AD-EYE
Towards Autonomous Driving on KTH Campus
SOTA report

MF2058 VT22 period 4

Malin Dyberg
Axel Hedvall
Johan Hultenheim
Óscar Poveda Ruiz
Shaya Rahmanian
Gowtham Rangaraju
Elvira Troillet
Ehab Ziad Raheem

May 23, 2022

Abstract

AD-EYE is an ongoing collection of projects at Kungliga Tekniska Högskolan (KTH) in Sweden, aiming to develop platforms and systems for autonomous driving. The goal of the project presented in this report is to implement the AD-EYE software on a commercial car intended for driving on public roads. The proper hardware required for autonomous driving will be integrated to the car, using appropriate mountings and interfaces. An NVIDIA DRIVE™ PX 2 computer will be used as the central unit, and serve both as the platform for the software and as the channel between the software and hardware. Additionally, communication will be established with a Road Side Unit (RSU), developed in a previous project in the AD-EYE suite, in order to demonstrate the possibilities of the integration of smart infrastructure to the AD-EYE platform. Furthermore, the scope includes the realisation of several driving scenarios, where the performance of the car will be finally tested and evaluated.

This report covers the State of the Art (SOTA) study of the project. The relevant background information as well as similar projects and existing solutions of partially autonomous vehicles are covered, followed by a careful investigation of the relevant facts for the execution of the project. The related AD-EYE projects are also addressed, with emphasis on the projects that are relevant for the accomplishment of this project. Finally, based on the knowledge gained in this SOTA study, potential designs of the autonomous car are presented, including sensor placement and mounting as well as the integration and interfacing to the NVIDIA DRIVE™ PX 2 and RSU.

Keywords: Autonomous, Driving, NVIDIA DRIVE PX 2, AD-EYE, ADI, RSU, Sensors

Contents

List of Figures

List of Abbreviations

ACC	Adaptive Cruise Control	10
AD	Automated Driving	2
ADAS	Autonomous Driving Assistance Systems	5, 7
ADI	Autonomous Driving Intelligence	13
AI	Artificial Intelligence	6, 8
ALV	Autonomous Land Vehicle	1
APU	Application Processing Unit	14
C-V2X	Cellular V2X	9, 10
CAN	Controller Area Network	1–3, 7, 8, 17–20
CAN-FD	CAN with Flexible Data Rate	7
CAV	Connected and Autonomous Vehicles	6, 7
CPU	Central Processing Unit	8, 17
CU	Controller Unit	14
DARPA	Defense Advanced Research Projects Agency	1
DGPS	Differential GPS	6
DNN	Deep Neural Network	8
DSRC	Dedicated Short-Range Communications	2, 9, 14
ECU	Electronic Control Unit	7
EU	European Union	9
FSD	Full Self-Driving	6
GLONASS	Global Navigation Satellite System	11
GMSL	Gigabit Multimedia Serial Link	8
GNSS	Global Navigation Satellite Systems	2, 10, 11, 14–16
GPS	Global Positioning System	11
GPU	Graphics Processing Unit	8, 14
GUI	Graphical User Interface	3, 4
HIL	Hardware-In-The-Loop	3
IEEE	Institute of Electrical and Electronics Engineers	9
IMU	Inertial Measurement Unit	1, 2, 11, 13, 14, 16, 20
INS	Inertial Navigation System	10
ITRL	Integrated Transport Research Lab	1, 14
ITS	Intelligent Transportation Systems	2
ITU	International Telecommunications Union	10
KTH	Kungliga Tekniska Högskolan	1, 2, 13, 14
LiDAR	Light Detection And Ranging	1, 2, 5, 6, 10, 12–17, 20

MU-MIMO	Multi-User Multiple-Input Multiple-Output	9
OBD	On-Board Diagnostics	7, 18
OBU	On Board Unit	14
OFDMA	Orthogonal Frequency Division Multiple Access	9
OSI	Open Systems Interconnection	7
RCV	Research Concept Vehicle	13, 14
ROS	Robot Operating System	2, 13, 18
RSU	Road Side Unit	1–4, 8, 9, 14, 17, 18, 20
RTOS	Real-Time Operating System	8
SAE	Society of Automotive Engineers	1, 4
SoC	System on a Chip	8
SOTA	State of the Art	2, 5, 14
TRL	Technology Readiness Level	2
UART	Universal Asynchronous Receiver-Transmitter	8
UN	United Nations	2
USB	Universal Serial Bus	8, 17
V2X	Vehicle-To-Everything	1–3, 8, 9, 14
WASP	Wallenberg AI, Autonomous Systems and Software Program	1
WLAN	Wireless Local Area Network	8, 9

1 Introduction

Since 2017, the AD-EYE testbed for autonomous driving has been developed at KTH in Stockholm, Sweden [1]. This report covers the implementation of the AD-EYE system on a real car intended for driving on public roads at KTH Campus. The goal for this car is to serve as a realistic testbed for research in autonomous vehicles.

1.1 Scope

The scope of this project covers the integration and implementation of external sensors and the existing AD-EYE software on a commercial vehicle. The aim is to successfully accomplish autonomous driving scenarios, e.g., attaining the vehicle to autonomously follow a predefined trajectory and simulating an non-existing object to the vehicles perception. Integration of the external sensors includes mounting of two to three cameras, a LiDAR and an Inertial Measurement Unit (IMU) on optimal places on the vehicle. The goal is to gather sensor data and provide it to an existing perception system that perceives its surroundings, in order to allow the vehicle to navigate through the roads at KTH Campus.

An intelligent infrastructure will be implemented by establishing communication between the vehicle and a RSU[1]. The RSU will act as a Vehicle-To-Everything (V2X)[2] server by receiving and transmitting information from and to the vehicle. An on board computer, i.e., the NVIDIA DRIVE™ PX 2, will be used in order to control the vehicle through the Controller Area Network (CAN) protocol. The on board computer will be an essential component for interfacing the communication between the sensors, the AD-EYE software, the CAN system and the RSU.

The specifications of the desired driving scenarios and other stakeholder requirements are specified in detail in section 1.4.

1.2 Background

The first concept of self-driving cars was introduced in the 1920s with Houdina Radio Control, demonstrating the radio-controlled "American Wonder" on New York City streets [2]. The first self-sufficient autonomous cars appeared in the 1980s, through projects such as the Autonomous Land Vehicle (ALV) funded by Defense Advanced Research Projects Agency (DARPA) and Navlab by Carnegie Mellon University [3]. At present, the topic of automated vehicles is still one of the most promising research areas in the automotive industry [4].

Despite claims that self-driving cars will arrive in the near future, fully automated self-driving cars are still not market-ready [5]. Instead, most manufacturers offer systems that feature capabilities to support driving activities, such as parking assistance, lane keeping assistance and collision avoidance. Such capabilities require a human driver at the wheel, thus falling within the first three of the six levels[3] of autonomy as defined by the Society of Automotive Engineers (SAE). This might be due to the technological hurdles that needs to be improved, i.e., development of the hardware or optimization of the decision making algorithms, in order to improve the performance of the vehicle. The aim should be to reach an autonomous driving experience that surpasses the performance of human drivers in any scenario, in order to ensure reliable autonomous driving. Furthermore, self-driving car algorithms require rigorous and comprehensive testing which is achieved by either physical testing or simulation testing [6].

Integrated Transport Research Lab (ITRL) and Wallenberg AI, Autonomous Systems and Software Program (WASP) are establishments and programmes within KTH actively focusing on the field of self-driving cars research. In particular, the AD-EYE Open Modular Testbed is one of three projects launched by WASP as part of it's new Industry Bridge Instrument. The goal of AD-EYE is to aid researchers to

[1] See section 3.4
[2] See section 2.3
[3] See section 2.1

overcome hurdles in making the transition from lower to mid/high Technology Readiness Level (TRL), especially challenging for research in safety critical fields such as Automated Driving (AD) and Intelligent Transportation Systems (ITS).

The AD-EYE platform is built upon Robot Operating System (ROS) (middleware), PreScan (Worlds/Environment), Autoware (Nominal Channel) and KTH Safety Channel. Within the Nominal Channel exists the core decision making algorithm that relies upon the perception node to plan actions. Perception within the AD-EYE is enabled through automotive sensors such as cameras, LiDAR, radars, IMU and Global Navigation Satellite Systems (GNSS). Beyond scene perception, V2X technology is used to achieve safer driving standards by enabling cars to communicate with their surroundings [7]. The RSU acts as V2X server which is also a Dedicated Short-Range Communications (DSRC) receiver that is placed where decision-making in traffic is difficult and crucial, for example at a pedestrian passageway or intersection.

1.3 Ethics and Sustainability

In the 11th goal of the 2030 agenda for sustainable development by the United Nations (UN), it is stated that cities and human settlements should be inclusive, safe, resilient and sustainable. For instance, by 2030 access to safe and sustainable transport systems should be available for all humans. Furthermore, the 13th goal describes the urgency of combating the climate change and its impacts [8]. The development of autonomous vehicles is a promising step in the direction of fulfilling these goals. A higher level of automation on public roads can lead to greater road safety by reducing road accidents and deaths. In addition, autonomous vehicles provide the possibility of greater inclusion, since it gives people of different abilities and disabilities equal possibility of travelling by car. Finally, the development of smart infrastructure has the potential of reducing emissions as a result of fewer traffic congestions and smarter planning [9].

The 11th goal can further be discussed from an ethical perspective. A fully autonomous vehicle would have to replace the human driver in a decision making process. In some situations, abiding by the laws of traffic might not be the safest decision, and in some situations there might be no decision that ensures safety for all people. How does one create an algorithm for ethical decisions, and how does one make these algorithms treat all people equally? Who decides which human has the right to be safe, without discriminating on the basis of e.g., age or disabilities [10]? The moral questions are many. However, the steps to solve any ethical dilemma related to self-driving cars is beyond the scope of this project.

1.4 Requirements

This section covers all the stakeholder requirements as provided by the stakeholder of this project. The provided requirements were divided into three levels of completion based upon the testing environment listed as follows:

1. Initial scenarios including performance evaluation (metrics and measurements): in lab
2. Intermediate scenarios (sensing, feed forward controls): the real world
3. End scenarios: autonomous driving scenarios.

Additionally, the technical requirements to accomplish the stakeholder requirements have been stated.

1.4.1 Stakeholder Requirements

1. Initial scenarios

    1.1. CAN shall be interfaced to the vehicle and verify vehicle controls.

1.2. All provided sensors shall be integrated optimally on-vehicle, that is, mounting, power and electrical interface.

1.3. All provided sensors shall be logically interfaced to AD-EYE.

1.4. V2X communication shall be enabled between the RSU and the vehicle using 5G and a selected wireless communication protocol.

1.5. AD-EYE shall be interfaced towards the simulated vehicle, that is, AD-EYE on NVIDIA DRIVE™ PX 2 within a simulator environment.

1.6. AD-EYE shall be interfaced towards vehicle in a Hardware-In-The-Loop (HIL) mode, that is, AD-EYE on NVIDIA DRIVE™ PX 2 with CAN interfaced to the vehicle.

2. Intermediate scenarios

2.1. The AD-EYE Graphical User Interface (GUI) shall be developed to visualize the vehicle state, environment through external sensors and communications.

2.2. The vehicle shall be controlled via "open loop controls" through the AD-EYE GUI by a cabled connection.

2.3. Localization shall be tested by mapping the driving area and integrating it so that both the localization map and road network map are available.

2.4. The vehicle must be able to drive autonomously along a short predefined trajectory.

2.5. The vehicle must be able to drive autonomously to a given goal.

3. End scenarios

3.1. The vehicle must safely stop when an emergency signal is received from the RSU over V2X.

3.2. The vehicle must be able to react and navigate around a non-existent object introduced into the vehicle's perception by the RSU.

3.3. The vehicle must be able to enter follow mode and match its speed to a non-existent leading vehicle introduced into the vehicle's perception by the RSU.

1.4.2 Technical requirements

The given stakeholder requirements are divided into technical requirements as listed below. The goal of this list below is to simplify stakeholder requirements into potential milestones and understand inter dependencies between tasks.

1. CAN

1.1. Only wires designated for CAN operation by the sensor manufacturer shall be used for interfacing to the CAN bus.

1.2. CAN shall only be setup as per instructions from NVIDIA DRIVE™ PX 2 documentation and sensor manufacturer's documentation.

1.3. Any errors and bugs caused by CAN shall be logged appropriately for debugging, optimization and maintenance purposes.

2. Sensor Integration - Mounting

2.1. The mounted sensors shall not cause any damage to the car.

2.2. The sensor mount shall protect the sensors from any external forces and vibrations that might harm them.

2.3. The wires leading from the sensors to the NVIDIA DRIVE™ PX 2 shall not be overly restricted such that they might be subject to excessive wear and tear.

2.4. The sensors shall be mounted in their optimal location on board the vehicle such that they have:

i. zero to minimal signal or noise interference from other sensors and
ii. unrestricted field of view within the sensor's capabilities.

3. Sensor Integration - Power

3.1. The power supplied to the sensors shall abide by the sensor specifications of input voltage and current.

3.2. Any custom wires installed for power delivery shall be rated to carry the load and include a safety margin.

4. Sensor Integration - Electrical Interfacing

4.1. All power wires shall be tagged and colour coded as per SAE automotive wiring standards.

4.2. All wires shall be connected securely without any loose contact at the terminals.

4.3. All wires shall not expose the bare conductor throughout the wire.

4.4. All wires shall be routed efficiently such that they:

i. do not obstruct any passengers or driver's field of view and
ii. ensure voltage drop is within permissible levels.

4.5. All power wires and signal wires shall be separately bundled within the wire harness.

5. Sensor Integration - Logical Interfacing

5.1. All codes related to sensor data interfacing shall follow internationally recognized coding conventions.

5.2. All codes related to sensor data interfacing shall be committed and pushed to the designated GitHub along with the coding process documentation.

5.3. All sensor data shall be tested against sensor manufacturer's specifications to ensure data quality and performance of sensor.

5.4. Any errors and bugs experienced by the sensor shall be logged appropriately for debugging, optimization and maintenance purposes.

6. AD-EYE

6.1. The core AD-EYE algorithms shall not be directly modified.

6.2. All codes and progress related to AD-EYE GUI development shall be documented, committed and pushed to the designated GitHub.

6.3. Any errors and bugs experienced by the AD-EYE during the interfacing shall be logged appropriately for debugging, optimization and maintenance purposes.

7. RSU

7.1. The RSU shall communicate with the vehicle using 5G and a selected wireless communication protocol.

7.2. The core logic of the RSU shall not be modified directly and be only treated as a point of communication between the cloud and the vehicle.

7.3. Any code written for the RSU shall follow internationally recognized coding conventions.

7.4. Any code written for the RSU shall be committed and pushed to the designated GitHub along with the coding process documentation.

7.5. Any errors and bugs experienced by the RSU that affects the AD-EYE or the vehicle shall be logged appropriately for debugging, optimization and maintenance purposes.

2 State of the Art

The following section presents the conducted SOTA for this project, which covers the existing solutions of partially autonomous vehicles and testbeds. Additionally, the required hardware and communication technologies for executing this project are included.

2.1 Existing Solutions

The existing solutions of autonomous vehicles will be generalised into two different segments:

1. *Partially Autonomous Vehicles*, which refers to the commercial availability of partially autonomous vehicles that exist on the market today.
2. *Testbeds*, which refers to vehicles developed for testing, verifying and validating technologies used for autonomous driving.

Both partially autonomous vehicles and testbeds may be equipped with different levels of automated driving. In order to distinguish these different levels, the grading scale Levels of Driving Automation may be used, as shown in Figure 1. The figure shows a range of increasing levels of driving automation, ranging between level 0 (L0) and level 5 (L5). The figure also clearly states the meaning of each level [11].

