University of Padova

Department of Information Engineering

Master Thesis in ICT for Internet and Multimedia

Improved Road Safety for Vulnerable Road Users Means

Supervisor

Andrea Zanella

University of Padova

Master Candidate

Maxwell Maloba

Student ID

2050787

Academic Year

2021-2022

"Dedication or Quote"
— To my family

Abstract

Road safety is crucial for the development of sustainable cities and communities. Recently, e-scooters have become a popular micro-mobility option; however, the shared road space between e-scooters and vehicles of varying speeds and sizes has led to numerous safety incidents.

This thesis addresses these safety concerns by developing a system that uses radar technology and Arduino microcontrollers to detect approaching vehicles on scooter bikes. The radar sensor collects distance data from nearby vehicles or obstacles, which is then processed locally to determine their presence and speed. The system provides real-time feedback to the rider, displaying vital information such as the distance, speed, and relative location (ahead or behind) of approaching vehicles.

An alarm system is activated for enhanced safety if the detected distance and speed indicate an imminent collision risk. This comprehensive approach aims to improve riders' situational awareness, allowing them to respond promptly to potential hazards, thus promoting the mobility-as-a-service paradigm.

The collected data is analyzed across different road segments to identify areas with higher traffic incident risks. Validation is performed to assess the accuracy of distance and speed measurements at various intervals. The thesis also discusses several limitations, offers recommendations for improvement, and shares lessons learned for future research and development.

Contents

Listing of figures

Listing of tables

Listing of acronyms

VRUs . . . . . Vulnerable Road Users

e-scooter . . . . . Electric Scooter

APA . . . . . Automatic Park Assistant

SRAM . . . . . Static Random Access Memory

VANET . . . . . Vehicular Ad-hoc Networks

1 Introduction

1.1 Background

The adoption of sustainable cities and communities by improving urban transport systems has been a major theme for many cities globally to create a well-connected, smooth mobility experience while reducing traffic and noise. To that end, the rise of micro-mobility in recent years has been an unexpected success story in the urban transport sector. In cities across Europe, the USA, and Asia, thousands of users are taking advantage of a growing range of micro-mobility options [1]. E-scooters, in particular, have grown in popularity at an incredible rate as a convenient and eco-friendly means of transportation, particularly for short trips inside urban areas. Their small size, user-friendly nature, and reasonable cost have made them an appealing choice for a diverse group of individuals, from daily commuters to vacationers.

Despite the advantages that e-scooters offer as a micro-mobility solution for improved urban transportation, they have been associated with safety risks due to frequent reports of traffic incidents, often resulting in e-scooter riders becoming casualties. Therefore, road safety has classified e-scooter riders as vulnerable.

With this in mind, Vulnerable Road Users (VRUs) are people who do not commute in motorized vehicles such as cars, buses, or trucks. They primarily include pedestrians, cyclists, motorbike riders, e-scooter riders, children, and the elderly. These individuals are especially susceptible to traffic incidents due to their limited protection from traffic [2].

This thesis will focus primarily on e-scooter riders. Since the emergence of e-scooters and their rapid expansion, there has been limited and scattered information about their safety. There have been various strategies put in place to address VRU safety; one of these strategies includes implementing a speed limit for e-scooters, and it was found that speeds between 15 km/h and 20 km/h are considered safe in reducing traffic occurrences [3]. This is because riders have better control and are less likely to sustain serious injuries in the event of a traffic incident. The implementation of micro-mobility lanes in different sections of urban areas has also significantly improved VRU safety by separating pedestrians from motor vehicles. Furthermore, there has been a strong emphasis on promoting a general understanding of road safety when using e-scooters, encouraging VRUs to prioritize their safety.

In contrast, in collisions with passenger cars, the speed of the vehicle is a major factor in the level of injury risk. Collisions with cars traveling above 40 km/h can cause severe to fatal injuries to e-scooter riders [4]. Despite numerous attempts to enhance the safety of VRUs, there has been a lack of effective solutions for detecting approaching vehicles in traffic, a problem that remains unresolved. This thesis seeks to tackle this issue by employing low-cost sensors to implement radar technology, a strategy that has yielded promising results from existing research and developments in radar technology focusing on the detection and processing of data, as we will discuss in the following sections.

1.2 Related Work

The topic of detection by applying ultrasonic and radar technologies has undergone significant research and development, highlighting its critical role in improving safety and automation in a variety of applications. This section dives into the various studies and developments that have taken place in this field. Several studies have explored the use of ultrasonic sensors due to their cost and simplicity, particularly in automotive and robotics applications. Meanwhile, the improved accuracy and performance of radar technology in unfavorable situations have led to its use in increasingly complex and demanding environments. By evaluating these contributions, we obtain insight into each technology's strengths and limits, as well as the novel ways researchers have combined them to create effective detection systems.

1.2.1 Parking Assist Systems

Auto Parking Assist (APA), an intelligent parking assistance system, uses radar technology to automatically and safely steer the vehicle into a parking space [5]. The system recognizes the parking space by sensing information about the vehicle's surroundings through ultrasonic and image sensors. It generates a corresponding parking trajectory based on the relative positions of the vehicle and the parking space. The system then controls the vehicle's speed and steering wheel to complete the automatic parking process [6]. The automatic parking system uses ultrasonic radar to monitor environmental obstacles, parking spaces, and other information. Based on the received data, the APA controller makes corresponding judgments and controls the vehicle's steering and braking.

