Chapter 1: Introduction

Chapter: 1

Introduction

Computer on Wheels, is a software system for a car that can drive themself with minimal intervention. The system is capable of autonomously controlling the vehicle's movement, including **throttle control**, **acceleration, braking, and steering**. Furthermore, it incorporates obstacle detection capabilities to detect and respond to obstacles ensuring safe navigation. Moreover, path planning algorithms are used to determine optimal routes from point A to point B. By leveraging state-of-the-art technologies such as the **CARLA (Car Learning to Act) simulator, CARLA-ROS bridge, and ROS (Robot Operating System)**, our project endeavours to create a software solution capable of empowering autonomous vehicles to navigate urban environment with precision and confidence.

1.1 Opportunity and Stakeholder

- According to a **National Highway Traffic Safety Administration (NHTSA)** study, driver error led to **94% of the crashes** examined.
- According to the **U.S. General Services Administration (GSA)**, human error causes **98% of crashes**.
- A 2017 study by **RAND Corporation** found that self-driving cars could reduce traffic fatalities by up to **25% by 2040**.
- A 2019 study by the National Highway Traffic Safety Administration (NHTSA) found that self-driving cars were involved in fewer crashes than human-driven cars per mile driven.
- A 2020 study by the **Massachusetts Institute of Technology (MIT)** found that self-driving cars could **prevent up to 90%** of crashes caused by human error.

1.1.1 Stakeholders

- Driver
- Passengers
- Maanz AI
- Domain Expert

1.2 Motivations and Challenges

Our project is motivated by the importance of enhancing safety for passengers, drivers, and pedestrians through autonomous vehicle technology. By alleviating the need for human drivers, we aim to enable multitasking and provide independence to individuals, including those with disabilities. Challenges such as time management and acquiring a physical model car for demonstrations were overcome by transitioning to the CARLA simulator. However, GPU resource limitations were encountered, which were addressed through assistance from **Maanz AI**, securing workspace and expert guidance.

1.3 Goals and Objectives

Our goals are clear: complete the project on time while ensuring high-quality deliverables and develop autonomous vehicle software to eliminate accidents caused by human error and enhance mobility for individuals with disabilities. These objectives will minimize errors, boost stakeholder productivity, and provide mobility for aged persons and people having disabilities.

1.4 Solution Overview

Our solution includes developing autonomous vehicle software utilizing cutting-edge technologies like the CARLA simulator, ROS Noetic, CARLA-ROS bridge, and rospy. This software will enable vehicles to autonomously navigate complex environments by implementing key functionalities:

- Path Planning
- Path Following
- Obstacle Detection
- Obstacle Avoidance

Also, ensuring safety and precision while minimizing accidents caused by human error. Additionally, our solution prioritizes accessibility, aiming to provide mobility for individuals with disabilities and the elderly. Through rigorous development and testing, we endeavour to deliver a reliable and efficient solution that revolutionizes autonomous vehicle navigation.

1.4.1 Project Scope

The scope of this project encompasses the development and implementation of key functionalities:

1.4.1.1 Integration

- Involve integrating various sensors and algorithms to enable the vehicle to perceive its environment accurately, make decisions, and navigate safely through dynamic scenarios.

1.4.1.2 Path Planning:

- Determining a feasible and shortest path from user-specified source and destination locations
- Implementing a navigation algorithm to handle dynamic environments and potential rerouting.

1.4.1.3 Path Following:

- Implementing control algorithms for precise vehicle guidance along the planned trajectory.
- Maintaining vehicle position and orientation relative to the path using steering, acceleration, and braking control.

1.4.1.4 Obstacle Detection:

- Utilizing sensor data (such as lidar, radar or cameras) to detect objects within the vehicle's surroundings.
- Employing algorithms/models to classify detected objects and assess their characteristics, such as size, shape, and distance.
- Integrating machine learning techniques to improve the accuracy and reliability of obstacle recognition.
- Providing real-time information about detected obstacles to inform path planning and navigation decisions.

1.4.1.5 Obstacle avoidance:

- Implement reactive obstacle avoidance strategies, allowing the autonomous vehicle to dynamically adjust its trajectory based on the detected obstacles, enabling safe navigation.
- Develop algorithms/maneuver for real-time analysis of obstacle data to facilitate swift decision-making by the autonomous vehicle.