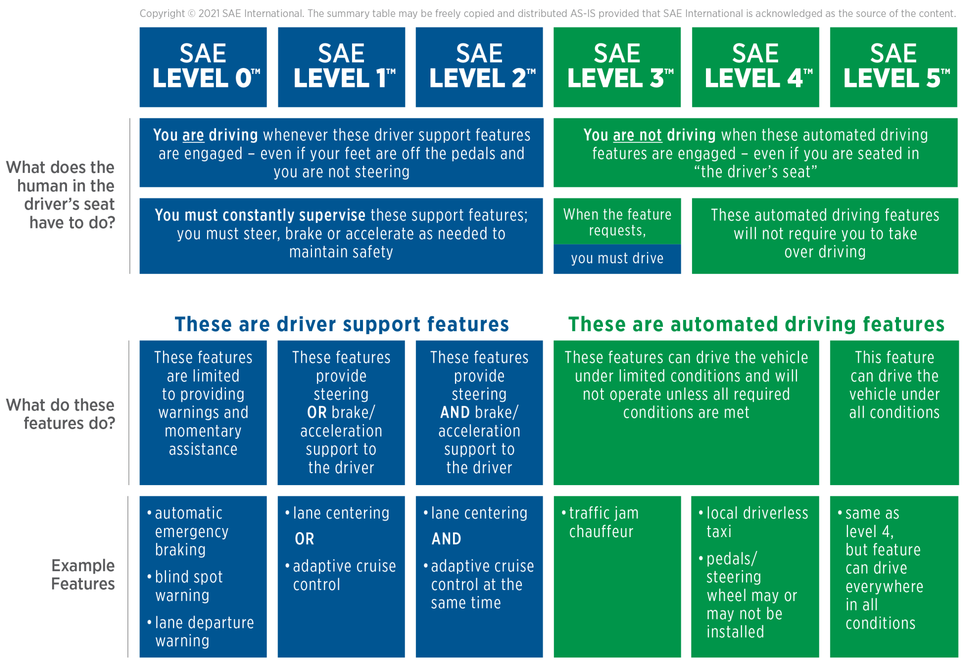


Figure 1: The various Levels of Driving Automation, ranging between level 0 (L0) and level 5 (L5) [11], edited with [12].

2.1.1 Partially Autonomous Vehicles

Different driver assistance systems are becoming increasingly ubiquitous in modern vehicles. These systems are used to increase performance and safety in various car related activities, such as parking, lane keeping, traffic sign identification and collision avoidance. By combining Autonomous Driving Assistance

Systems (ADAS) with some form of Artificial Intelligence (AI), the automotive industry is developing automated technologies that are evolving towards becoming more autonomous and requiring less user interaction [13].

Currently, many well-known car manufacturers are involved in development of vehicles equipped with some level of driving automation. Several manufacturers are already supplying vehicles equipped with L1 and L2 capability for automated driving, such as Volvo with the Pilot Assist [14], BMW with the Driver Assistance Package [15] and Ford with the Co-Pilot360™ [16]. Features such as lane keeping assistance, collision avoidance, evasive steering assist, hill descent control and cruise control are categorised as L1 and L2 capabilities. These features make use of sensors, microcontrollers and algorithms to produce the desired output [16].

The Tesla line of vehicles is yet another example of vehicles with L2 capability for automated driving. The Tesla AutoPilot™ technology takes input from surrounding environments using eight cameras, one radar and twelve sonars [17]. The vehicle is also equipped with Tesla's self developed computer, the Full Self-Driving (FSD) chip. The chip has been developed with the aim of eventually equipping their line of vehicles with higher levels of capability for automated driving [18]. According to Tesla, their line of cars provide a fully self-driving experience. However, the technology in Tesla cars still requires overseeing by a driver and erratic behaviour during driving has been reported by their own users [19].

Currently, there exists manufacturers that offer vehicles with L3 capability, such as Audi and Honda [5]. However, reaching higher levels of driving automation is still limited by safety regulations, performance issues in adverse conditions and a multitude of technical difficulties that inhibit the research and commercialization of fully L4 or L5 capability [21]. Figure 2 shows an example of how a research vehicle may be designed.

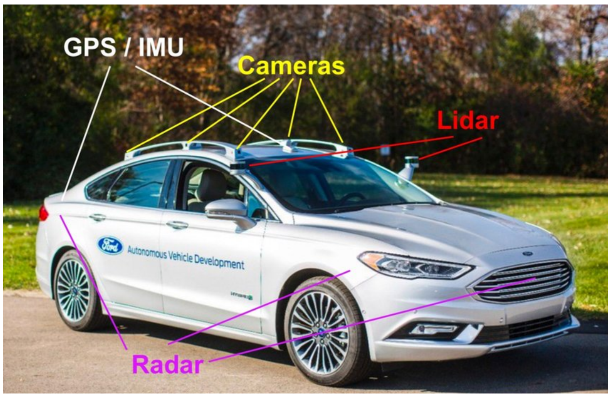


Figure 2: Ford's research autonomous vehicle [20].

A few companies provide a robotaxi experience [22], such as Waymo and Autox [23]. The Waymo line of vehicle is equipped with a LiDAR, about 19 to 29 cameras (depending on the vehicle model), a radar and an on board computer responsible for decision making, trajectory planning and processing sensor data [24]. Waymo has been granted permit in the US to offer a robotaxi service in a few states to a limited amount of users [25].

2.1.2 Testbeds

Testbeds are used during development of autonomous vehicles for implementing rigorous and comprehensive testing. Simulation and physical testing in specific conditions are crucial for ensuring proper behaviour and quality of both hardware and software. The vehicle must go through extensive testing in order to assure that it complies with the existing demands, laws and regulations related to autonomous driving. Although testbeds for autonomous driving are rare in general, there are a few establishments that are developing testbeds for autonomous driving. Development of testbeds is being driven forward in order to satisfy the need of a safe environment where testing can be executed with accuracy and close similarities to real world situations [26].

AstaZero is a testbed located near Göteborg in Sweden. It has been stated to provide a rural road lane, a high speed area and an indoors dry zone for testing in optimal conditions. The testbed provides bi-directional, right-hand traffic and a road width of 7 metres. The infrastructure has been developed to support a myriad of modules for the test vehicles, such as communication resources, Differential GPS (DGPS), dummies and wireless coverage [27].

Another testbed of note is Millbrook, located in Culham in the UK. Development of Connected and

Autonomous Vehicles (CAV) technologies and its implementation using 5G is emphasized at Millbrook to test and validate ADAS. The testbed also contains a facility called the Autonomous Driving Village, where the development of the aforementioned CAV takes place [28].

There are many more upcoming testbeds, like Tesla's Gigaberlin [29], TiHaN-IIT in India [30] and Hyundai in Namyang R&D Centre in South Korea [31]. Universities are expanding their research into autonomous vehicles as well, with Oxford creating the Selenium software system with the goal of uploading it into any vehicle with the right type of sensors. The University of Michigan is creating a large model of a city that will serve as a testbed to evaulate how autonomous driving will affect urban planning [32]. As the future of autonomous vehicles approaches, there is a growing demand to create environments where vehicles can be safely tested in simulations that are close to reality.

2.2 On Board Resources

This section covers the resources available on board the car, along with their integration strategy.

2.2.1 CAN

The CAN protocol is the internationally dominating serial network protocol for embedded control systems in passenger cars [33]. The protocol covers the lower layers of the Open Systems Interconnection (OSI) seven-layer model for data communication, including the data link layer and the physical layer. However, the medium of data transportation and the interfaces depends on the application [34]. In a vehicle, the various electrical systems are controlled by the Electronic Control Unit (ECU)s. These devices receives information from e.g., sensors and commands the various actuating systems to perform the proper action [35]. The CAN bus connects these ECUs in a physical network, resulting in all ECUs being able to broadcast and receive information on a single bus [36]. For the data communication, the CAN protocol uses a non-destructive arbitration scheme, where higher priority messages can be transmitted on the bus without interruption or notion of collisions [37]. All together, this makes the CAN system extremely robust and efficient and hence the defacto standard for communication in the majority of the vehicles today [36].

CAN provides the base of the communication in a vehicle, but higher-layer protocols are necessary in order to further specify how the data is transmitted over the network and how to interpret and use the data. Several standard higher-level protocols exists, of which On-Board Diagnostics (OBD) is the most commonly used in cars [38].

However, the original CAN protocol has limitations in terms of performance and communication bandwidth. The serial network protocol CAN with Flexible Data Rate (CAN-FD) was hence developed to meet the demands of complex electrical systems in the automotive industry. CAN-FD has, compared to the original CAN protocol, increased size of data field and higher communication bandwidth. CAN-FD supports a data field up to 64 bytes and a bandwidth up to 8 Mbps [39]. The protocol is internationally standardised, as well as the original CAN protocol. The CAN-FD protocol enables higher speeds compared to CAN, and is thus more suitable for meeting the increasing complexity of requirements on bandwidth and higher data transmission rates. In order to obtain the faster performance, the CAN-FD protocol will increase the bitrate when only one node is transmitting. The nodes are then re-synchronized before the transmission is resumed [40].

2.2.2 On Board Computer

To handle the additional sensor inputs and calculations required for an autonomous vehicle, a dedicated computer is required. The NVIDIA DRIVE™ PX 2 is a computer from NVIDIA Corporation designed with this in mind. The NVIDIA DRIVE™ PX 2 is the second generation of hardware for NVIDIA DRIVE™, a computing platform for autonomous driving. NVIDIA DRIVE™ leverages artificial intelligence and deep learning for the heavy computational tasks required for autonomous driving [42]. For the rest of the report, NVIDIA DRIVE™ PX 2 will be referred to as ’PX2’ only.

The PX2 is based on the Pascal microarchitecture and is available in two configurations:

- AutoCruise with one Tegra X2 System on a Chip (SoC)
- AutoChauffeur with two Tegra X2 SoCs

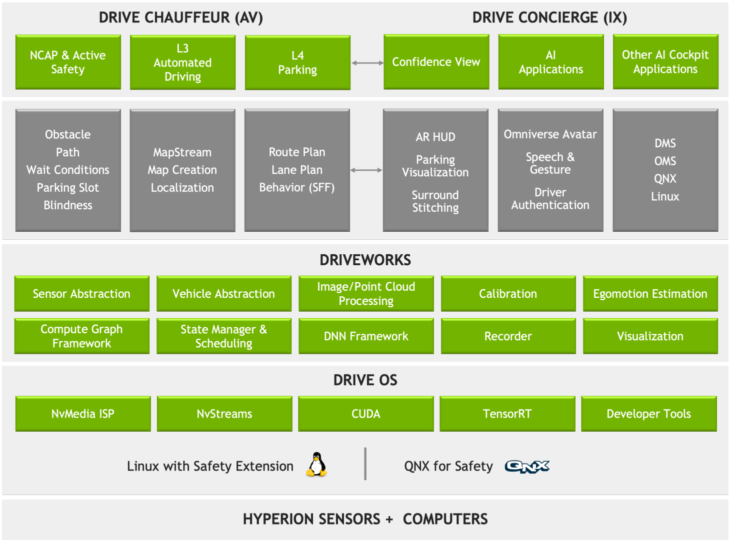


Figure 3: The software layers of NVIDIA DRIVE™ [41].

Each Tegra X2 SoC contains a Denver 2 dual-core Central Processing Unit (CPU), an ARM Cortex-A57 quad-core CPU, and a Pascal Graphics Processing Unit (GPU) [43]. The PX2 includes several relevant hardware interfaces including CAN, Universal Asynchronous Receiver-Transmitter (UART), Gigabit Multimedia Serial Link (GMSL) camera inputs and HSAutoLink II. Interfaces found on personal computers are also available on the PX2s, such as display outputs, Ethernet, and Universal Serial Bus (USB) [44]. The less powerful of the two, AutoCruise, is mainly designed to aid in driving by handling automated driving on highways and mapping. The more powerful, AutoChauffeur, can handle point-to-point driving. If more computational power is required, multiple systems can be interconnected [45].

NVIDIA DRIVE™ does not only offer hardware solutions, but also software, ranging from a simulation environment [46] to a virtual assistant and fully autonomous point-to-point driving [47]. The software is built of layers on top of each other, as can be seen in Figure 3. At the core is DRIVE OS, an embedded Real-Time Operating System (RTOS) that is designed to be safe to use in-vehicle. DRIVE OS includes libraries to interface with the NVIDIA DRIVE™ hardware, as well as different deep learning libraries. The middleware DriveWorks is implemented on top of DRIVE OS in order to interface with sensors using different tools and libraries. DriveWorks also provides a Deep Neural Network (DNN) framework. The next layer in the NVIDIA DRIVE™ software is DRIVE AV. DRIVE AV can handle the rest of the tasks required for point-to-point autonomous driving. This includes pre-trained DNNs for mapping, including perception, and planning. DRIVE AV also includes NVIDIA Safety Force Field™, a computational framework for safety on the road. The NVIDIA DRIVE™ software also provides DRIVE IX, an interior AI for occupant monitoring, and DRIVE Concierge, a virtual assistant [41].

2.3 V2X Communication

V2X communications refers mainly to the group of technologies and standards that can be used to establish a communication between a vehicle and the surrounding environment including other vehicles, pedestrians, networks and infrastructures such as RSUs[1]. Using V2X enables different vehicles to share various sensor data in order to avoid collisions or traffic congestion, which creates a safer and a more efficient transport environment. Depending on the chosen set of standards of V2X, different wireless communication technologies can be implemented. Latency, coverage, reliability and power consumption are examples of factors that should be taken into consideration when choosing the proper communication technology [48]. However, the need of transmitting heavy amount of data narrows down the choices to either Wireless Local Area Network (WLAN) or cellular technologies.

[1]See section 3.4

2.3.1 WLAN

WLAN represents a group of devices and computers that are connected to each other using radio transmissions. One of the most known WLAN technologies is WiFi, which refers to a class of certified wireless networking solutions that comply with the Institute of Electrical and Electronics Engineers (IEEE) 802.11b industry standard [49]. However, due to the fact that the previous WiFi versions are no longer able to meet the latency and coverage requirements for autonomous vehicles, only latest WiFi technologies, namely WiFi 5 and WiFi 6, are taken into consideration.

WiFi 5 is a user friendly word for the IEEE 802.11ac standard. WiFi 5 uses a frequency of 5 GHz, and it offers a maximum throughput of 7 Gbps across different channels. On the other hand, IEEE 802.11ax standard, known as WiFi 6, uses a frequency of either 2.4 or 5 GHz and is optimised for a better performance in large outdoor implementations. Furthermore, WiFi 6 offers reduced power consumption and an up to 9.6 Gbps data rate through multiple channels by using more advanced features such as Orthogonal Frequency Division Multiple Access (OFDMA) and Multi-User Multiple-Input Multiple-Output (MU-MIMO) [50].

Put to the test, WiFi 6 provided up to 75% less latency resulting in significant higher speed than WiFi 5 [51]. However, the performance of the used WiFi technology varies strongly depending on the location of the user, used devices and the local radio environment.

2.3.2 Cellular Telecommunication

Cellular Telecommunication is based on the usage of a large number of radio stations, where each station is able to communicate with a large number of devices in a certain small area. Furthermore, these stations have the ability to communicate with each other, which establishes a huge amount of communications between different devices [52].

The latest cellular telecommunication technology, namely 5G, is expected to be able to address the problem of reliability, latency and peak throughput per connection. Due to its high use of virtualization of network infrastructure, 5G provides a reliability of almost 100% and an end to end latency of less than 1 ms with a data rate up to 100 Mbps. This is done by making a better use of edge computing technologies compared to previous cellular telecommunication technologies [50].