The system operates by emitting radio waves that reflect off nearby objects and return to the radar sensor. It then calculates the distance and relative speed of these objects based on the reflected signals' time delay and frequency shift. This information provides the driver with real-time feedback, typically in the form of visual and auditory warnings. Advanced driver-assistance systems (ADAS) integrate modern radar-based parking assist systems, enabling them to perform a variety of functions [7].

Obstacle Detection: Radar sensors can detect obstacles in the vehicle's path, including low-lying objects that may not be visible to the driver. This feature is particularly useful when parking in tight spaces or near curbs.

Automatic Braking: Some systems can autonomously apply the brakes if an obstacle is detected, preventing potential collisions during parking maneuvers.

Guided Parking: Advanced systems offer guided parking assistance, where the vehicle autonomously steers into a parking space while the driver controls the accelerator and brakes. This functionality is made possible by combining radar data with other sensors and cameras for comprehensive environmental awareness.

360-Degree Surveillance: Combining multiple radar sensors around the vehicle allows for a 360-degree view, providing enhanced situational awareness and reducing blind spots.

Recent advancements in radar technology have further improved the performance and reliability of parking assist systems. Key developments include frequency radars, where modern systems utilize high-frequency radars (77 GHz) compared to earlier models (24 GHz) [8]. Higher frequencies offer better resolution and accuracy, enabling the detection of smaller objects and more precise distance measurements. Use multi-mode radars that can operate in different modes, such as long-range and short-range detection, providing versatility in a variety

of parking scenarios. Long-range mode is useful for detecting obstacles when approaching a parking space, while short-range mode ensures precision during the final parking maneuvers.

By incorporating AI and machine learning algorithms, radar-based systems can more accurately predict the movement of nearby objects and pedestrians, offering enhanced safety features. These systems can learn and adapt to different parking environments, improving their performance over time [9].

Advances in semiconductor technology have led to smaller and more cost-effective radar sensors, making it feasible to equip a broader range of vehicles with advanced parking assist features[10]. Radar technology has revolutionized parking assist systems, providing superior obstacle detection, automatic braking, guided parking, and comprehensive situational awareness. The evolution from ultrasonic to radar-based systems marks a significant improvement in both safety and convenience for drivers[11].

1.2.2 Road Safety Alerting System with Radar and GPS Cooperation in a VANET Environment

Automotive safety systems have long used radar technology to detect surrounding objects. Its application to VRU safety is a more recent development. Modern radar systems can identify the location, speed, and trajectory of VRUs, providing critical data to prevent collisions. These systems operate effectively in poor visibility conditions, such as fog, rain, or darkness, which is a significant advantage over optical systems like cameras.

This study explores a novel approach to integrating radar technology with vehicle ad hoc networks (VANETs) and GPS to improve road safety. This study aims to meet market demands and foster collaboration among on-board devices.

The vehicles are equipped with on-board units (OBU) and on-board radar units (OBRU), which can send alert messages throughout the network concerning warnings and dangerous conditions exploiting the IEEE 802.llp standard[12]. The vehicles travel along roadways, taking note of the environment, traffic, road conditions, and vehicle parameters. This information can be expanded and shared with neighboring automotives, roadside units (RSUs), and, of course, the Internet, enabling inter-system communications via a Road Traffic Manager (RTM) [13]. Radar systems are responsible for detecting the environment to enhance comprehension of the current road conditions. Once the surrounding objects have been identified using the onboard devices, the exact position of the captured event can be determined, and the information is distributed over the network following data elaboration. Upon notification,

drivers can take preventive measures, such as reducing travel speed or altering their route path.

In [12], the author outlines several key contributions to the field: The work provides a detailed system architecture for a road safety alerting system that uses both radar and GPS data. This hybrid method improves the system's ability to detect VRUs and provide timely alerts to drivers and other road users. The study also underlines the significance of data fusion, which combines radar and GPS data to improve the accuracy of VRU detection. This fusion addresses the limitations of each technology when used in isolation, such as GPS inaccuracies in urban canyons or radar's limited range. The system uses VANET to enable real-time communication between vehicles and infrastructure, enabling an immediate response to potential hazards. This real-time feature is crucial for avoiding accidents involving VRUs, who are often less protected and volatile than motor vehicle occupants.

The integration of radar technology with VANETs and GPS represents a significant advancement in improving VRU safety [14], [15]. By leveraging the strengths of radar and GPS and enabling real-time data sharing through VANETs, this approach offers a promising solution to the improved safety of vulnerable road users.

1.2.3 Blind Spot Detection Systems

With the advancement of Sustainable Mobility as a Service (SMaaS), it is increasingly important to assure safety through technology that can detect surrounding vehicles and respond to potentially dangerous situations while driving. Before changing lanes, drivers are taught and expected to assess surrounding traffic by checking their rear-view and side mirrors and looking over each shoulder. However, even for those who follow this series of checks, the vehicle's blind spot, most commonly the area just behind and alongside the car, is a persistent source of potential risks and often the cause of traffic incidents [16]. The side mirrors do not fully cover the blind spot areas, or when a driver turns to look behind while on the steering wheel, the driver's view is obscured by the pillar between the front and back seats of the vehicle. To improve safety, car manufacturers have long explored new ways to address this design issue, which is where the blind spot detection system comes in.

The system monitors vehicles approaching from the sides or behind. When the driver signals a change of lane, the system alerts them in various ways, depending on the car model. A flashing light on the wing mirror or the interior front pillar is accompanied by either a warning sound or a slight vibration in the steering wheel. The system also assists the driver during overtaking maneuvers when returning to the original lane. The alert remains active until the vehicle beside

the driver moves ahead or is directly alongside and visible to the driver. Even without signaling, the system warns the driver about approaching adjacent traffic by keeping the light steady.

