7 - STOP

USER	ALGORITHM
 Stops signs can be easy to go unnoticed or ignored but they are essential to avoid dangerous situations	 The algorithm makes the car go at a controlled speed and makes it stop on the stop sign

8 - Traffic Light

USER	ALGORITHM
 Sometimes people disregard traffic lights due to distractions or a rush	 As soon as the algorithm detects the red light, awaits for the green light to keep going

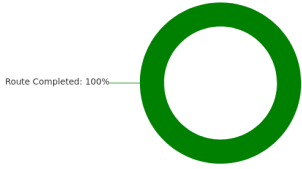
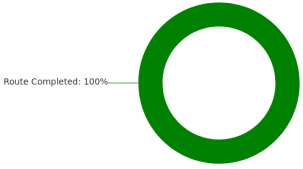
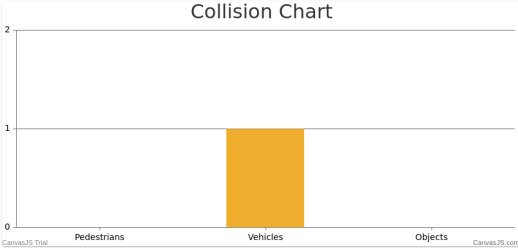
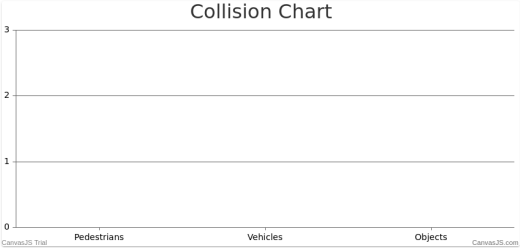
9 - Cyclist Crosses The Road

USER	ALGORITHM
 Pay attention to unexpected cyclist crossings Low visibility and higher speeds diminish your reaction time	 The algorithm makes the car go at a constant controlled speed and needs a lower reaction time to avoid the cyclist