While the 5G technologies provide a solution to a faster communication, it requires access to large amounts of frequency spectrum meaning that both coverage and speed can vary strongly depending on the available frequency spectrum. A frequency spectrum between 1 GHz to 6 GHz allows for transmitting a good amount of data through a significant distance. On the other hand, a frequency spectrum from 24 GHz, called the millimeter wave spectrum, is able to transmit huge amounts of data, being more suitable for vehicle communication. However, millimeter waves can easily be absorbed by gases and rain, and it can be impeded by buildings and trees which makes the coverage distance for this spectrum short [53].

2.3.3 ITS-G5 and C-V2X

There are two existing solutions for implementing wireless communication technologies to establish a reliable communication for vehicles. The first solution is DSRC, also known as ITS-G5 and based on the IEEEs 802.11p technologies. This solution introduces the standards to exchange wireless short-range information messages between the vehicle and the RSU at a certain location. The other solution is based on cellular communication technologies and is known as the Cellular V2X (C-V2X). Using this standard, a vehicle is able to communicate with a base station to connect to the surrounding environment as well as to information bases. Both of these technologies use a radio spectrum band of 5.9 GHz which is allocated by the European Union (EU) commission to be exclusively used for V2X based applications [54].

ITS-G5 uses the radio spectrum in a simple and efficient way to obtain a robust V2X short-range communication. However, it suffers from a great deal of challenges including limited mobility support, poor coverage range, latency and reliability. This has raised the need to develop C-V2X to address these problems by leveraging existing cellular network infrastructures. C-V2X provides a reliable, high-speed communication by combining short-range direct communication and long-range network communication [54].

2.4 Sensors

In the following subsections relevant sensors for the project will be presented.

2.4.1 LiDAR

A LiDAR, see Figure 4, is a device that determines the distance to an object or a surface by using a laser emitter and a pulsing laser beam. The distance to the object is determined by measuring the time delay between the emission of the beam from the laser emitter to the detection through the reflected signal.

With the information of the pulsing laser beam and the use of a differential GNSS system and an Inertial Navigation System (INS), tens of thousands of points per second are obtained, forming point clouds [2], which are useful for navigating in surrounding environments [56].

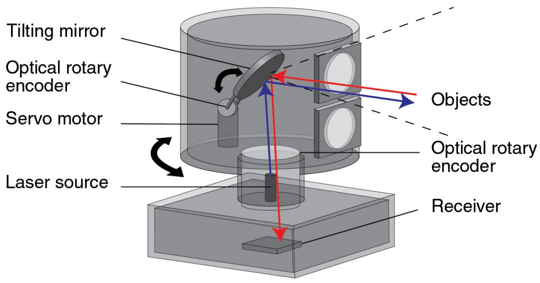


Figure 4: LiDAR components and operation [55].

2.4.2 Camera

Cameras are widely used on autonomous vehicles to detect surrounding objects. 3D cameras are capable of capturing data at high resolution. Two types of cameras are commonly used in autonomous vehicles: mono-cameras and stereo-cameras. A camera with only one sensor capturing video that has to be processed is called monocular system. If the system has two separated cameras it is a stereo-vision system [57].

The distance to an object can be obtained by determining the projection difference of the object of two stereo cameras. The maximum distance range and depth that a camera is capable to detect depends on different parameters, such as the focal length of the cameras, the baseline and the density of pixels of the camera. In addition to defining the distance to the objects, the images of the cameras have to be processed to interpret the detected object, and these computations are typically very heavy. Moreover, the processing has to occur in real time which adds difficulty to the implementation of cameras in a system [58].

2.4.3 Radar and Ultrasonic Sensors

Radars are sensors that send and receive electromagnetic waves to detect and locate objects in the environment. As vehicles are good reflectors, radars can be used to obtain the position and distance of the other surrounding vehicles.

The International Telecommunications Union (ITU) defines two categories of radar systems for vehicles. The first one enables an Adaptive Cruise Control (ACC) and collision avoidance in ranges up to 250m.

[2] See section 3.2

The second category adds functionalities for passive and active safety like pedestrian detection in a range between 50 and 100m. It is possible to find three types of radars depending on the maximum distance range and the field of view. In order to obtain 360° coverage, radars should be placed on several sides of the car [59].

Ultrasonic sensors uses echolocation by transmitting high-frequency sound waves to obtain the distance to nearby objects. They can combine information with multiple sensors such as LiDARs or cameras of the vehicle to obtain full data information of the objects around the car. Ultrasonic sensors are useful for detecting objects at close distances that are moving at low velocities. They are important for increasing visibility in the close surroundings of the car. [60].

2.4.4 Inertial Measurement Unit

An IMU is a device that tracks the orientation, position and acceleration of an object. This is done with accelerometers, gyroscopes and sometimes also magnetometers, which tracks the change of the Earth magnetic field in order to perceive its movement. IMUs are often used in inertial positioning systems where the sensor data from the IMU is used to determine its position relative a global coordinate system. One specific use case could be to aid the GNSS system of a car when it enters a tunnel and the GNSS signal is blocked, with an IMU the car can still track its position with dead reckoning [61]. A disadvantage of the IMU compared to other positioning techniques is that the positional error is accumulated over time. This is why it is best used together with another positioning technology which can correct the sensor when it is drifting out of position [62]. The IMU should be placed as close to the center of gravity as possible [63].

2.4.5 Global Navigation Satellite Systems

GNSS is a collective name for satellite based navigation and positioning systems [64]. There are several different GNSS systems such as Global Positioning System (GPS), Global Navigation Satellite System (GLONASS) and Galileo [65]. In order to use GNSS systems, free line of sight to the satellites and a GNSS receiver is required [66]. A GNSS system may be used to position a vehicle in a global coordinate system along with a IMU mentioned above. The GNSS antennas should be placed at the top of a vehicle, so it always is in line of sight with the satellites.

2.5 Sensor Placement

Deciding what type of sensors to be used in an autonomous driving project is crucial for gathering the right type of data. However, choosing the right spot to place these sensors is of equal importance as it determines the amount of information that can be used to perceive the surrounding environment. This section presents different placement strategies that are used by autonomous vehicle manufacturers.

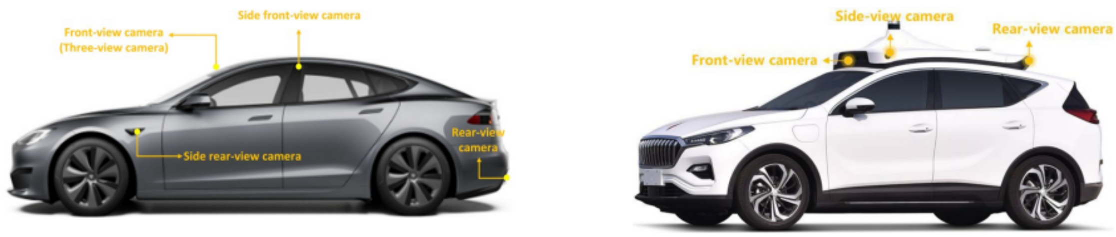
Figure 5: Sensors configuration in a Waymo vehicle [67], edited with [12].

2.5.1 LiDAR Placement

Most manufacturers, including Waymo, choose to equip their autonomous research vehicles with a main LiDAR at the top of the vehicle in order to create a full 360° map of its surroundings, see Figure 5, where the blue dots represent the placement of the LiDAR. However, LiDARs can also be placed on the sides of the vehicle in order to cover the blind spots that the main LiDAR is unable to see.

2.5.2 Camera Placement

According to Wang et al., a perception system needs to be equipped with at least six cameras in order to fulfill the basic requirements for autonomous driving and create a 360° coverage for the car's surrounding [58]. For that purpose, three main types of cameras are being widely used by autonomous vehicles manufacturers. These cameras are front-view, rear-view and side-view cameras. The camera distribution for the Apollo vehicle, shown in Figure 6b, is an example of how these three types of cameras can be placed in an autonomous vehicle. In this vehicle, the minimal number of required cameras is being used, one front-view camera, one rear-view camera and four side-view cameras. On the other hand, Tesla's Model 3 vehicle, seen in Figure 6a, has a different placement of these cameras, where four side-cameras are being used. Two of them are pointed forward and the other two are pointed backwards. In addition, the vehicle is equipped with triple front-view cameras to ensure a full coverage of the surrounding environment [58].



(a) Tesla's Model 3 vehicle.

(b) Apollo vehicle.

Figure 6: Cameras configurations in autonomous vehicles [58]

2.5.3 Radar Placement

A vehicle equipped with a LiDAR and a set of cameras would be able to detect objects and understand the surrounding environment. However, radars are still required to complement these properties by providing instant object detection and accurate velocity measurements. It is therefore important to install radars in positions where their electromagnetic waves does not get interrupted. Furthermore, radars should be placed in positions that allows them to detect obstacles and cover the blind spots that the LiDAR and cameras might miss [68]. In Figure 5, the green dots represent the radars' placement in a Waymo vehicle. By distributing the radars around the vehicle in this fashion, the detection can cover the complete surrounding area, see Figure 7. This is important in order to minimize the presence of blind spots.

Figure 7: Radars' coverage in a Waymo vehicle [68].

3 Related AD-EYE Projects

Under this section other related AD-EYE projects are presented along with a short summary of the AD-EYE software stack.

3.1 Software Stack

A schematic view of the AD-EYE software stack can be seen in Figure 8 along with the PX2[1] in the bottom right corner and the Research Concept Vehicle (RCV)[2] to the bottom left. The software stack can roughly be divided into two parts, the Autonomous Driving Intelligence (ADI) and the simulation environment. The ADI system is based on Autoware [69] together with a safety channel developed by KTH, see the right part of Figure 8. Both Autoware and the safety channel run in a middleware called ROS. The simulation is done in Prescan which is a simulation tool from Siemens [70]. Simulink [71] is used as a interface between ROS [72] and the Prescan environment which is not natively compatible. The simulation environment makes it safer and easier to test new sensor models, vehicle models and driving scenarios.

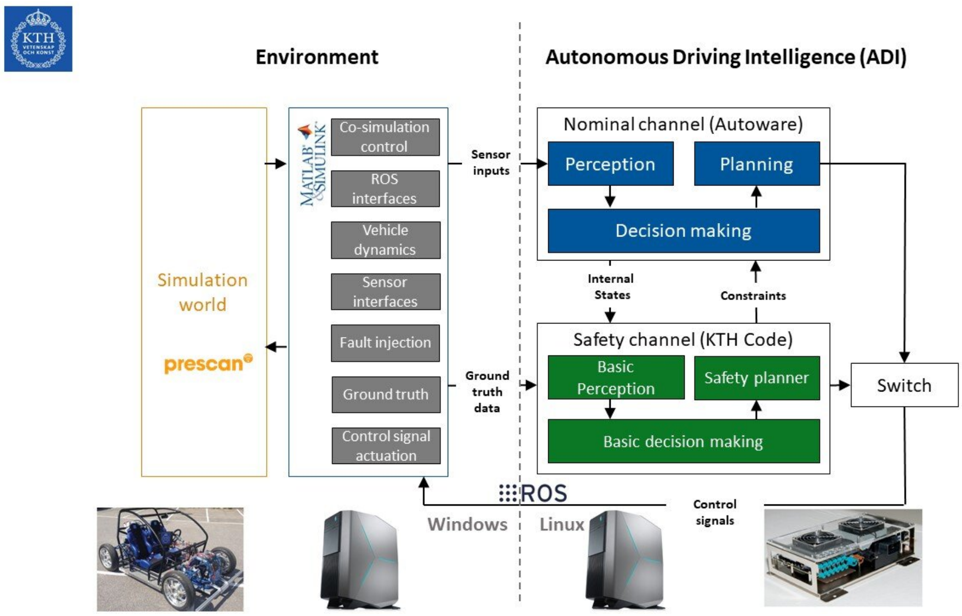


Figure 8: Schematic view of AD-EYE software stack [73]

3.2 Point Cloud Maps Implementation

In order to test the AD-EYE platform in real life applications, highly detailed point cloud maps need to be generated and integrated into the system. This would introduce a real world reference to the platform, which with the use of real time data from sensors, will enable trajectory planning and autonomous driving [74].

During a previous research work held by students at KTH, several point cloud maps were established. This was done by using a LiDAR and an IMU to generate and record the point cloud map into several ROS-bag files which were then processed by the Autoware channel into a point cloud data file. Furthermore, the *VectorMapBuilder* tool was used to generate vector maps that could be transformed to the PX2 to be used during autonomous driving [74].

[1]See section 2.2.2
[2]See section 3.3

3.3 AD-EYE On Research Concept Vehicle

The RCV and RCV-E are research platforms developed by the ITRL lab at KTH. The purpose of these research platforms is to provide vehicles where research results can be tested and verified in the real world. The RCVs are developed in collaboration with students and multiple departments at KTH [75]. The RCVs are equipped with technology which enables them to be controlled via an API or manually [76].

In 2020, students from the the mechatronics capstone project used the RCV-E to test the AD-EYE software in a real environment. The AD-EYE software stack was installed on a PX2 and integrated into the car. The on board hardware and software from the RCVs was interfaced to the AD-EYE software to actuate the vehicle. The sensor setup consisted of a LiDAR, odometers and an IMU. The team achieved autonomous driving for a short distance on campus. However, the AD-EYE software would lose track of the RCV's position after a short while. The authors discussed contributing factors to this particular performance issue. The used solution did not implement GNSS, and consequently, the software struggled to navigate when moving objects, such as trees moving with the wind, were present [74].

3.4 Road Side Unit

A road-side unit is a communication relay, designed to be an access point for a vehicle. The RSU connects with the on board unit of a vehicle, over a DSRC protocol [77]. By doing so, the vehicle and the RSU can share information wirelessly, such as GNSS, automatic toll collection, or help schedule traffic light timing. By incorporating multiple RSUs, the vehicles on the road can effectively communicate with each other, sharing important data, such as location and speed. The RSU is intended to act as a remote agent supplying vehicles with V2X-data, in order to improve traffic management.

The design of the RSUs differs depending on the type of functions they need to fulfill. Regardless of the functions, some key components are to be expected to have mandatory features, such as wireless capabilities, message caching, an antenna to receive GNSS signals and communicate wirelessly, an Application Processing Unit (APU) to run applications, a Controller Unit (CU) to direct the operations and a modem to communicate with a data center [78]. Moreover, the placement of a RSU in the roadside infrastructure is important for establishing effective coverage for the vehicular network.

A RSU was designed to be used with the AD-EYE software by the students at KTH during 2021. This RSU houses a WiFi router, a WiFi antenna, an ITS-G5 router and an antenna, GNSS, a car battery, a motherboard, a GPU and a thermostat. The aim was to relieve the On Board Unit (OBU) computing by using Edge Computing, as the project made use of Raspberry Pi's and Turtlebots. The RSU would then process data from the aforementioned units.Furthermore, two algorithms of note were developed, an object detection algorithm and a platooning algorithm [79].

The project described in this report would employ the RSU developed by the AD-EYE group of 2021, in order to fulfill the stakeholder requirements 1.4, 1.5, 3.1, 3.2, 3.3.

4 Concept Design

In the following section a concept design is proposed, based on the information presented in the SOTA, and the Current Design section.

4.1 Sensors

For the autonomous car to be able to perceive obstacles on the road and to locate itself, different sensors are necessary. To fully define the concept design for the sensors, it is necessary to explore the possible placement of the sensors and the required mounting of them. One LiDAR, two to three cameras, one radar and a GNSS are expected to be used.