The blind spot detection system is implemented using ultrasonic or corner radar sensors. These sensors are strategically placed to cover the hazardous blind spot area [17], which is often invisible to the driver through conventional mirrors. When another vehicle enters the monitored area, the system alerts the driver to the potential danger through a warning signal displayed in the side mirror. This immediate feedback allows the driver to take corrective action to avoid a collision. If the driver ignores or disregards this initial warning and activates the turn signal to change lanes, the system escalates the alert by triggering an additional, more pronounced warning. This multi-tiered alert system enhances driver awareness and safety by providing ample opportunities to recognize and respond to potential hazards.

Moreover, the system is engineered to differentiate between moving vehicles and stationary objects. It can identify stationary objects on or alongside the road, such as guardrails, masts, or parked vehicles, as well as the driver's overtaking maneuvers. This capability ensures that unnecessary warnings are not triggered, which could otherwise lead to driver desensitization to the alerts. By filtering out stationary objects, the system maintains a high level of accuracy and reliability in real-time traffic conditions.

The advanced variant of the blind spot detection system incorporates two corner radar sensors that are concealed within the rear bumper, one on the left and one on the right side of the vehicle. These sensors are designed to monitor the areas alongside and behind the vehicle comprehensively. The advanced control software integrates the data collected by these sensors to create a detailed and dynamic representation of the traffic environment behind the vehicle. This integration allows the system to track multiple vehicles simultaneously, providing a more robust safety net for the driver.

In comparison to the ultrasonic sensors, the rear corner radar sensors offer an extended sensing range. This extended range is particularly beneficial in detecting fast-approaching vehicles, which might not be identified early enough by ultrasonic sensors. The advanced radar-based system can thus detect these vehicles at an earlier stage, well before the driver initiates a lane change. This early detection capability is crucial in preventing potential collisions, especially in high-speed traffic conditions where reaction time is critical.This technology not only improves driver awareness but also actively contributes to reducing accidents and enhancing road safety for all users [18].

1.3 Objectives

Accurately identifying and minimizing the probability of VRUs encountering potential traffic incidents is an important challenge. Unlike larger vehicles equipped with advanced safety features, e-scooters lack mechanisms to alert VRUs to approaching vehicles or obstacles, particularly from the rear. This lack of awareness can lead to accidents, as motor vehicles may pass e-scooters unnoticed, thereby jeopardizing the safety of the VRUs. This thesis aims to address the critical issue of road safety for VRUs, particularly e-scooter riders, by developing and evaluating a viable detection system with comprehensive objectives.

The primary objective of this thesis is to enhance the safety of e-scooter riders, who are particularly vulnerable due to the mixed traffic environment they navigate. By developing a system that alerts riders to the presence of approaching vehicles, this research aims to reduce the incidence of accidents and improve overall road safety by developing a detection system using radar technology and Arduino microcontrollers. These technologies are integrated into e-scooters, allowing for the collection of distance data from nearby vehicles or obstacles. The system is designed to process this data locally, providing real-time feedback to the rider about the presence and speed of approaching vehicles or obstacles.

Another significant goal of this study is to provide real-time feedback to VRUs. By informing riders about crucial parameters such as the distance, speed, and relative location (ahead or behind) of approaching vehicles, the system enables them to make informed decisions and take timely actions to avoid potential hazards. We are also implementing a safety alarm system that activates when the calculated distance and speed values fall within a critical range, indicating an imminent collision risk. This alarm system enhances the rider's situational awareness and ability to respond promptly to potential dangers.

By improving safety measures for e-scooter riders, this thesis supports the broader objective of promoting sustainable and efficient urban mobility solutions. The enhanced safety provided by the developed system aligns with the Mobility as a Service (MaaS) paradigm, which advocates for integrated, user-centric, and sustainable transport services.

An important aspect of this research is the collection and analysis of data across different road segments. By understanding the behavior of the collected measurements, the study aims to identify which roads have a higher likelihood of traffic incidents involving e-scooters. This analysis can inform targeted safety improvements and urban planning decisions, including the accuracy of the detection system in measuring distance and speed at various intervals. Assessing the accuracy and reliability of these measurements is critical to ensuring the system's effective-

ness in real-world conditions and building confidence in its practical application.

The research identifies several limitations of the current detection system, discusses these constraints in detail, and provides recommendations for future improvements. Addressing these limitations is essential for the ongoing development and refinement of safer micro-mobility solutions, with the ultimate goal of contributing to the broader goal of adopting sustainable cities and communities. By addressing the safety concerns associated with e-scooters, the research aligns with global urban development goals, promoting safer and more sustainable urban mobility options.

2

Detection of Approaching Vehicles

The detection of approaching vehicles is an important aspect of intelligent transportation systems, aiming to improve and enhance road safety. This chapter explores the technologies and methodologies utilized in identifying the presence of vehicles as they approach a vulnerable road user. By exploiting real-time data processing and sensor technologies, this research seeks to develop a reliable system that is capable of accurately detecting and responding to the presence of vehicles approaching a vulnerable road user. The chapter will explore how the detection of approaching vehicles (DAVE) was implemented. We will explore the components and devices used, the code implementation, and the experimental procedures of data acquisition and sensor placement.