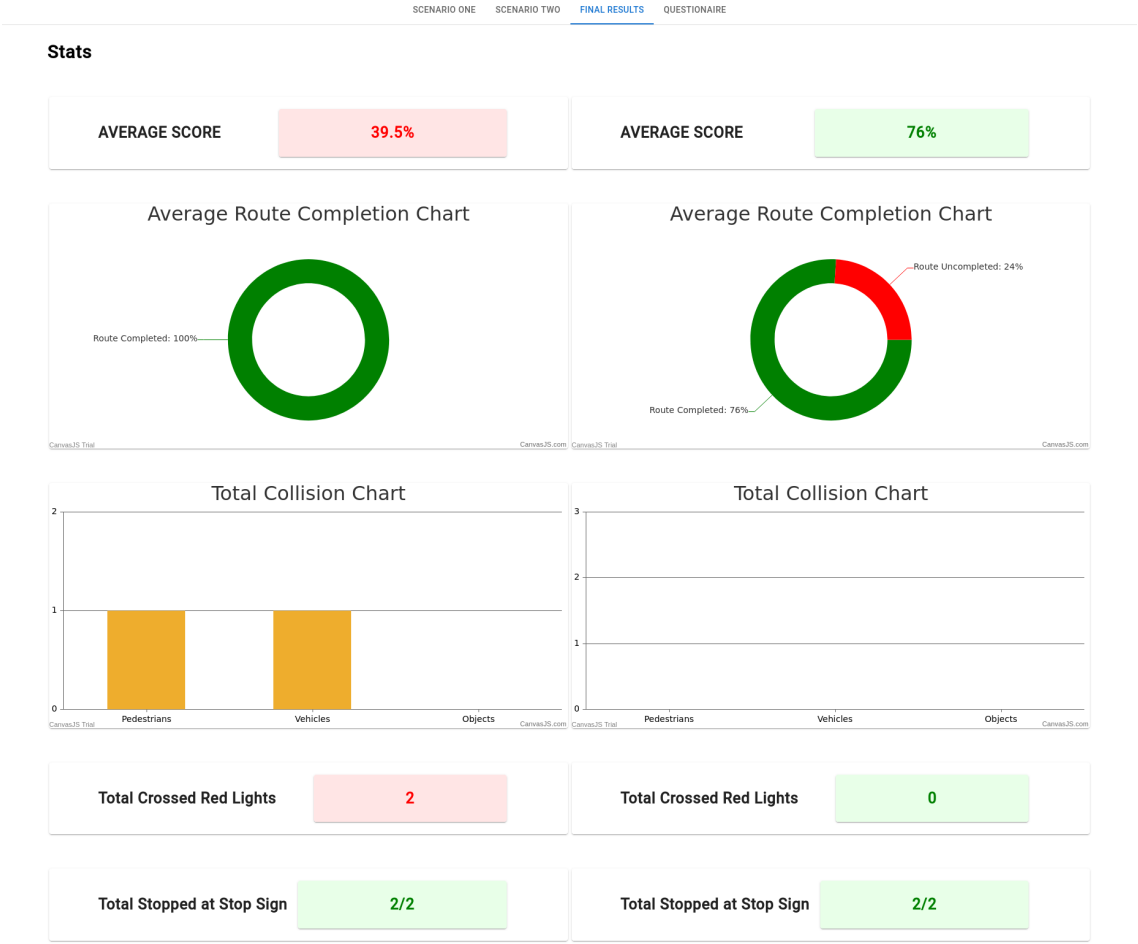
10 - Traffic Light

USER	ALGORITHM
 Sometimes people disregard traffic lights due to distractions or a rush	 As soon as the algorithm detects the red light, awaits for the green light to keep going

Stats

SCORE 29%	SCORE 100%
Route Completion Chart  Route Completed: 100%	Route Completion Chart  Route Completed: 100%
Collision Chart  Collisions: 1	Collision Chart  Collisions: 0
Crossed Red Lights 2	Crossed Red Lights 0
Stopped at Stop Sign 2/2	Stopped at Stop Sign 2/2

A.3 Final Results Page



A.4 Google Forms Page

SCENARIO ONE

SCENARIO TWO

FINAL RESULTS

QUESTIONNAIRE

ADAS Questionnaire

[Inicie sessão no Google](#) para guardar o seu progresso. [Saiba mais](#)

* Indica uma pergunta obrigatória

Pos-test Questionnaire

Como classifica os acontecimentos apresentados nos dois cenários em termos de realismo? *

1

2

3

4

5

nunca acontecem na vida real

☐

☐

☐

☐

☐

acontecem na vida real

Já esteve exposto a situações críticas enquanto condutor ou passageiro? Se sim, quais? *

☐ Passar semáforo no vermelho

☐ Peão atravessar a estrada num sítio inesperado e com pouca visão

☐ Ciclistas a atravessarem a estrada indevidamente

☐ Passar num stop sem parar

☐ Acidente com outro veículo

☐ Acidente com obstáculo

☐ Acidente com peão

☐ Nunca estive exposto a qualquer das situações anteriores

☐ Outra:

No cenário 1, obteve melhor pontuação que a condução com algoritmos ADAS? *

Appendix B

Forms

B.1 Informed Consent Form

Dear participant,

This experiment aims at communicating the benefits of ADAS through an interactive experience, where you will drive a car in a simulation with two scenarios and an AV algorithm with ADAS features will drive the same scenarios. Your performance indicators will be extracted and compared with the algorithm.

Your participation in this experiment is greatly appreciated. It is voluntary, anonymous and it only takes 35 minutes to complete. When you participate in this experiment, information from the forms filled by you is used to study the results.

This is an experiment intended for over 18 with driver's license only. The scenarios' events must not be shared with other participants before they have completed the experiment themselves.

Please feel free to contact us if you have any questions or concerns: up201805387@up.pt

Thank you for your interest in participating in the experiment.

Best regards,

Pedro Simões

I hereby consent to the processing of the personal data provided by me for the purpose of the experiment evaluation.

B.2 Pre-Test Questionnaire

- **Q1:** Age
- **Q2:** Years of driver's license
- **Q3:** Gender
- **Q4:** Level of education
- **Q5:** Do you know what are ADAS?
- **Q6:** What is your trust in ADAS? From 1 to 10, 1 being complete distrust and 10 complete trust.

B.3 Pos-Test Questionnaire

- **Q7:** Did you find the scenarios happenings realistic? From 1 to 5, 1 being that it never happens in real life and 5 being happens in real life.
- **Q8:** Have you ever been exposed to the following critical moments as a driver or passenger?
 - Run a red light
 - Pedestrian crossing the road in an unexpected place and short-sighted
 - Cyclists crossing the road improperly
 - Run a stop sign
 - Accident with another vehicle
 - Accident with obstacle
 - Accident with pedestrian
 - None of them
- **Q9:** Did you have a better score than the algorithm in scenario 1?
- **Q10:** Did you have a better score than the algorithm in scenario 2?
- **Q11:** Did you have a better score than the algorithm in the final results?
- **Q12:** Were the results presented in the dashboard clear? From 1 to 5, 1 being didn't understand anything and 5 being understood everything.
- **Q13:** How do you rate this experience in terms of communicating the benefits of driver assistance systems? From 1 to 7, 1 being poorly communicated and 7 being extremely well communicated.

- **Q14:** What is your level of ADAS trust after the experiment? From 1 to 7, 1 being decreased a lot and 7 being increased a lot.
- **Q15:** If you answered lower than 4 in either of the last two questions, explain why.
What can be improved to provide a better communication and trust of ADAS?