4.1.1 Placement

A single 3D-LiDAR could be placed in the middle of the vehicle roof, as is shown in Figure 9. This setup obtains only one point cloud, that does not require alignment with other point clouds of other LiDARs. Moreover, with this placement a 360°coverage would be obtained. The main disadvantage of this placement would be that the roof of the vehicle would create blind areas around it, as can be seen in Figure 9. One way to reduce these blind spots could be to elevate the LiDAR to give it a greater range of view. However, this would increase the need of a more robust mounting system, since the sensor would be more exposed.

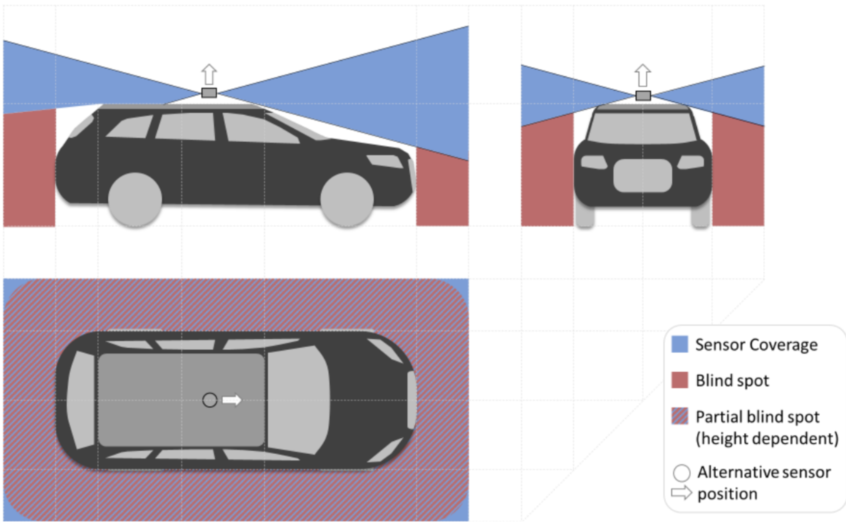


Figure 9: LiDAR placement [80]

If two cameras are provided, the optimal choice would be to place one in the front and one in the rear of the car as can be seen in Figure 10. By placing one camera in the front, detection of traffic lights, lanes in the road and obstacles would be enabled. The main purpose of the placement of the rear camera would be to perform auxiliary reversing for safe parking while detecting static and dynamic obstacles. The drawback of this sensor placement however, would be limited detection of objects that are located on the sides of the vehicle.

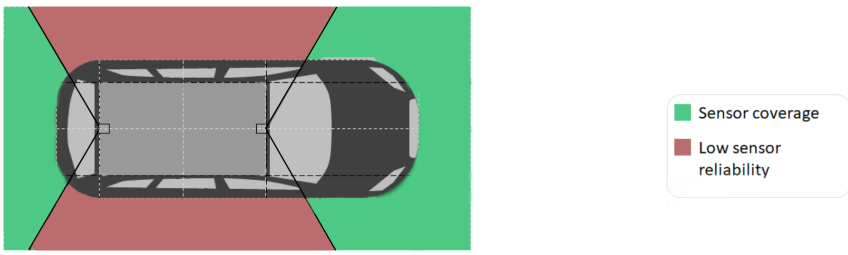


Figure 10: Two cameras placement [80], edited with [12]

If three cameras are provided, one placement possibility would be to have two cameras in the front and one in the back, as is illustrated in Figure 11. This setup would allow the complete view of the front and increment the view of both sides around the vehicle. This placement would result in an increased field of view, that could help detect vehicles and other object around the car and evidently improve the features for parking safely.

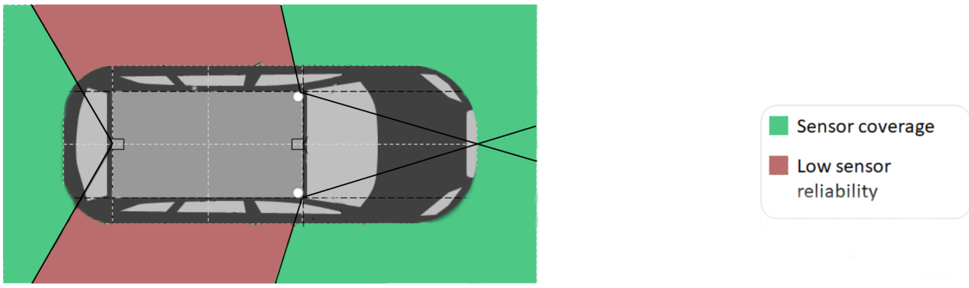


Figure 11: Three cameras placement [80], edited with [12]

If the vertical field of view of the front camera turns out to not be wide enough to correctly detect traffic lights, one of the front cameras could be placed on the side tilted upwards. This placement would help to correctly detect the traffic lights and road signs. In Table 1 a comparison between the different camera placement options can be seen.

	2 cameras	**3 cameras**	
		One on each side	**One in the center, one tilted**
Pros	Simple mounting, simple sensor fusion	More visible parts on the sides	Ability to detect high traffic lights and signals
Cons	Shorter field of view, blind spots in the side, difficulty detecting signs and traffic lights	Complexity in data fusion, difficulty detecting signs and traffic lights	Complexity in data fusion, shorter field of view horizontally

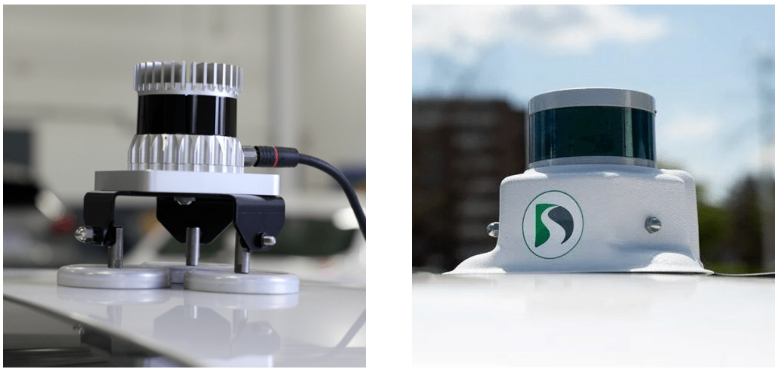
Table 1: Camera placement options comparison

In case that one radar or one ultrasonic sensor would be provided, it could then be placed in the front of the vehicle. As commented in section 2.4.3, radars can be helpful in detecting the surroundings while ultrasonic sensors can be important in poor visibility conditions.

Referring to the localization sensors, if the GNSS has a receiver and its own antennas, the receiver would preferably be placed near the center of gravity of the car and the antennas should be placed in an uncovered place. The IMU can be installed in a shielded container, deep into the vehicle's chassis. By choosing this placement, the IMU is completely protected from weather and other environmental conditions [81].

4.1.2 Mounting

As mentioned in section 4.1.1, the position of the LiDAR could be on the roof of the vehicle. This placement would cause some blind spots as illustrated in Figure 9. In order to reduce the blind spots, the sensor could be elevated by using the mounting gadget shown in Figure 12. That mount consists of three magnetic plates that could be used on slightly curved surfaces.



(a) Without shroud (b) With shroud

Figure 12: Magnetic LiDAR mount [82]

Another option that could be used to mount both LiDAR and cameras is the roof rack of the car with load bars. If the vehicle has a naked roof, a load carrier feet could be used to replace the roof rack. These could be installed to the side of the car, in the front and in the back, connected by load bars that could be used to mount LiDAR and cameras.

A third option for the perception sensors mounting is to use suction cups. Suction cups use negative pressure to adhere to the nonporous surface. It is a cheap solution and easy to install. However, vibrations would produce a lot of noise on the data. For that reason, additional vibration dampers could be used [83].

4.2 Communication

The vehicle has several components that need to communicate with the PX2, which in turn has to communicate with the RSU, see Figure 13. The design should encompass these components so that the line of communication is fast, efficient and constructed with regards to minimizing risks such as delay, package loss or a faulty connection, which in turn could impair time-critical functions.

The sensors could be connected through a combination of the CAN, Ethernet, and USB hardware to the PX2, which is the main CPU. The PX2 could in turn be connected to the RSU through the use of a wireless protocol, e.g., the ITS-G5 or any other wireless protocol that fulfills the performance needs.

The PX2 contains two Tegra cores, which are already connected to each other but do not share operations like other multi-core processors do. Instead, to allow the burden of operations to be relieved, the two cores connection should be harnessed to improve the performance or add redundancy in order to increase safety.

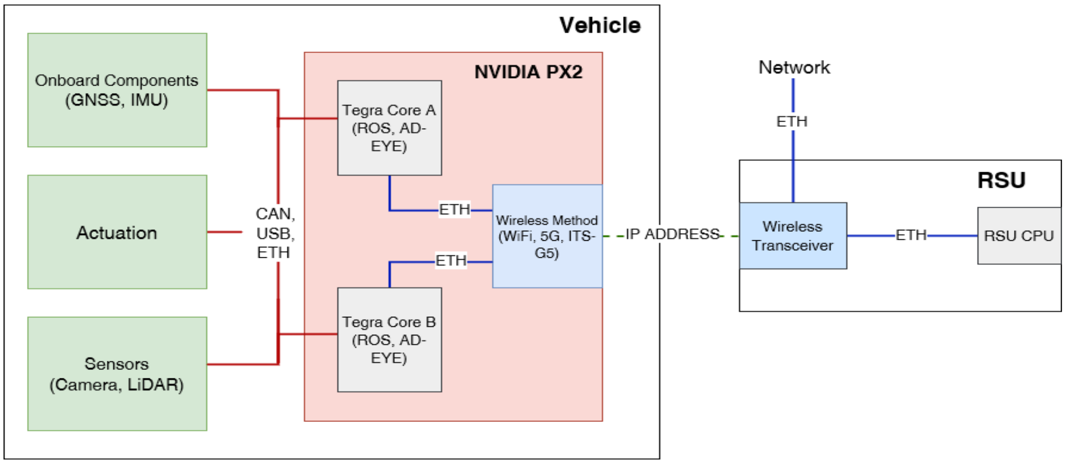


Figure 13: Schematic of the communication interface, created with [84].

4.3 Interfacing

As stated in 2.2.2 On Board Computer, the PX2 includes several relevant interfaces for an autonomous vehicle. The easiest approach would be to choose sensors with these interfaces. However, as per the stakeholder requirements, the choice of sensors will be decided and provided by the stakeholder. If the group is provided sensors that are not equipped with these interfaces, additional adapters and software might be required for the connection.

As stated in the stakeholder requirement 1.1, and further in 1.6, the PX2 shall be connected to the existing CAN bus of the vehicle. This could be done either by connecting to the OBD port of the vehicle, or directly to the wires for the CAN bus. The easiest approach would be to connect to the OBD port since it is accessible from inside the vehicle without any modifications. However, there is no guarantee that the OBD port includes a connection to the CAN bus, and if the connection is there, the information and control available might be limited. This means that to ensure access to the full CAN bus, the PX2 has to be connected directly through wiring to the vehicle's CAN bus. The wires are not easily accessible and hidden behind interior panels, hence an interior panel has to be removed or modified to give access to the vehicle's internal wires. Since the PX2's CAN interfaces use DB9 connectors [44], an adapter is required to connect directly to the wires.

Another required connection is the power connection to the PX2 and the sensors. The PX2 has a 12 V connector for power which could be connected directly to the vehicle's 12 V power system. The installation guide for the PX2 recommends an 80 A fuse in series and a diode in parallel to suppress transient voltages [85]. If the sensors are compatible with 12 V input then they could also be connected to the 12 V system. Otherwise, power regulators are required to step down or up the voltage to a level the sensors can handle. The sensors also requires appropriate fuses in series to ensure safe operations. The PX2 also requires a connection to an antenna or network card to handle the wireless communication with the RSU, since the PX2 only includes wired networking.

5 Organisation And Planning

The entire team consists of eight members in total. In order to establish a clear and efficient structure for the project, the team identified four different topics that are essential to cover all the main aspects of the project. These topics are listed below:

- CAN and Control System
- AD-EYE and ROS Software

- Hardware and Mounting of Sensors
- Communication

On a weekly basis, the group members participated in group meetings and coach meetings where different solutions, suggestions and problems were discussed. Meetings with the stakeholder was frequently scheduled to ensure clear communication and achievement of the project requirements. In the autumn, when moving forward with the project, the team intends to maintain these good habits.

5.1 Planning the Autumn 2022

The project will continue during the autumn of 2022. To ensure the success of the project, certain milestones have been set for the autumn. The list of the major milestones is stated as below:

- Sensor placement on commercial vehicle
- Sensor mounting
- Mount design and manufacturing
- Electrical set up and wiring
- Interfacing sensors to the AD-EYE software
- On board CAN interface setup
- Sensor optimization for real world environment
- Testing in real world environment

For more details regarding the planning of the project during the autumn, a detailed Gantt chart is available in Appendix A.

5.2 Risk Analysis

Analysing the risks that could arise during the project is of high importance since it helps the group members to identify weaknesses, strengths and possible opportunities. To establish a reliable risk analysis, risks should be identified and categorised into four main categories; Technical, External, Organisational and Project management risks. Once the risks are identified, they are examined in order to determine their probability of occurrence and impact on the project. This can be used to calculate the risk exposure, which is a representation of how urgent a risk response should be implemented if the risk occurs [86].

For this project, a risk analysis, shown in Appendix B, is implemented to determine the possible risks and introduce the appropriate strategies that can be followed to minimize the negative impact on the project.

6 Discussion

6.1 Sensor Placement

During the time devoted for this project, different designs for the sensor placement have been taken into consideration. However, the design and choice of the sensors, as well as their placement on the car, is limited by the directives set by the stakeholder. At present, it is stated that one LiDAR, two or three cameras, a radar and an IMU will be provided. Nevertheless, the exact quantity of the cameras is yet not known and this information might change depending on the directives from the stakeholder. It has also been indicated by the stakeholder that it could be a good idea to mimic the placement of the sensors as they are represented in the simulation environment Prescan.

6.2 Design Freedom

Due to the fact that the stakeholder is providing both the car and the sensors to be used in this project, the final design will have to be adapted to the amount and the type of sensors that will be provided. Furthermore, the lack of information regarding the type of the car and sensor models makes it challenging to finalise the mounting strategy at present. The challenges in identifying the mounting strategy at this point of the project is not caused by the availability of the mounting parts, as they are mainly cheap and easy to construct. The real limiting factor is the possible modifications that can be done on the car. As the car is the most expensive part of the project, all modifications must be carried out with care, precision and, of course, with the permission of the stakeholder. Another mounting constraint that should be considered is the location of the on board computer, which should be located close to the vehicle's control system and at the same time accessible to the user.

From a technical perspective, the design will also be constrained by the limited accessibility to the car communication system. The CAN communication protocol of the car is protected by the manufacturer and the team will have limited access to some parts of the system. This will limit the amount of information and data that can be sent to and received from the car, which might affect the implementation of the AD-EYE system in the car.

6.3 Knowledge Gaps

The group members possess a various set of skills and experiences including hardware design, software design and system engineering. However, usage of the CAN protocol and wireless communication is not something that is familiar to the team. This will add some challenges to the project, but help and support may be sought from the teaching team. Furthermore, a significant part of the devices and technologies that are associated with the AD-EYE system originates from projects conducted by former students, such as the RSU and the Point Cloud Map. This might add difficulties when troubleshooting and integrating these parts into the project. However, this has already been discussed with the teaching team and the stakeholder. If necessary, the group has access to the contact information of the former students that have been involved in the development of the RSU.