2.1 COMPONENTS

2.1.1 Ultrasonic sensor HCSR04

The HC-SR04 ultrasonic sensor is a low-cost, low-power sensor device that can detect distance measurements for up to 5 meters. The sensor employs ultrasonic sound, which is a high-frequency sound wave that surpasses the perceptible range of human hearing. Humans have the ability to perceive sound waves that oscillate at frequencies ranging from 20 to 20,000 hertz (a high-pitched whistle). Ultrasound, at a frequency exceeding 20,000 Hz, is not audible to humans.

Figure 2.1: HC-SR04 ultrasonic ranging module

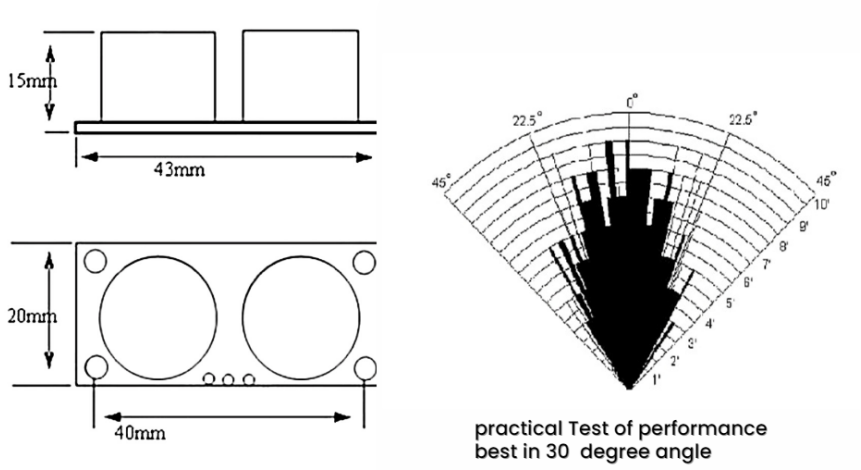


Figure 2.2: Schematics and Performance Range of the Ultrasonic Sensor Module

The HC-SR04 module consists of 4 pins:

- VCC – supplies power to the sensor and is connected to the 5V output of the Arduino board
- Trig pin – Triggers ultrasonic sound pulses. The pin receives a HIGH (5v) pulse for 10μs, thus initiates an ultrasonic burst from the transmitter.
- Echo pin -Pin goes high when the ultrasonic burst is transmitted and remains high until the sensor receives an echo
- GND – ground pin and is connected to the ground of Arduino

The principle of working behind the ultrasonic sensor is that the sensor emits ultrasonic sound waves and waits for the reflected sound waves. The device has two electronic transducers: one is the transmitter that sends the sound waves, and the other is the receiver transducer that listens to the reflected sound waves. The measurement process is initiated when the trigger pin receives a high-voltage pulse from the connected microcontroller unit for 10 microseconds, which in turn sets the pin state to HIGH. This will trigger the transmission of 8 bursts of ultrasonic pulses at 40 kHz. The 8-pulse pattern is designed so that the echo can differentiate between the transmitted and reflected pulses.

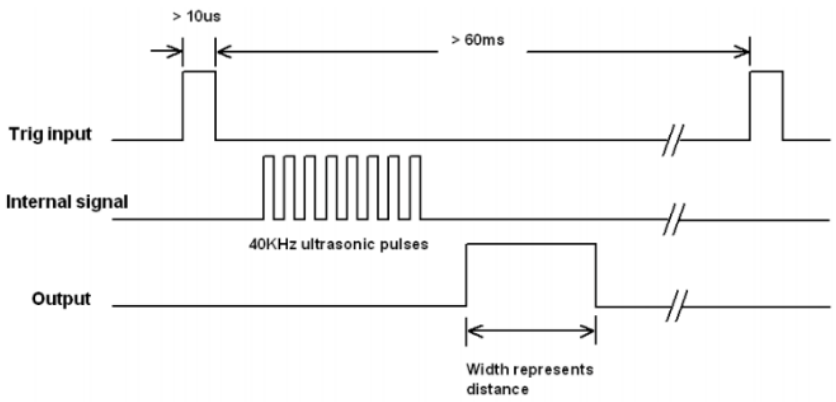


Figure 2.3: Timing Diagram for HC-SR04 Ultrasonic Sensor

The pulse width directly relates to the distance between the sensor and the target object. The sensor measures the time between sending the pulse and receiving the echo. This time interval can be reffered to as time of flight (ToF). The distance is calculated using the speed of sound (SS) in air (approximately 343 m/s) and the measured time interval. Since the sound wave travels to the object and back, the distance to the object is half the total distance traveled by the sound wave.

$$Distance = \frac{SS * TOF}{2} \tag{2.1}$$

To use the HC-SR04 sensor, it needs to be connected to an Arduino board. some basic Wiring needs to be followed. The ground and VCC pins of the module are connected to the ground and 5-volt pins on the Arduino Board, respectively, while the trig and echo pins are connected to any digital input or output pin on the Arduino Board.

The code that implements distance measurements while using the HC-SR04 ultrasonic sensor and Arduino board is as follows.

```
// Define variables:
#define trigPin 2
#define echoPin 3
float duration, distance;

void setup() {
    // Define inputs and outputs:
    pinMode(trigPin, OUTPUT);
    pinMode(echoPin, INPUT);
    //Begin Serial communication at a baudrate of 9600:
    Serial.begin(9600);
}
void loop() {
    // Clear the trigPin by setting it LOW:
    digitalWrite(trigPin, LOW);
    delayMicroseconds(5);
    // Trigger the sensor by setting the trigPin high for 10 microseconds:
    digitalWrite(trigPin, HIGH);
    delayMicroseconds(10);
    digitalWrite(trigPin, LOW);
    /Measuree the duration of the echo pulse
    duration = pulseIn(echoPin, HIGH);
    // Calculate the distance:
    distance = duration * 0.034 / 2;
    // Print the distance on the Serial Monitor:
    Serial.print(distance);
    delay(50);
}
```

Listing 2.1: Arduino Code for Ultrasonic Sensor

The code operates by first defining the `trigPin` and `echoPin`. These pins are connected to digital pins on the Arduino board and are defined as constant values using the `#define` directive. The compiler will replace any references to these constants during program execution.