6.4 Feasibility

The most challenging part of the project is to accomplish the stakeholder and technical requirements on time. Most of the necessary components are provided by the stakeholder which could pose a problem when assessing the planning and scheduling of the project for the autumn. Nevertheless, the stakeholder has stated that the components will be available by the start of the next semester, and based on this information, a plan for the autumn semester has been set. Moreover, the lack of experience regarding the CAN interface and wireless communication might affect the feasibility of the project. Therefore, a large part of the allocated time during the autumn has to be devoted to the interface part and vehicle controlling.

7 Bibliography

References

[1] *The mechatronics hk 2022 project descriptions*, 2022-03-18.

[2] C. Engelking. "The 'driverless' car era began more than 90 years." (2022), [Online]. Available: https://www.discovermagazine.com/technology/the-driverless-car-era-began-more-than-90-years-ago (visited on 2022-05-21).

[3] J. Schmidhuber. "Prof. schmidhuber's highlights of robot car history." (2022), [Online]. Available: https://people.idsia.ch/~juergen/robotcars.html (visited on 2022-05-21).

[4] Y. K. Ryosuke Okuda and K. Terashima, "A survey of technical trend of adas and autonomous driving," *Technical Papers of 2014 International Symposium on VLSI Design, Automation and Test*, 2014. DOI: 10.1109/VLSI-DAT.2014.6834940. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/6834940/authors#authors.

[5] K. Kirkpatrick, "Still waiting for self-driving cars; why is it taking so long for autonomous vehicles to hit the road?" *Communications of the ACM*, vol. 65, 4 2022. DOI: DOI:10.1145/3516517. [Online]. Available: https://dl.acm.org/doi/pdf/10.1145/3516517?casa_token=pOIf-nyWXFsAAAAA:SQeX76PGQIPT__Sr0zw0KemV870JmzO6FH6fL57_nVdbh9pldFmga3gOrm11xUrQULrEfTFOGsg5g.

[6] Z. T. Prabhjot Kaur Samira Taghavi and W. Shi, "A survey on simulators for testing self-driving cars," *Fourth International Conference on Connected and Autonomous Driving (MetroCAD)*, 2021. DOI: 10.1109/MetroCAD51599.2021.00018. [Online]. Available: https://dl.acm.org/doi/pdf/10.1145/3516517?casa_token=pOIf-nyWXFsAAAAA:SQeX76PGQIPT__Sr0zw0K2emV870JmzO6FH6fL57_nVdbh9pldFmga3gOrm11xUrQULrEfTFOGsg5g.

[7] B. Ahlborn. "Five reasons why we benefit from v2x." (2016-05-26), [Online]. Available: https://www.nxp.com/company/blog/five-reasons-why-we-benefit-from-v2x:BL-V2X-REASONS (visited on 2022-05-21).

[8] "Transforming our world: The 2030 agenda for sustainable development," United Nations General Assembly, 2015. [Online]. Available: https://sustainabledevelopment.un.org/content/documents/21252030%20Agenda%20for%20Sustainable%20Development%20web.pdf (visited on 2022-05-02).

[9] Coalition for future mobility. "Highly automated technologies, often called self-driving cars, promise a range of potential benefits." (2018-03-19), [Online]. Available: https://coalitionforfuturemobility.com/benefits-of-self-driving-vehicles/ (visited on 2022-05-02).

[10] M. Maurer, J. Gerdes, B. Lenz, and H. Winner, *Autonomous Driving, Technical, Legal and Social Aspects*. Springer Nature, 2015, pp. 69–82, ISBN: 978-3-662-48845-4.

[11] S. of Automotive Engineers. "Sae levels of driving automation™ refined for clarity and international audience." (2021-05-03), [Online]. Available: https://www.sae.org/blog/sae-j3016-update (visited on 2022-05-22).

[12] J. K. Thomas Knoll, *Photoshop*, version 22.1.0, 2021. [Online]. Available: https://www.adobe.com/es/products/photoshop.html.

[13] J. Cusack. "Self-driving vehicles are steadily becoming a reality despite the many hurdles still to be overcome – and they could change our world in some unexpected ways." (2021-10-30), [Online]. Available: https://www.bbc.com/future/article/20211126-how-driverless-cars-will-change-our-world (visited on 2022-05-23).

[14] F. M. Company. "Volvo pilot assist." (2022), [Online]. Available: https://www.volvocars.com/se/support/topics/anvanda-din-bil/bra-att-veta/pilot-assist-och-korfaltsassistans (visited on 2022-05-06).

[15] B. AG. "Overview of driver assistance systems." (2021-09-16), [Online]. Available: https://www.bmw.com/en/innovation/the-main-driver-assistance-systems.html (visited on 2022-05-12).

[16] "Ford co-pilot360tm technology." (), [Online]. Available: https://www.ford.com/technology/driver-assist-technology/.

[17] Tesla. "Tesla autopilot." (2022), [Online]. Available: https://www.tesla.com/sv_SE/autopilot (visited on 2022-05-19).

[18] D. Coldewey. "Tesla vaunts creation of 'the best chip in the world' for self-driving." (2019-04-22), [Online]. Available: https://techcrunch.com/2019/04/22/tesla-vaunts-creation-of-the-best-chip-in-the-world-for-self-driving/ (visited on 2022-05-18).

[19] M. McFarland. "Tesla owners say they are wowed – and alarmed – by 'full self-driving'." (2022-03-16), [Online]. Available: https://edition.cnn.com/2021/11/03/cars/tesla-full-self-driving-fsd/index.html (visited on 2022-05-17).

[20] T. Reid. "Example of ford research vehicle." (2019-09), [Online]. Available: https://www.researchgate.net/figure/Example-of-Fords-research-autonomous-vehicle-platform-sensors-used-in-localization-and_fig5_336515529 (visited on 2022-05-15).

[21] R. Hussain and S. Zeadally, "Autonomous cars: Research results, issues, and future challenges," *IEEE Communications Surveys Tutorials*, vol. 21, no. 2, pp. 1275–1313, 2019. DOI: 10.1109/COMST.2018.2869360.

[22] D. Ghosh. "What are robotaxis?" (2021-12-17), [Online]. Available: https://htschool.hindustantimes.com/editorsdesk/news-trends/robotaxis-are-coming-up-but-what-are-they/ (visited on 2022-05-21).

[23] M. Herger. "Autox opens driverless robotaxis for the public." (2021-01-27), [Online]. Available: https://thelastdriverlicenseholder.com/2021/01/27/autox-opens-driverless-robotaxis-for-the-public/ (visited on 2022-05-23).

[24] W. LLC. "Waymo driver." (2022), [Online]. Available: https://waymo.com/waymo-driver/ (visited on 2022-05-10).

[25] R. Bellan. "Waymo opens driverless robotaxi service to san franscisco employees." (2022-03-20), [Online]. Available: https://techcrunch.com/2022/03/30/waymo-opens-driverless-robotaxi-service-to-san-francisco-employees/ (visited on 2022-05-05).

[26] D. Michael, M. Kleanthous, M. Savva, S. Christodoulou, M. Pampaka, and A. Gregoriades, "Impact of immersion and realism in driving simulator studies," *Int. J. Interdiscip. Telecommun. Netw.*, vol. 6, no. 1, pp. 10–25, 2014-01, ISSN: 1941-8663.

[27] AstaZero. "Test site." (2022), [Online]. Available: https://www.astazero.com/en/tracks-facilities/test-site/ (visited on 2022-05-21).

[28] Millbrook. "Test bed for connected and autonomous vehicles — cav — millbrook." (2022), [Online]. Available: https://www.millbrook.co.uk/services/connected-and-autonomous-vehicle-testing/ (visited on 2022-05-21).

[29] G. H. Ruffo. "Musk says giga berlin will be testbed for new tesla technologies." (2020-10-07), [Online]. Available: https://insideevs.com/news/447771/musk-giga-belin-testbed-new-tesla-technologies/ (visited on 2022-05-07).

[30] P. Biswas. "Iit-h sets up testbed on autonomous navigation (tihan) for unmanned aerial vehicles." (2020-12-29), [Online]. Available: https://timesofindia.indiatimes.com/home/education/news/iit-h-sets-up-testbed-on-autonomous-navigation-tihan-for-unmanned-aerial-vehicles/articleshow/80011899.cms (visited on 2022-05-02).

[31] J. He-Rim. "Hyundai motor to launch self-driving tech test bed." (2021-10-12), [Online]. Available: http://www.koreaherald.com/view.php?ud=20211012000799 (visited on 2022-05-09).

[32] A. Salter. "7 universities that are pushing the boundaries of autonomous driving." (2021-10-18), [Online]. Available: https://www.2025ad.com/7-universities-that-are-pushing-the-boundaries-of-autonomous-driving-2021 (visited on 2022-05-16).

[33] C. in Automation - CiA. "Classical controller area network (can)." (2022), [Online]. Available: https://www.can-cia.org/can-knowledge/can/classical-can/ (visited on 2022-04-10).

[34] C. in Automation - CiA. "Can: From physical layer to application layer and beyond." (2022), [Online]. Available: https://www.can-cia.org/can-knowledge/ (visited on 2022-04-10).

[35] M. S. U. Alam, "Securing vehicle electronic control unit (ecu) communications and stored data," pp. 9–10, 2019.

[36] C. Electronics. "Can bus explained - a simple intro [2021]." (2021-11-01), [Online]. Available: https://www.csselectronics.com/pages/can-bus-simple-intro-tutorial (visited on 2022-04-10).

[37] K. H. Johansson, M. Törngren, and L. Nielsen, "Vehicle applications of controller area network," *Springer*, vol. 6, no. 6, pp. 741–765, 2005. DOI: 10.1007/0-8176-4404-0_32.

[38] C. Electronics. "Obd2 explained - a simple intro [2021]." (2021-11-01), [Online]. Available: https://www.csselectronics.com/pages/obd2-explained-simple-intro (visited on 2022-04-10).

[39] R. De Andrade, K. Hodel, J. Fransisco Justo, A. Laganá, and M. Mauro Santos, "Analytical and experimental performance evaluations of can-fd bus," *IEEE Access*, vol. 6, no. 3, pp. 21 287–21 289, 2018. DOI: 10.1109/ACCESS.2018.2826522.

[40] C. in Automation - CiA. "Can fd - the basic idea." (2022), [Online]. Available: https://www.can-cia.org/can-knowledge/can/can-fd/ (visited on 2022-04-04).

[41] N. Corporation. "Nvidia drive sdk." (2022), [Online]. Available: https://developer.nvidia.com/drive/drive-sdk/ (visited on 2022-04-09).

[42] N. Corporation. "Nvidia drive px, Scalable ai supercomputer for autonomous driving." (2022), [Online]. Available: https://www.nvidia.com/content/nvidiaGDC/sg/en_SG/self-driving-cars/drive-px/ (visited on 2022-04-08).

[43] R. Smith. "Nvidia announces drive px 2 - pascal power for self-driving cars." (2016-01-05), [Online]. Available: https://www.anandtech.com/show/9903/nvidia-announces-drive-px-2-pascal-power-for-selfdriving-cars/ (visited on 2022-04-08).

[44] N. Automotive, *Drive px 2 hands-on, Drive px 2 sdk 4.1.8.0*, Rev. 20170725, 2017-07-25.

[45] N. Corporation. "Introducing the new nvidia drive px 2, For autocruise driving and hd mapping." (2016-10-13), [Online]. Available: http://web.archive.org/web/20161013023854/http://www.nvidia.com/object/drive-px.html/ (visited on 2022-04-09).

[46] N. Corporation. "Solutions for self-driving cars & autonomous vehicles." (2022), [Online]. Available: https://www.nvidia.com/en-us/self-driving-cars/ (visited on 2022-04-08).

[47] N. Corporation. "Software for self-driving cars." (2022), [Online]. Available: https://www.nvidia.com/en-us/self-driving-cars/drive-platform/software/ (visited on 2022-04-08).

[48] H. Ullah, N. G. Nair, A. Moore, C. Nugent, P. Muschamp, and M. Cuevas, "5g communication: An overview of vehicle-to-everything, drones, and healthcare use-cases," *IEEE Access*, vol. 7, pp. 37 251–37 268, 2019.

[49] A. I. Al-Alawi, "Wifi technology: Future market challenges and opportunities," *Journal of computer science*, vol. 2, no. 1, pp. 13–18, 2006.

[50] E. J. Oughton, W. Lehr, K. Katsaros, I. Selinis, D. Bubley, and J. Kusuma, "Revisiting wireless internet connectivity: 5g vs wi-fi 6," *Telecommunications Policy*, vol. 45, no. 5, p. 102 127, 2021, ISSN: 0308-5961. DOI: https://doi.org/10.1016/j.telpol.2021.102127. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S030859612100032X.

[51] I. Corporation. "What is wi-fi?" (), [Online]. Available: https://www.intel.com/content/www/us/en/gaming/resources/wifi-6.html (visited on 2022-04-07).

[52] E. Notes. "What is cellular communications: Mobile technology?" (), [Online]. Available: https://www.electronics-notes.com/articles/connectivity/cellular-mobile-phone/what-is-cellular-communications.php (visited on 2022-04-07).

[53] R.-s. Ltd. "What's the difference between high-band, mid-band and low-band 5g?" (), [Online]. Available: https://www.router-switch.com/faq/what-s-the-difference-between-high-band-mid-band-and-low-band-5g.html (visited on 2022-04-07).

[54] A. K. Gulia, *A simulation study on the performance comparison of the v2x communication systems: Its-g5 and c-v2x*, 2020. [Online]. Available: http://www.diva-portal.org/smash/record.jsf?pid=diva2%3A1422828&dswid=-2617 (visited on 2022-05-21).

[55] RENISHAW. "Optical encoders and lidar scanning." (), [Online]. Available: https://www.renishaw.fr/fr/optical-encoders-and-lidar-scanning--39244 (visited on 2022-05-02).

[56] "Point clouds for beginners." (2022), [Online]. Available: https://geoslam.com/point-clouds/ (visited on 2022-05-03).

[57] A. Dubey, "Stereo vision—facing the challenges and seeing the opportunities for adas applications," Texas Instruments, 2020. [Online]. Available: https://www.ti.com/lit/wp/spry300a/spry300a.pdf?ts=1650270194572&ref_url=https%253A%252F%252Fwww.google.com%252F#:~:text=If%20we%20have%20a%20single,called%20a%20stereo%20vision%20system. (visited on 2021-05-01).

[58] C. Wang, X. Wang, H. Hu, Y. Liang, and G. Shen1, "On the application of cameras used in autonomous vehicles," 2022-04-12. [Online]. Available: https://link.springer.com/article/10.1007/s11831-022-09741-8 (visited on 2022-05-21).

[59] M. M. Vázquez. "Radar transceivers: A key component for adas & autonomous driving." (2021), [Online]. Available: https://www.renesas.com/us/en/blogs/radar-transceivers-key-component-adas-autonomous-driving-blog-1-why-do-we-need-radar (visited on 2022-05-21).

[60] J. J. Asiag. "How ultrasonic sensor data is powering automotive iot." (2021-04-17), [Online]. Available: https://otonomo.io/blog/ultrasonic-data-automotive-iot/#:~:text=Ultrasonic%20sensors%20mimic%20echolocation%20used,immediate%%20of%20a%20vehicle. (visited on 2022-05-21).

[61] X. Lu, X. Xin, L. Yishi, *et al.*, "Imu-based automated vehicle slip angle and attitude estimation aided by vehicle dynamics," *Sensors*, vol. 19, no. 8, 2019, ISSN: 1424-8220. DOI: 10.3390/s19081930. [Online]. Available: https://www.mdpi.com/1424-8220/19/8/1930.

[62] S. Sukkarieh, E. Nebot, and H. Durrant-Whyte, "A high integrity imu/gps navigation loop for autonomous land vehicle applications," *IEEE Transactions on Robotics and Automation*, vol. 15, no. 3, pp. 572–578, 1999. DOI: 10.1109/70.768189.