Next, we define the `duration` and `distance` variables. The `duration` variable records the time interval between sending and receiving the sound waves, while the `distance` variable stores the calculated distance.

In the `setup()`, we configure the `echoPin` as an input and the `trigPin` as an output. We then initialize serial communication at a baud rate of 9600, which will be used to display the readings on the serial monitor. `loop()`, we trigger the sensor by setting the `trigPin` to a HIGH state for

10 microseconds. We then set the `trigPin` to a LOW state for 5 microseconds to ensure a clean signal. The length of the pulse is measured using the `pulseIn()` function, which times the duration the `echoPin` remains HIGH.

After obtaining the pulse duration, we calculate the distance using the previously mentioned formula, and the result is printed to the serial monitor. Before the sensor can take another measurement, it needs to settle and stabilize. The delay parameter is used to introduce a brief delay between measurements to allow the sensor to settle and ensure that the previous measurement has been completed before the next one begins.The length of the delay parameter may vary depending on the specific implementation and the characteristics of the sensor being used. It is typically set to a few milliseconds to ensure reliable and accurate measurements.

2.1.2 Arduino MKR WiFi 1010

The Arduino MKR WiFi 1010 is a powerful IoT device that was designed to make the process of developing IoT applications and nodes communicating in IP protocols and as Bluetooth clients based on Wi-Fi connectivity much faster and easier through libraries provided for WiFi and Bluetooth connectivity. This was made possible thanks to the flexibility of the ESP32 module of the U-BLOX NINA-W10 and ARM Cortex-M0 (SAMD21) processors, together with their low power consumption on a single compact board. The board can be implemented in vast applications ranging from building a sensor network connected to your office or home router to creating a Bluetooth low-energy device sending data to a cellphone. The MKR WiFi 1010 offers a unified solution for many of the basic IoT application scenarios.

The board is made up of three building blocks:

- **Arm® Cortex®-M0 32-bit SAMD21 processor**

  This microcontroller is the heart of the Arduino MKR WiFi 1010. It is significant for a variety of applications due to its balance of flexibility, power efficiency, and performance. The architecture of the SAMD21 Cortex-M0+ Microcontroller is designed to offer high performance while maintaining low power consumption. It is ideal for embedded applications that require efficient processing capabilities. It operates at a clock speed of 48 MHz, providing sufficient processing power for various tasks while keeping power consumption low. It has 256 KB of flash memory available for storing the program code. This non-volatile memory ensures that the program is retained even when the power is turned off. 32 KB of SRAM is available for dynamic data storage during program execution. This memory is used for variables, stack, and other runtime data. The microcontroller also offers interfaces such as the digital I/O Pins: It provides 8 digital input/output pins, which can be configured as either inputs or outputs as needed. These

pins also support Pulse Width Modulation (PWM) for tasks such as dimming LEDs or controlling motors and a USB Interface, allowing for easy connection to a computer for programming and data transfer.

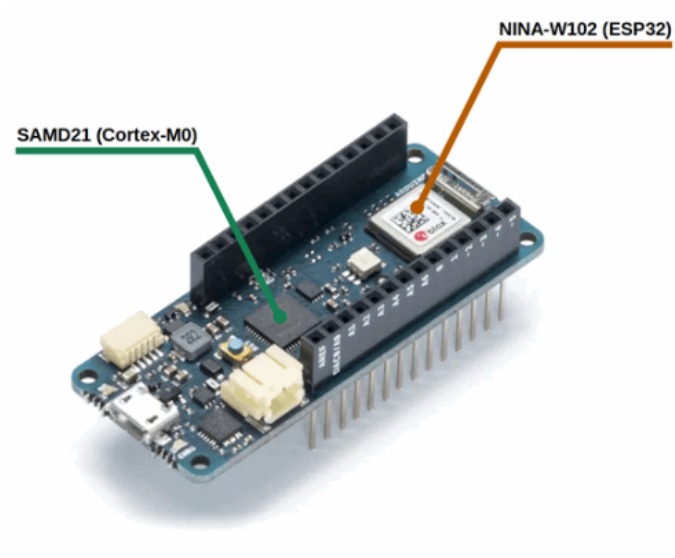


Figure 2.4: Arduino MKR WiFi 1010 Module

- **ECC508 CryptoAuthentication**

  It includes a cryptographic co-processor designed by Microchip Technology. This component is essential to guarantee secure communications and data integrity in IoT applications by providing advanced security features that protect against cyber threats. The ECC508 co-processor implements data security and integrity by utilizing cryptographic algorithms such as the SHA-256 hashing algorithm and random number generation algorithm. The former is used for creating unique, fixed-size hash values from input data useful for data integrity checks and digital signature while the latter is essential for generating cryptographic keys and nonces, ensuring the unpredictability and security of cryptographic operations. The ECC508 cryptographic co-processor significantly enhances the security capabilities of the Arduino MKR WiFi 1010. By providing robust cryptographic functions, secure key storage, and advanced security features, the ATECC508A ensures the integrity, authenticity, and confidentiality of data in IoT applications. Its integration into the Arduino platform simplifies the implementation of complex security measures, allowing developers to focus on building innovative and secure connected devices.

- **U-BLOX NINA-W10 Series Low Power 2.4GHz IEEE® 802.11 b/g/n Wi-Fi**

The Arduino MKR WiFi 1010 leverages the U-Blox NINA-W102, an all-around and functional Wi-Fi module designed to provide reliable wireless connectivity for many IoT applications. This module is a key component that enables the board to communicate with Wi-Fi networks, facilitating communication, remote control functionalities and data exchange. U-Blox NINA-W102 can achieve Wireless Connectivity by supporting the IEEE 802.11 b/g/n network standards, ensuring compatibility with most Wi-Fi networks. It operates in the 2.4 GHz ISM band, which is widely used and supported. The module can also operate in various modes, including access point (AP), client (STA), and simultaneous STA+AP modes. This flexibility allows it to connect to existing networks or create its network for direct device-to-device communication. The module includes an integrated TCP/IP stack that can handle the lower-level networking protocols. This balances the load on the microcontroller making it easier to develop networked applications. Regarding data speed, the module can support data rates of up to 72.2 Mbps in 802.11n mode, providing sufficient bandwidth for most IoT applications. The U-Blox NINA-W102 Wi-Fi module is a crucial component of the Arduino MKR WiFi 1010, offering robust wireless connectivity essential for modern IoT applications. Its support for various Wi-Fi modes, secure communication protocols, and low power consumption makes it suitable for smart home devices, industrial automation, and environmental monitoring.

To use the uBlox NINA W102 wifi module on the board, you'll need to install the library called WiFiNINA using Tools > Manage Libraries in the Arduino IDE.

2.1.3 ESP32-WROOM-32

The ESP32-WROOM-32 board is a powerful, Wi-Fi + Bluetooth® + Bluetooth LE low-cost module designed by Espressif Systems, that provides a versatile platform for a wide variety of IoT and embedded applications ranging from low-power sensor networks to the most demanding tasks, such as music streaming, voice encoding, and MP3 decoding.

At the core of this module is the ESP32-D0WDQ6 chip*. The chip embedded is designed to be scalable and adaptive. Two CPU cores can be individually controlled, and the CPU clock frequency is adjustable from 80 MHz to 240 MHz. The chip also has a low-power co-processor that can be used instead of the CPU to save power while performing tasks that do not require much computing power, such as monitoring of peripherals. ESP32 integrates a rich set of peripherals, ranging from capacitive touch sensors, SD card interface, Ethernet, high-speed SPI, UART, I2S, and I2C.

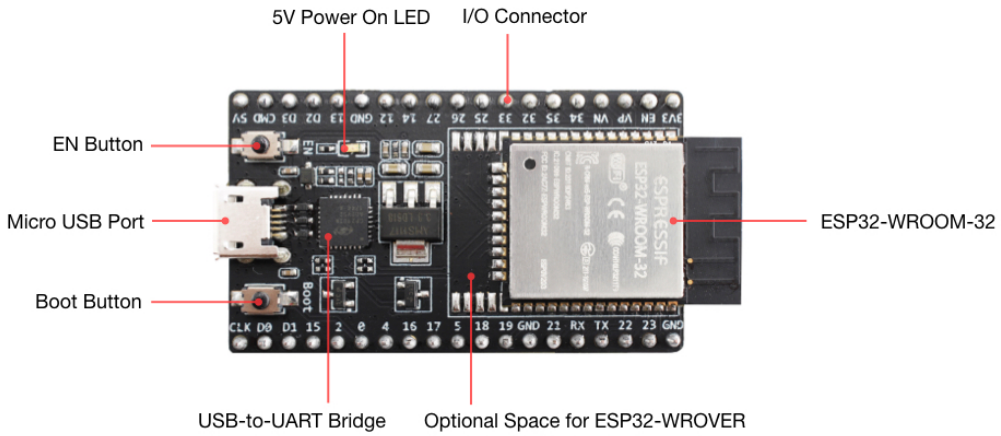


Figure 2.5: ESP32-WROOM-32 module

One of the main features of the ESP32-WROOM-32 is its integrated wireless communication capabilities. The module supports Wi-Fi 802.11 b/g/n standards, enabling robust and high-speed internet connectivity as well as Bluetooth which supports both Classic Bluetooth v4.2 and Bluetooth Low Energy (BLE), The feature provides options for wireless communication with various devices.

The ESP32-WROOM-32 module is equipped with numerous peripheral interfaces, making it highly versatile for various applications:

- General Purpose Input/Output (GPIO): 34 GPIO pins, which can be configured for input, output, and special functions.
- Analog-to-Digital Converter (ADC): Two 12-bit SAR ADCs with up to 18 channels, allowing for high-resolution analog signal conversion.
- Digital-to-Analog Converter (DAC): Two 8-bit DACs, provide analog signal output.
- Communication Interfaces: Including SPI, I2C, I2S, and UART, supporting various data communication protocols.
- Touch Sensors: 10 capacitive sensing GPIOs for touch input detection. Pulse Width Modulation (PWM): For controlling the power delivered to electrical devices, such as LEDs and motors.
- CAN Bus: For automotive and industrial applications requiring robust communication between multiple microcontrollers.

Programming and developing programs for the ESP32-WROOM-32 is simplified by a range of development environments and tools, making it suitable for both novice and professional developers.