[63] A. D. Team. "Sensor position offset compensation." (2021), [Online]. Available: https://ardupilot.org/plane/docs/common-sensor-offset-compensation.html (visited on 2022-04-04).

[64] Lantmäteriet. "Gps och andra gnss." (2022), [Online]. Available: https://www.lantmateriet.se/sv/Kartor-och-geografisk-information/gps-geodesi-och-swepos/GPS-och-satellitpositionering/GPS-och-andra-GNSS/ (visited on 2022-04-04).

[65] EUSPA. "What is gnss?" (2021-12-04), [Online]. Available: https://www.euspa.europa.eu/european-space/eu-space-programme/what-gnss (visited on 2022-04-04).

[66] Lantmäteriet. "Gps och satellitpositionering." (2022), [Online]. Available: https://www.lantmateriet.se/sv/Kartor-och-geografisk-information/gps-geodesi-och-swepos/GPS-och-satellitpositionering/ (visited on 2022-04-04).

[67] T. Lombardo. "Waymo rolls out driverless taxis in phoenix." (2020-10-27), [Online]. Available: https://www.engineering.com/story/waymo-rolls-out-driverless-taxis-in-phoenix (visited on 2022-05-20).

[68] S. Jeyachandran. "Introducing the 5th-generation waymo driver: Informed by experience, designed for scale, engineered to tackle more environments." (2020-03-04), [Online]. Available: https://blog.waymo.com/2020/03/introducing-5th-generation-waymo-driver.html (visited on 2022-05-20).

[69] "Autoware." (2021), [Online]. Available: https://www.autoware.org (visited on 2022-04-25).

[70] "Simcenter prescan." (2022), [Online]. Available: https://www.plm.automation.siemens.com/global/en/products/simcenter/prescan.html (visited on 2022-04-25).

[71] "Simulink." (2022), [Online]. Available: https://se.mathworks.com/products/simulink.html (visited on 2022-04-25).

[72] "Why ros?" (2021), [Online]. Available: https://www.ros.org/blog/why-ros/ (visited on 2022-04-25).

[73] N. Mohan. "Ad-eye." (2022), [Online]. Available: https://www.adeye.se/home (visited on 2022-05-02).

[74] V. Hanefors, C. Jensen, J. Jungåker, *et al.*, "Ad-eye, Implementation of an autonomous driving platform," Kungliga Tekniska Högskolan, 2020. [Online]. Available: https://www.kth.se/social/files/605e01487fb52d433333764d/ad-eye-final-report.pdf (visited on 2022-04-29).

[75] "Kth research concept vehicle." (2021-01-07), [Online]. Available: https://www.itrl.kth.se/about-us/labs/rcv-1.476469 (visited on 2022-04-30).

[76] "Research concept vehicle (rcv) – research platform for future vehicle concepts." (2014-06-10), [Online]. Available: https://www.itrl.kth.se/polopoly_fs/1.487591.1600688751!/RCV_Mikael_Nybacka_v3_red.pdf (visited on 2022-04-30).

[77] S. Ackels, P. Benavidez, and M. Jamshidi, "A survey of modern roadside unit deployment research," eng, in *World Automation Congress Proceedings*, vol. 2021-, 2021, pp. 137–141, ISBN: 1685241115.

[78] U. D. of Transportation. "Standards training." (2021), [Online]. Available: https://www.pcb.its.dot.gov/StandardsTraining/mod57/ppt/m57ppt.pdf (visited on 2022-05-21).

[79] M. Akkila, E. Haltorp, N. Määttä, *et al.*, "Ad-eye, Automated driving & its testbed," Kungliga Tekniska Högskolan, 2021. [Online]. Available: https://www.kth.se/social/files/62430685fa6b7fbe26328fb2/ad-eye-finalreport.pdf (visited on 2021-05-01).

[80] Felix, "Sensor set design patterns for autonomous vehicles," 2019-01-25. [Online]. Available: https://autonomous-driving.org/2019/01/25/positioning-sensors-for-autonomous-vehicles/ (visited on 2022-05-21).

[81] "Inertial measurement units for localisation and navigation of autonomous vehicles," 2021-07-21. [Online]. Available: https://sensible4.fi/technology/articles/imu-and-autonomous-vehicles/ (visited on 2022-05-21).

[82] L. F. S. Ltd. "Magnetic lidar mount." (2022), [Online]. Available: https://levelfivesupplies.com/product/magnetic-lidar-mount/ (visited on 2022-05-21).

[83] V. Hanefors, C. Jensen, J. Jungåker, *et al.*, "Ad-eye, State of the art report," Kungliga Tekniska Högskolan, 2020. [Online]. Available: https://www.kth.se/social/files/605cbdd0206fc2b8f16daa2f/ad-eye-springterm-report.pdf (visited on 2022-05-16).

[84] J. Ltd, *Diagrams.net/draw.io*, version 18.1.1. [Online]. Available: https://app.diagrams.net.

[85] N. Corporation. "Nvidia drive px 2 autochauffeur mechanical and installation guide." (2018-04-27), [Online]. Available: https://developer.nvidia.com/driveworks/files/drive-px2-mechanical-installation (visited on 2022-05-16).

[86] N. Lavanya and T. Malarvizhi. "Risk analysis and management." (2008-03-03), [Online]. Available: https://www.pmi.org/learning/library/risk-analysis-project-management-7070 (visited on 2022-05-06).

A Gantt Schedules

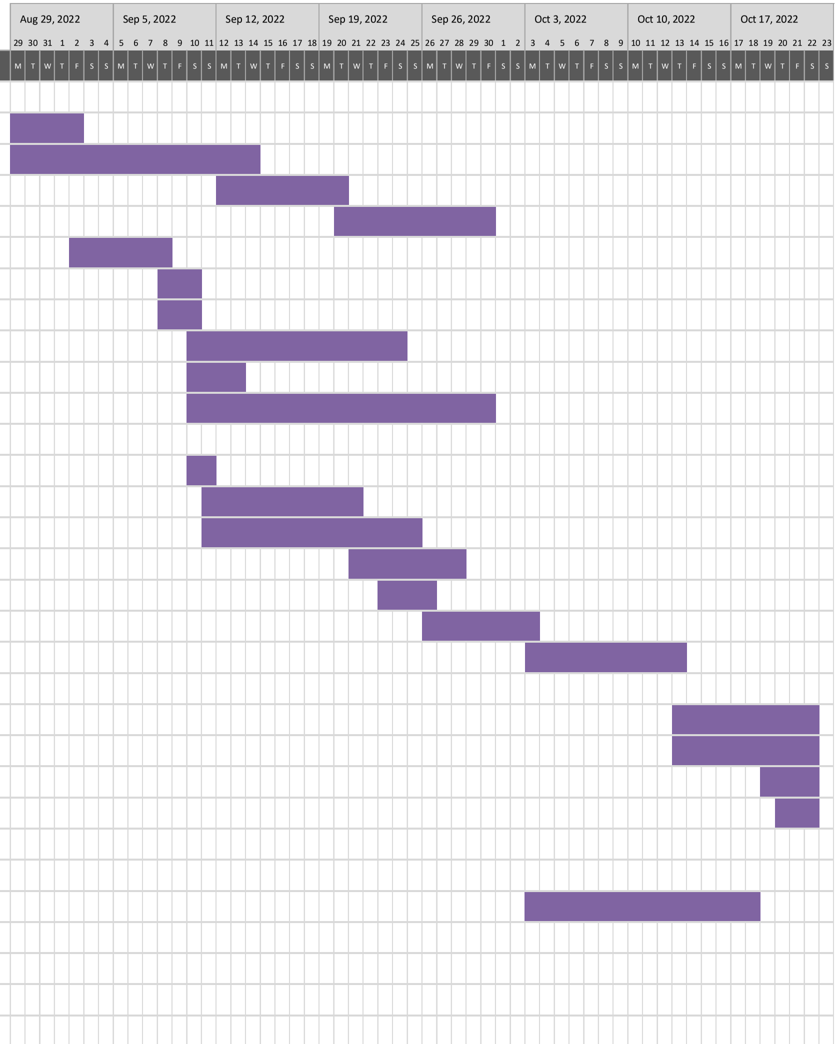
AD-EYE: Towards Autonomous Driving on KTH Campus

KTH

Project Start: Aug 29, 2022

Display Week: 1

TASK	ASSIGNED TO	TIME	PROGRESS	START	END
Phase 1: Set up sensors and sensor mounting					
Obtain all sensors and the commercial vehicle			0%	2022-08-29	2022-09-02
Elecrical set up and interface			0%	2022-08-29	2022-09-14
Connect the sensors to the NVIDIA PX2			0%	2022-09-12	2022-09-20
Ensure data transfer from sensors to NVIDIA PX2			0%	2022-09-20	2022-09-30
Design a container for the NVIDIA PX2 inside the vehicle			0%	2022-09-02	2022-09-08
Establish a safe placement for the container inside the vehicle			0%	2022-09-08	2022-09-10
Establish a safe placement for the GUI set up inside the vehicle			0%	2022-09-08	2022-09-10
Design/obtain armature for mounting			0%	2022-09-10	2022-09-24
Determine final sensor placement			0%	2022-09-10	2022-09-13
Mount sensors on vehicle			0%	2022-09-10	2022-09-30
Phase 2: Interfacing and optimization					
Install ADE-EYE on NVIDIA PX2			0%	2022-09-10	2022-09-11
Interfacing the sensors to the AD-EYE software			0%	2022-09-11	2022-09-21
Interface CAN to the vehicle			0%	2022-09-11	2022-09-25
Interface AD-EYE towards simulated environment			0%	2022-09-21	2022-09-28
Visualize the vehicle state through the GUI			0%	2022-09-23	2022-09-26
Optimize sensor placement			0%	2022-09-26	2022-10-03
Tune sensors for real world environment			0%	2022-10-03	2022-10-13
Phase 3: Control and V2X communication					
Verify vehicle controls			0%	2022-10-13	2022-10-23
Control the vehicle through the GUI by cabled connection			0%	2022-10-13	2022-10-23
Establish connection between the RSU and NVIDIA PX2 via cable			0%	2022-10-19	2022-10-29
Establish connection between the RSU and NVIDIA PX2 via WLAN			0%	2022-10-20	2022-10-30
Establish connection between the RSU and NVIDIA PX2 via 5G			0%	2022-10-30	2022-11-14
Phase 4: Testing					
Ensure that localization and driving map are available			0%	2022-10-03	2022-10-18
Vehicle follows a short predefined trajectory			0%	2022-10-29	2022-11-18
Vehicle drives autonomously to a given goal			0%	2022-10-29	2022-11-23
Safety feature: Vehicle should stop when receiving emergency signal			0%	2022-11-23	2022-12-13



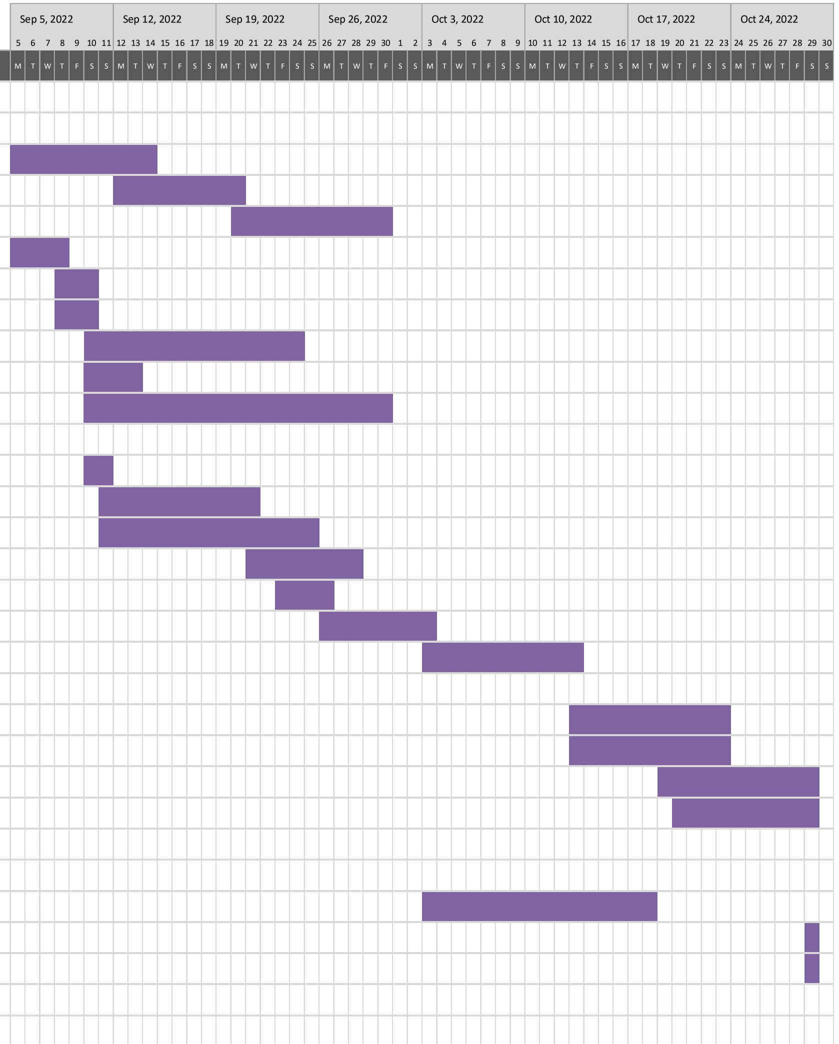
AD-EYE: Towards Autonomous Driving on KTH Campus

KTH

Project Start: Aug 29, 2022

Display Week: 2

TASK	ASSIGNED TO	TIME	PROGRESS	START	END
Phase 1: Set up sensors and sensor mounting					
Obtain all sensors and the commercial vehicle			0%	2022-08-29	2022-09-02
Elecrical set up and interface			0%	2022-08-29	2022-09-14
Connect the sensors to the NVIDIA PX2			0%	2022-09-12	2022-09-20
Ensure data transfer from sensors to NVIDIA PX2			0%	2022-09-20	2022-09-30
Design a container for the NVIDIA PX2 inside the vehicle			0%	2022-09-02	2022-09-08
Establish a safe placement for the container inside the vehicle			0%	2022-09-08	2022-09-10
Establish a safe placement for the GUI set up inside the vehicle			0%	2022-09-08	2022-09-10
Design/obtain armature for mounting			0%	2022-09-10	2022-09-24
Determine final sensor placement			0%	2022-09-10	2022-09-13
Mount sensors on vehicle			0%	2022-09-10	2022-09-30
Phase 2: Interfacing and optimization					
Install ADE-EYE on NVIDIA PX2			0%	2022-09-10	2022-09-11
Interfacing the sensors to the AD-EYE software			0%	2022-09-11	2022-09-21
Interface CAN to the vehicle			0%	2022-09-11	2022-09-25
Interface AD-EYE towards simulated environment			0%	2022-09-21	2022-09-28
Visualize the vehicle state through the GUI			0%	2022-09-23	2022-09-26
Optimize sensor placement			0%	2022-09-26	2022-10-03
Tune sensors for real world environment			0%	2022-10-03	2022-10-13
Phase 3: Control and V2X communication					
Verify vehicle controls			0%	2022-10-13	2022-10-23
Control the vehicle through the GUI by cabled connection			0%	2022-10-13	2022-10-23
Establish connection between the RSU and NVIDIA PX2 via cable			0%	2022-10-19	2022-10-29
Establish connection between the RSU and NVIDIA PX2 via WLAN			0%	2022-10-20	2022-10-30
Establish connection between the RSU and NVIDIA PX2 via 5G			0%	2022-10-30	2022-11-14
Phase 4: Testing					
Ensure that localization and driving map are available			0%	2022-10-03	2022-10-18
Vehicle follows a short predefined trajectory			0%	2022-10-29	2022-11-18
Vehicle drives autonomously to a given goal			0%	2022-10-29	2022-11-23
Safety feature: Vehicle should stop when receiving emergency signal			0%	2022-11-23	2022-12-13