Programming on the ESP32 module can be implemented through development environments including Arduino IDE, ESP-IDF which is Espressif's official IoT Development Framework, and Platform IO. The environment used for this research will be the Arduino IDE.

2.1.4 20x4 Liquid Crystal Display Module

The 20x4 LCD is a popular module in microcontroller projects, particularly for Arduino boards. The reasons are that LCDs are inexpensive, easily programmable, and have no limitations on displaying special and even customized characters, animations. This display module supports the presentation of text and simple graphics, providing a convenient way to interact with and receive feedback from embedded systems.

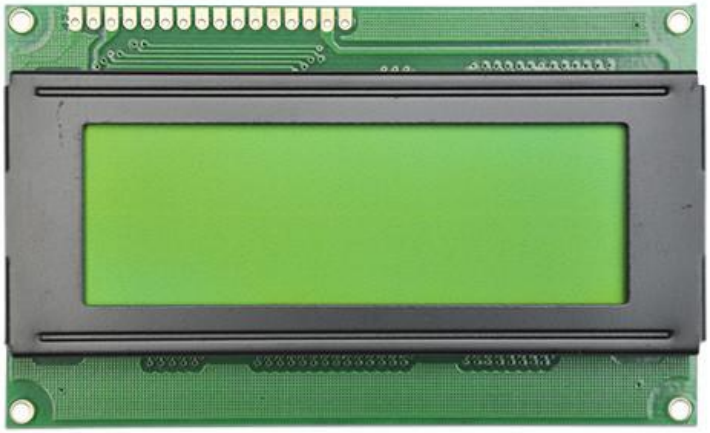


Figure 2.6: 20x4 LCD Module, Character LCD Display 20x4

The 20x4 LCD means that it can display 20 characters per line across four lines. It uses liquid crystal technology to produce visible text on a small screen, with each character typically displayed in a 5x8 pixel matrix. The most common controller used in these displays is the HD44780, which is compatible with many microcontrollers, including Arduino.

The interfacing between the 20x4 LCD and an Arduino can be achieved by using a library called `LiquidCrystal`. This library streamlines the process of sending commands and data to

the display. A basic example of how to connect and program the 20x4 LCD with an Arduino is illustrated below.

```
#include <LiquidCrystal.h>

// Initialize the library with the numbers of the interface pins
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);

void setup() {
  // Set up the LCD's number of columns and rows
  lcd.begin(20, 4);
  // Print a message to the LCD
  lcd.print("Improved Safety for Vulnerable Road Users Means");
}

void loop() {
  // Set the cursor to column 0, line 1
  lcd.setCursor(0, 1);
  // Print a message to the second row
  lcd.print("Safety for All!");
}
```

Listing 2.2: Arduino Code for Ultrasonic Sensor

2.2 Building the System

This section details the assembly and integration of the various components used in the project, describing the technologies employed and their interconnections to achieve the desired functionality. The system comprises several components, including HCSR04 ultrasonic sensors, an MKR-WIFI-1010 board, an ESP32-WROOM-32 board, and a 20x4 LCD. The combination of these components enables real-time sensing and data display for enhancing the safety and awareness of VRUs.

The system employs two HCSR04 ultrasonic sensors, one is mounted and installed on the back end of the e-scooter, behind the wheel, to detect approaching vehicles. The second sensor is positioned at the front of the scooter to identify objects from the side. These placements ensure thorough monitoring of the scooter's immediate environment, providing timely alerts to the rider.

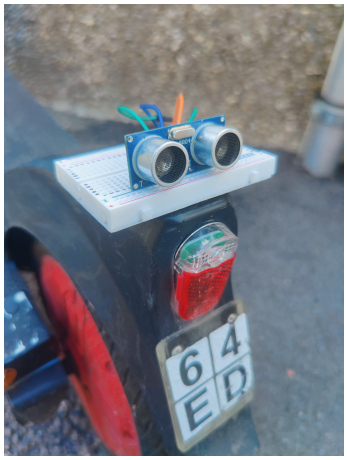
The MKR-WIFI-1010 board is utilized for real-time processing and sensing of the data from

the HCSR04 sensors. This board was selected for its wireless communication capabilities and compatibility with the ultrasonic sensors. Positioned centrally within the scooter's electronics housing, the MKR-WIFI-1010 board acts as a client, transmitting the processed distance data to the ESP32-WROOM-32 board.

The ESP32-WROOM-32 board functions as the server, receiving data from the MKR-WIFI-1010 board and relaying it to the 20x4 LCD display. This board was chosen for its robust processing power and ease of integration with both the MKR-WIFI-1010 board and the LCD display. The 20x4 LCD display is mounted near the handlebar, within the rider's line of sight, to provide real-time distance readings from both the rear and front sensors.

Figure 2.7: e-scooter bike

(a)



(b)

(c)

Figure 2.8: Ultrasonic sensor and display sensor placement

3

Data Collection

The chapter delves into the methods and techniques employed to gather the necessary data for the project. It highlights the process of capturing and organizing data relevant to the system's functionality and objectives. Specifically, This chapter illustrates how distance and speed readings from the HCSR04 ultrasonic sensors were collected in real-time to alert the rider to potential obstructions and approaching cars.

3.1 Sensor Data Acquisition

3.1.1 Distance Readings

The sensors were mounted on the back and front of the scooter to maximize coverage of the surrounding environment. They were securely mounted to reduce vibrations. The distance readings were acquired as described in Listing 2.1.