AD-EYE: Towards Autonomous Driving on KTH Campus

KTH

Project Start: Aug 29, 2022

Display Week: 3

Sep 12, 2022 | Sep 19, 2022 | Sep 26, 2022 | Oct 3, 2022 | Oct 10, 2022 | Oct 17, 2022 | Oct 24, 2022 | Oct 31, 2022

12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 1 2 3 4 5 6

M T W T F S S M T W T F S S M T W T F S S M T W T F S S M T W T F S S M T W T F S S M T W T F S S M T W T F S S

TASK	ASSIGNED TO	TIME	PROGRESS	START	END
Phase 1: Set up sensors and sensor mounting					
Obtain all sensors and the commercial vehicle			0%	2022-08-29	2022-09-02
Elecrical set up and interface			0%	2022-08-29	2022-09-14
Connect the sensors to the NVIDIA PX2			0%	2022-09-12	2022-09-20
Ensure data transfer from sensors to NVIDIA PX2			0%	2022-09-20	2022-09-30
Design a container for the NVIDIA PX2 inside the vehicle			0%	2022-09-02	2022-09-08
Establish a safe placement for the container inside the vehicle			0%	2022-09-08	2022-09-10
Establish a safe placement for the GUI set up inside the vehicle			0%	2022-09-08	2022-09-10
Design/obtain armature for mounting			0%	2022-09-10	2022-09-24
Determine final sensor placement			0%	2022-09-10	2022-09-13
Mount sensors on vehicle			0%	2022-09-10	2022-09-30
Phase 2: Interfacing and optimization					
Install ADE-EYE on NVIDIA PX2			0%	2022-09-10	2022-09-11
Interfacing the sensors to the AD-EYE software			0%	2022-09-11	2022-09-21
Interface CAN to the vehicle			0%	2022-09-11	2022-09-25
Interface AD-EYE towards simulated environment			0%	2022-09-21	2022-09-28
Visualize the vehicle state through the GUI			0%	2022-09-23	2022-09-26
Optimize sensor placement			0%	2022-09-26	2022-10-03
Tune sensors for real world environment			0%	2022-10-03	2022-10-13
Phase 3: Control and V2X communication					
Verify vehicle controls			0%	2022-10-13	2022-10-23
Control the vehicle through the GUI by cabled connection			0%	2022-10-13	2022-10-23
Establish connection between the RSU and NVIDIA PX2 via cable			0%	2022-10-19	2022-10-29
Establish connection between the RSU and NVIDIA PX2 via WLAN			0%	2022-10-20	2022-10-30
Establish connection between the RSU and NVIDIA PX2 via 5G			0%	2022-10-30	2022-11-14
Phase 4: Testing					
Ensure that localization and driving map are available			0%	2022-10-03	2022-10-18
Vehicle follows a short predefined trajectory			0%	2022-10-29	2022-11-18
Vehicle drives autonomously to a given goal			0%	2022-10-29	2022-11-23
Safety feature: Vehicle should stop when receiving emergency signal			0%	2022-11-23	2022-12-13

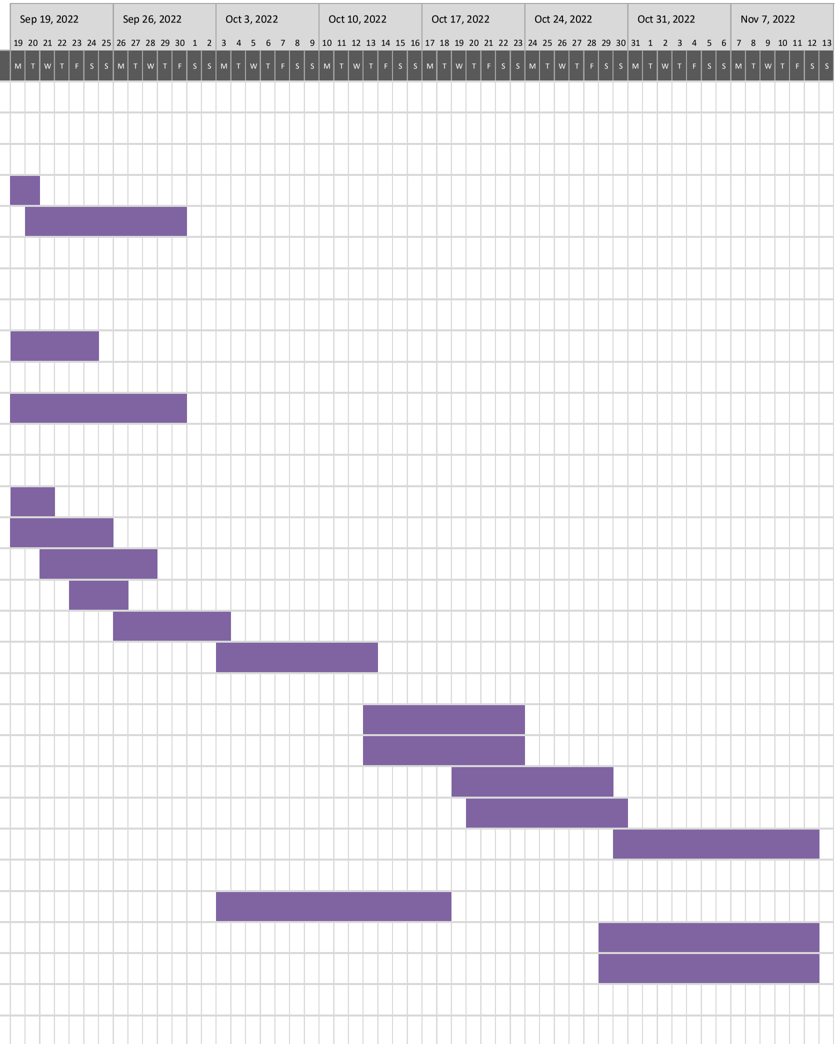
AD-EYE: Towards Autonomous Driving on KTH Campus

KTH

Project Start: Aug 29, 2022

Display Week: 4

TASK	ASSIGNED TO	TIME	PROGRESS	START	END
Phase 1: Set up sensors and sensor mounting					
Obtain all sensors and the commercial vehicle			0%	2022-08-29	2022-09-02
Elecrical set up and interface			0%	2022-08-29	2022-09-14
Connect the sensors to the NVIDIA PX2			0%	2022-09-12	2022-09-20
Ensure data transfer from sensors to NVIDIA PX2			0%	2022-09-20	2022-09-30
Design a container for the NVIDIA PX2 inside the vehicle			0%	2022-09-02	2022-09-08
Establish a safe placement for the container inside the vehicle			0%	2022-09-08	2022-09-10
Establish a safe placement for the GUI set up inside the vehicle			0%	2022-09-08	2022-09-10
Design/obtain armature for mounting			0%	2022-09-10	2022-09-24
Determine final sensor placement			0%	2022-09-10	2022-09-13
Mount sensors on vehicle			0%	2022-09-10	2022-09-30
Phase 2: Interfacing and optimization					
Install ADE-EYE on NVIDIA PX2			0%	2022-09-10	2022-09-11
Interfacing the sensors to the AD-EYE software			0%	2022-09-11	2022-09-21
Interface CAN to the vehicle			0%	2022-09-11	2022-09-25
Interface AD-EYE towards simulated environment			0%	2022-09-21	2022-09-28
Visualize the vehicle state through the GUI			0%	2022-09-23	2022-09-26
Optimize sensor placement			0%	2022-09-26	2022-10-03
Tune sensors for real world environment			0%	2022-10-03	2022-10-13
Phase 3: Control and V2X communication					
Verify vehicle controls			0%	2022-10-13	2022-10-23
Control the vehicle through the GUI by cabled connection			0%	2022-10-13	2022-10-23
Establish connection between the RSU and NVIDIA PX2 via cable			0%	2022-10-19	2022-10-29
Establish connection between the RSU and NVIDIA PX2 via WLAN			0%	2022-10-20	2022-10-30
Establish connection between the RSU and NVIDIA PX2 via 5G			0%	2022-10-30	2022-11-14
Phase 4: Testing					
Ensure that localization and driving map are available			0%	2022-10-03	2022-10-18
Vehicle follows a short predefined trajectory			0%	2022-10-29	2022-11-18
Vehicle drives autonomously to a given goal			0%	2022-10-29	2022-11-23
Safety feature: Vehicle should stop when receiving emergency signal			0%	2022-11-23	2022-12-13



AD-EYE: Towards Autonomous Driving on KTH Campus

KTH

Project Start: Aug 29, 2022

Display Week: 5

TASK	ASSIGNED TO	TIME	PROGRESS	START	END
Phase 1: Set up sensors and sensor mounting					
Obtain all sensors and the commercial vehicle			0%	2022-08-29	2022-09-02
Elecrical set up and interface			0%	2022-08-29	2022-09-14
Connect the sensors to the NVIDIA PX2			0%	2022-09-12	2022-09-20
Ensure data transfer from sensors to NVIDIA PX2			0%	2022-09-20	2022-09-30
Design a container for the NVIDIA PX2 inside the vehicle			0%	2022-09-02	2022-09-08
Establish a safe placement for the container inside the vehicle			0%	2022-09-08	2022-09-10
Establish a safe placement for the GUI set up inside the vehicle			0%	2022-09-08	2022-09-10
Design/obtain armature for mounting			0%	2022-09-10	2022-09-24
Determine final sensor placement			0%	2022-09-10	2022-09-13
Mount sensors on vehicle			0%	2022-09-10	2022-09-30
Phase 2: Interfacing and optimization					
Install ADE-EYE on NVIDIA PX2			0%	2022-09-10	2022-09-11
Interfacing the sensors to the AD-EYE software			0%	2022-09-11	2022-09-21
Interface CAN to the vehicle			0%	2022-09-11	2022-09-25
Interface AD-EYE towards simulated environment			0%	2022-09-21	2022-09-28
Visualize the vehicle state through the GUI			0%	2022-09-23	2022-09-26
Optimize sensor placement			0%	2022-09-26	2022-10-03
Tune sensors for real world environment			0%	2022-10-03	2022-10-13
Phase 3: Control and V2X communication					
Verify vehicle controls			0%	2022-10-13	2022-10-23
Control the vehicle through the GUI by cabled connection			0%	2022-10-13	2022-10-23
Establish connection between the RSU and NVIDIA PX2 via cable			0%	2022-10-19	2022-10-29
Establish connection between the RSU and NVIDIA PX2 via WLAN			0%	2022-10-20	2022-10-30
Establish connection between the RSU and NVIDIA PX2 via 5G			0%	2022-10-30	2022-11-14
Phase 4: Testing					
Ensure that localization and driving map are available			0%	2022-10-03	2022-10-18
Vehicle follows a short predefined trajectory			0%	2022-10-29	2022-11-18
Vehicle drives autonomously to a given goal			0%	2022-10-29	2022-11-23
Safety feature: Vehicle should stop when receiving emergency signal			0%	2022-11-23	2022-12-13

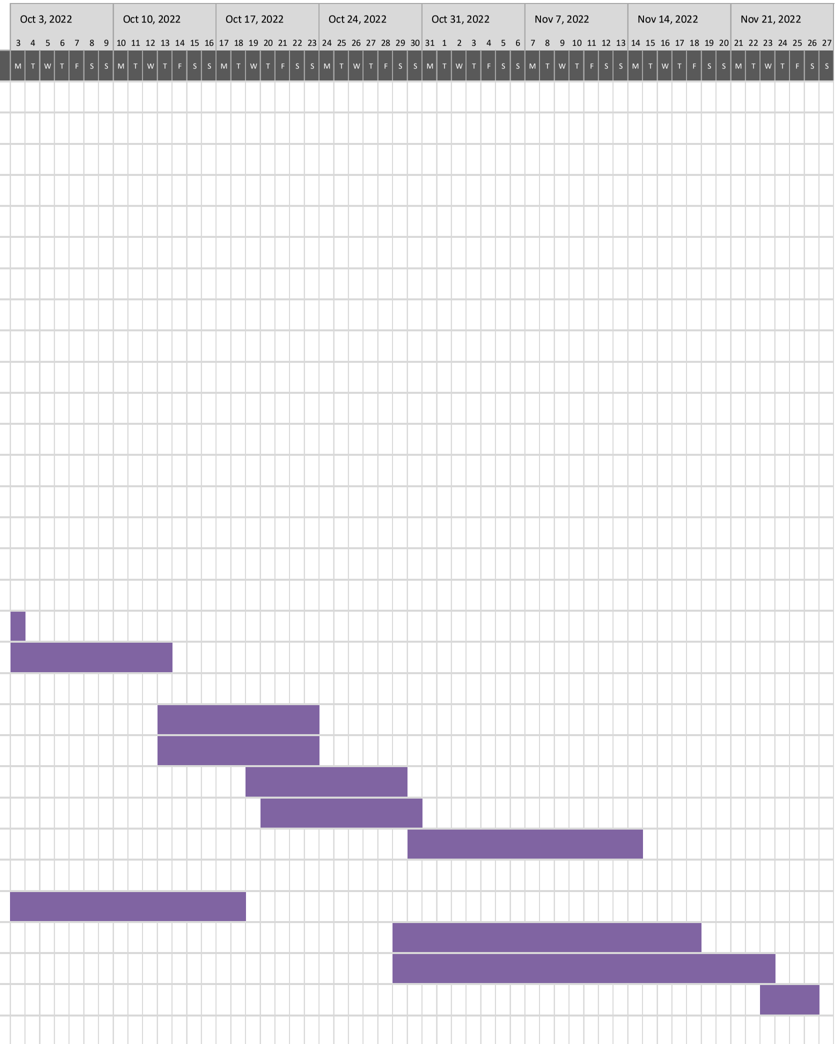
AD-EYE: Towards Autonomous Driving on KTH Campus

KTH

Project Start: Aug 29, 2022

Display Week: 6

TASK	ASSIGNED TO	TIME	PROGRESS	START	END
Phase 1: Set up sensors and sensor mounting					
Obtain all sensors and the commercial vehicle			0%	2022-08-29	2022-09-02
Elecrical set up and interface			0%	2022-08-29	2022-09-14
Connect the sensors to the NVIDIA PX2			0%	2022-09-12	2022-09-20
Ensure data transfer from sensors to NVIDIA PX2			0%	2022-09-20	2022-09-30
Design a container for the NVIDIA PX2 inside the vehicle			0%	2022-09-02	2022-09-08
Establish a safe placement for the container inside the vehicle			0%	2022-09-08	2022-09-10
Establish a safe placement for the GUI set up inside the vehicle			0%	2022-09-08	2022-09-10
Design/obtain armature for mounting			0%	2022-09-10	2022-09-24
Determine final sensor placement			0%	2022-09-10	2022-09-13
Mount sensors on vehicle			0%	2022-09-10	2022-09-30
Phase 2: Interfacing and optimization					
Install ADE-EYE on NVIDIA PX2			0%	2022-09-10	2022-09-11
Interfacing the sensors to the AD-EYE software			0%	2022-09-11	2022-09-21
Interface CAN to the vehicle			0%	2022-09-11	2022-09-25
Interface AD-EYE towards simulated environment			0%	2022-09-21	2022-09-28
Visualize the vehicle state through the GUI			0%	2022-09-23	2022-09-26
Optimize sensor placement			0%	2022-09-26	2022-10-03
Tune sensors for real world environment			0%	2022-10-03	2022-10-13
Phase 3: Control and V2X communication					
Verify vehicle controls			0%	2022-10-13	2022-10-23
Control the vehicle through the GUI by cabled connection			0%	2022-10-13	2022-10-23
Establish connection between the RSU and NVIDIA PX2 via cable			0%	2022-10-19	2022-10-29
Establish connection between the RSU and NVIDIA PX2 via WLAN			0%	2022-10-20	2022-10-30
Establish connection between the RSU and NVIDIA PX2 via 5G			0%	2022-10-30	2022-11-14
Phase 4: Testing					
Ensure that localization and driving map are available			0%	2022-10-03	2022-10-18
Vehicle follows a short predefined trajectory			0%	2022-10-29	2022-11-18
Vehicle drives autonomously to a given goal			0%	2022-10-29	2022-11-23
Safety feature: Vehicle should stop when receiving emergency signal			0%	2022-11-23	2022-12-13



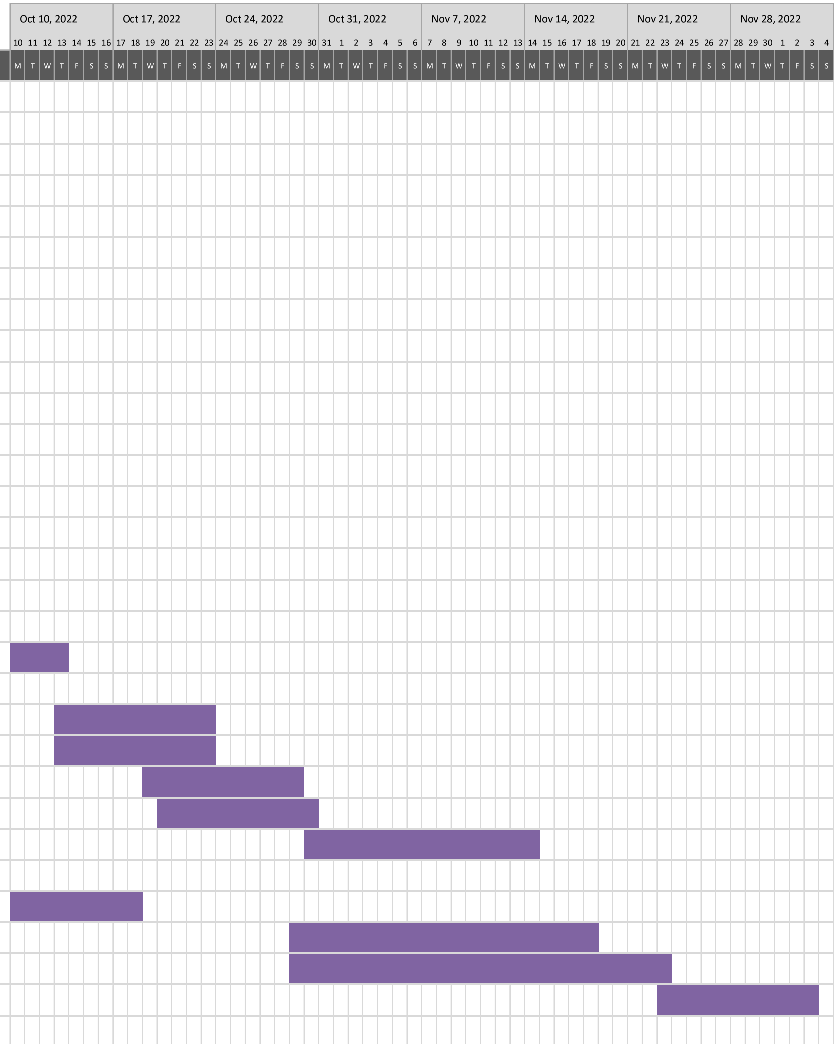
AD-EYE: Towards Autonomous Driving on KTH Campus

KTH

Project Start: Aug 29, 2022

Display Week: 7

TASK	ASSIGNED TO	TIME	PROGRESS	START	END
Phase 1: Set up sensors and sensor mounting					
Obtain all sensors and the commercial vehicle			0%	2022-08-29	2022-09-02
Elecrical set up and interface			0%	2022-08-29	2022-09-14
Connect the sensors to the NVIDIA PX2			0%	2022-09-12	2022-09-20
Ensure data transfer from sensors to NVIDIA PX2			0%	2022-09-20	2022-09-30
Design a container for the NVIDIA PX2 inside the vehicle			0%	2022-09-02	2022-09-08
Establish a safe placement for the container inside the vehicle			0%	2022-09-08	2022-09-10
Establish a safe placement for the GUI set up inside the vehicle			0%	2022-09-08	2022-09-10
Design/obtain armature for mounting			0%	2022-09-10	2022-09-24
Determine final sensor placement			0%	2022-09-10	2022-09-13
Mount sensors on vehicle			0%	2022-09-10	2022-09-30
Phase 2: Interfacing and optimization					
Install ADE-EYE on NVIDIA PX2			0%	2022-09-10	2022-09-11
Interfacing the sensors to the AD-EYE software			0%	2022-09-11	2022-09-21
Interface CAN to the vehicle			0%	2022-09-11	2022-09-25
Interface AD-EYE towards simulated environment			0%	2022-09-21	2022-09-28
Visualize the vehicle state through the GUI			0%	2022-09-23	2022-09-26
Optimize sensor placement			0%	2022-09-26	2022-10-03
Tune sensors for real world environment			0%	2022-10-03	2022-10-13
Phase 3: Control and V2X communication					
Verify vehicle controls			0%	2022-10-13	2022-10-23
Control the vehicle through the GUI by cabled connection			0%	2022-10-13	2022-10-23
Establish connection between the RSU and NVIDIA PX2 via cable			0%	2022-10-19	2022-10-29
Establish connection between the RSU and NVIDIA PX2 via WLAN			0%	2022-10-20	2022-10-30
Establish connection between the RSU and NVIDIA PX2 via 5G			0%	2022-10-30	2022-11-14
Phase 4: Testing					
Ensure that localization and driving map are available			0%	2022-10-03	2022-10-18
Vehicle follows a short predefined trajectory			0%	2022-10-29	2022-11-18
Vehicle drives autonomously to a given goal			0%	2022-10-29	2022-11-23
Safety feature: Vehicle should stop when receiving emergency signal			0%	2022-11-23	2022-12-13



AD-EYE: Towards Autonomous Driving on KTH Campus

KTH

Project Start: Aug 29, 2022

Display Week: 8

TASK	ASSIGNED TO	TIME	PROGRESS	START	END
Phase 1: Set up sensors and sensor mounting					
Obtain all sensors and the commercial vehicle			0%	2022-08-29	2022-09-02
Elecrical set up and interface			0%	2022-08-29	2022-09-14
Connect the sensors to the NVIDIA PX2			0%	2022-09-12	2022-09-20
Ensure data transfer from sensors to NVIDIA PX2			0%	2022-09-20	2022-09-30
Design a container for the NVIDIA PX2 inside the vehicle			0%	2022-09-02	2022-09-08
Establish a safe placement for the container inside the vehicle			0%	2022-09-08	2022-09-10
Establish a safe placement for the GUI set up inside the vehicle			0%	2022-09-08	2022-09-10
Design/obtain armature for mounting			0%	2022-09-10	2022-09-24
Determine final sensor placement			0%	2022-09-10	2022-09-13
Mount sensors on vehicle			0%	2022-09-10	2022-09-30
Phase 2: Interfacing and optimization					
Install ADE-EYE on NVIDIA PX2			0%	2022-09-10	2022-09-11
Interfacing the sensors to the AD-EYE software			0%	2022-09-11	2022-09-21
Interface CAN to the vehicle			0%	2022-09-11	2022-09-25
Interface AD-EYE towards simulated environment			0%	2022-09-21	2022-09-28
Visualize the vehicle state through the GUI			0%	2022-09-23	2022-09-26
Optimize sensor placement			0%	2022-09-26	2022-10-03
Tune sensors for real world environment			0%	2022-10-03	2022-10-13
Phase 3: Control and V2X communication					
Verify vehicle controls			0%	2022-10-13	2022-10-23
Control the vehicle through the GUI by cabled connection			0%	2022-10-13	2022-10-23
Establish connection between the RSU and NVIDIA PX2 via cable			0%	2022-10-19	2022-10-29
Establish connection between the RSU and NVIDIA PX2 via WLAN			0%	2022-10-20	2022-10-30
Establish connection between the RSU and NVIDIA PX2 via 5G			0%	2022-10-30	2022-11-14
Phase 4: Testing					
Ensure that localization and driving map are available			0%	2022-10-03	2022-10-18
Vehicle follows a short predefined trajectory			0%	2022-10-29	2022-11-18
Vehicle drives autonomously to a given goal			0%	2022-10-29	2022-11-23
Safety feature: Vehicle should stop when receiving emergency signal			0%	2022-11-23	2022-12-13

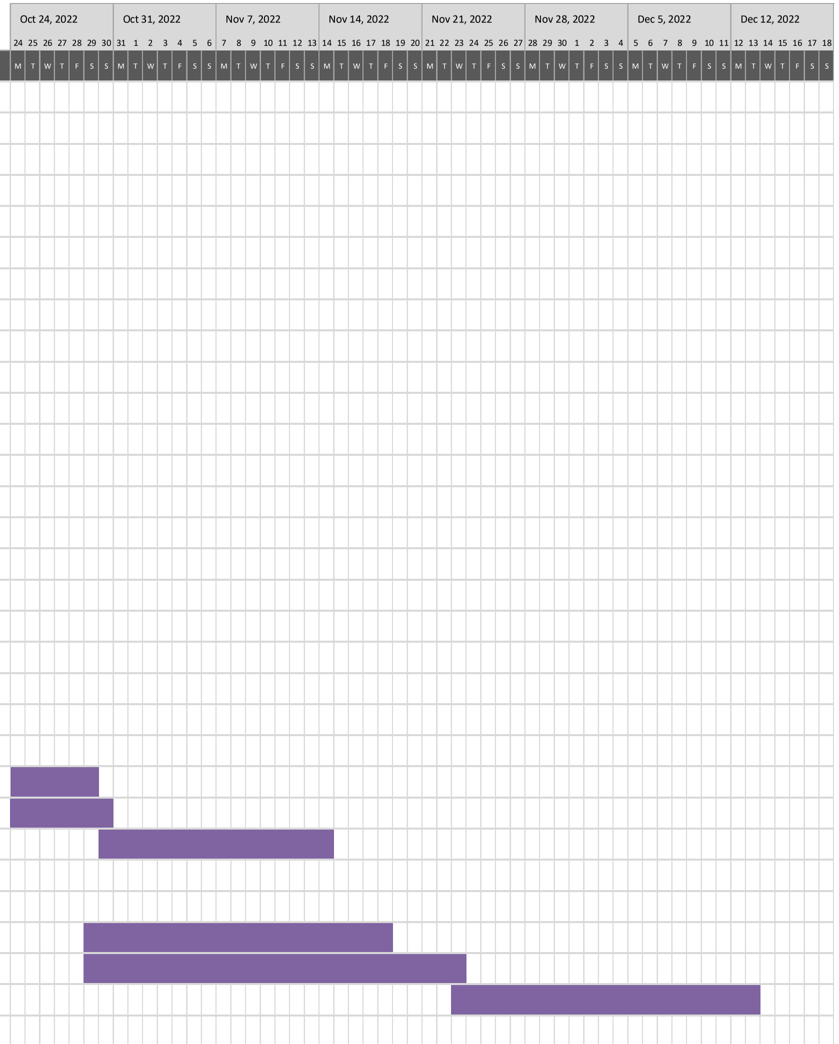
AD-EYE: Towards Autonomous Driving on KTH Campus

KTH

Project Start: Aug 29, 2022

Display Week: 9

TASK	ASSIGNED TO	TIME	PROGRESS	START	END
Phase 1: Set up sensors and sensor mounting					
Obtain all sensors and the commercial vehicle			0%	2022-08-29	2022-09-02
Elecrical set up and interface			0%	2022-08-29	2022-09-14
Connect the sensors to the NVIDIA PX2			0%	2022-09-12	2022-09-20
Ensure data transfer from sensors to NVIDIA PX2			0%	2022-09-20	2022-09-30
Design a container for the NVIDIA PX2 inside the vehicle			0%	2022-09-02	2022-09-08
Establish a safe placement for the container inside the vehicle			0%	2022-09-08	2022-09-10
Establish a safe placement for the GUI set up inside the vehicle			0%	2022-09-08	2022-09-10
Design/obtain armature for mounting			0%	2022-09-10	2022-09-24
Determine final sensor placement			0%	2022-09-10	2022-09-13
Mount sensors on vehicle			0%	2022-09-10	2022-09-30
Phase 2: Interfacing and optimization					
Install ADE-EYE on NVIDIA PX2			0%	2022-09-10	2022-09-11
Interfacing the sensors to the AD-EYE software			0%	2022-09-11	2022-09-21
Interface CAN to the vehicle			0%	2022-09-11	2022-09-25
Interface AD-EYE towards simulated environment			0%	2022-09-21	2022-09-28
Visualize the vehicle state through the GUI			0%	2022-09-23	2022-09-26
Optimize sensor placement			0%	2022-09-26	2022-10-03
Tune sensors for real world environment			0%	2022-10-03	2022-10-13
Phase 3: Control and V2X communication					
Verify vehicle controls			0%	2022-10-13	2022-10-23
Control the vehicle through the GUI by cabled connection			0%	2022-10-13	2022-10-23
Establish connection between the RSU and NVIDIA PX2 via cable			0%	2022-10-19	2022-10-29
Establish connection between the RSU and NVIDIA PX2 via WLAN			0%	2022-10-20	2022-10-30
Establish connection between the RSU and NVIDIA PX2 via 5G			0%	2022-10-30	2022-11-14
Phase 4: Testing					
Ensure that localization and driving map are available			0%	2022-10-03	2022-10-18
Vehicle follows a short predefined trajectory			0%	2022-10-29	2022-11-18
Vehicle drives autonomously to a given goal			0%	2022-10-29	2022-11-23
Safety feature: Vehicle should stop when receiving emergency signal			0%	2022-11-23	2022-12-13



B Risk Analysis

Risk Category	Risk Description	Probability of Occurrence	Risk Impact	Risk Exposure	Risk Response
Technical	On board computation failure (Including NVIDIA PX2, CAN system)	LOW	HIGH	MODERATE	Contact specialists to get help with the components
	Previous projects components not working probably (RSU)	MEDIUM	MEDIUM	MODERATE	Contacting the responsible team group or replacing the RSU with a stationary PC
	5G Cellular telecommunication not provided by stakeholders	MEDIUM	LOW	LOW	Replacing it with other wireless communication technology
	Not meeting the stakeholder's technical requirement	LOW	HIGH	MODERATE	Make sure to accomplish the basic requirements and inform the stakeholders
External	Long delivery time for the Commercial car	MEDIUM	HIGH	VERY HIGH	Contacting stakeholders
	Long delivery time for the sensors	MEDIUM	HIGH	VERY HIGH	Contacting stakeholders
	Weather changes during the validation phase affecting input data from sensors	MEDIUM	MEDIUM	MODERATE	Implementing filtering algorithms
Organizational	Group members not following the plan due to an overload in other courses	LOW	MEDIUM	LOW	Making sure that all members are working towards the same goal by implementing continuous meetings
Project management	Wrong estimation of the delivery time for the needed component leading to not following the project plan	MEDIUM	MEDIUM	MODERATE	Prioritize other tasks that are not affected by the delays
	Wrong estimation for the time needed to complete different tasks	LOW	MEDIUM	LOW	Create a new plan to rectify the situation