3.1.2 Speed Readings

In addition to capturing and analyzing distance readings to ascertain the proximity of potential hazards, it was equally necessary to measure and evaluate speed readings relative to the detected approaching vehicles, thereby providing a comprehensive understanding of the dynamic traffic environment surrounding the scooter. To capture the speed measurements, two distance

measurements were taken within a fixed time interval. The methodology involves determining the rate of change in distance over time as an object moves closer to or farther away from the sensor. When measuring an object's speed using an ultrasonic sensor, it's critical to consider the relative motion between the sensor and the object. As the object moves closer to the sensor, the distance measured decreases, and vice versa. Tracking these changes in distance over time allows us to derive speed measurements.

Let Position A and Position B represent the distances of the object from the sensor when measured at Time Position A and Time Position B, respectively. To capture speed, it is essential to measure the time between successive distance readings.

From this, we can derive the change in distance Δd as;

$$\Delta d = PositionB - PositionA, \tag{3.1}$$

whereas the corresponding time interval Δt can be derives as;

$$\Delta t = TimeatPositionB - TimeatPositionA. \tag{3.2}$$

Once we have the change in distance Δd and the corresponding time interval Δt, we can calculate the speed V of the detected approaching vehicle using the formula:

$$V = \frac{\Delta d}{\Delta t} \tag{3.3}$$

By capturing multiple distance readings at regular intervals, we can track the object's movement and calculate its relative speed over time. The following code implements the aforementioned algorithm to capture and calculate distance measurements.

```

void loop() {
  // Send a pulse to the Trig pin
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);

  // Measure the time it takes for the pulse to return
  long duration = pulseIn(echoPin, HIGH);

```

```
  // Calculate the distance in centimeters (speed of sound is about 343 meters per
    second)
  int distance = duration * 0.0343 / 2;

  // Calculate speed based on previous distance measurement (you may need to adjust
    the interval)
  static unsigned long prevTime = 0;
  static int prevDistance = distance;
  unsigned long currentTime = millis();
  float timeInterval = (currentTime - prevTime) / 1000.0;  // Convert to seconds

  // Calculate the speed in kilometers/hr
  float speed = abs((distance - prevDistance) / timeInterval) * 0.036 ;

  // Update previous values for the next iteration
  prevTime = currentTime;
  prevDistance = distance;

  // Print the distance and speed
  Serial.print("Distance: ");
  Serial.print(distance);
  Serial.print(" cm, Speed: ");
  Serial.print(speed);
  Serial.println(" km/h");

  // delay
  delay(500);
```

Listing 3.1: speed measurements implementation

The initial distance readings are in centimeters. A unit conversion is required to adhere to the standard unit of traffic speed, kilometers per hour (km/h). The ultrasonic sensor measures the time it takes for the sound wave to travel to an object and back, and this duration is used to calculate the distance in centimeters.

```
long duration = pulseIn(echoPin, HIGH);
int distance = duration * 0.0343 / 2;
```

Listing 3.2: initial speed measurements in centimeters

3.1.3 Real-Time Data Transmission

This thesis aims to achieve real-time data processing by allowing instantaneous ingestion and analysis of incoming data points and ensuring a continuous data stream that will give real-time information to the VRU while utilizing the e-scooter bike. Real-time data transmission and processing were handled by setting up a client/server network utilizing the TCP protocol. The Arduino MKR WiFi 1010 module captures and processes the sensor data locally by calculating the distance and corresponding speed readings illustrated in Listing 2.1. The WiFiNINA library was first installed to achieve data transmission over WiFi. The library enables network connection, both locally and over the Internet. With this library, it is possible to instantiate Servers and clients and send/receive UDP packets through WiFi. The board can establish a connection with either an unsecured or secured network, which can be encrypted using either WEP or WPA protocols. IP addresses can be assigned either statically or using DHCP, and the library can handle DNS. The general code for capturing and sending data to the server on the client side is as follows:

```
#include <WiFiNINA.h>

// WiFi credentials
char ssid[] = "OnePlus 8 Pro";
char pass[] = "X197_h20";
const int trigPin = 13;   // HC-SR04 Trig pin connected to D2
const int echoPin = 12;   // HC-SR04 Echo pin connected to D3
const float speedOfSound = 0.0343; // Speed of sound in cm/microsecond

// Server details
IPAddress serverIP(192, 168, 12, 118); // Change this to the IP address of your
    ESP32 board
const int serverPort = 80;

WiFiClient client;

void setup() {
  Serial.begin(9600);

  // Connect to WiFi network
  while (WiFi.begin(ssid, pass) != WL_CONNECTED) {
    delay(1000);
    Serial.println("Connecting to WiFi...");
```

```
  }

  Serial.println("Connected to WiFi");
}
```

Listing 3.3: Code for client

We start by including the necessary library and defining WiFi credentials with the SSID and password. The code also sets up pin assignments for an HC-SR04 ultrasonic sensor, with `trigPin`and `echoPin` connected to digital pins, . Additionally, it specifies the IP address and port (80) of a server the microcontroller will communicate with, using a `WiFiClient` object. In the setup function, the code initializes serial communication at a baud rate of 9600, and then attempts to connect to the WiFi network. It continuously checks the connection status every second until the connection is successful. This setup ensures the board is connected to the WiFi network before any further operations, such as server communication, are performed.

On the server side, the ESP32 module was utilized. to achieve a WiFi connection with the module, the Arduino WiFi library was installed. The general code for receiving and sending data to a more powerful processing unit is as follows:

```
#include <WiFi.h>

const char *ssid = "OnePlus 8 Pro";
const char *password = "X197_h20";
WiFiServer server(80);

void setup() {
  Serial.begin(115200);
  delay(500);

  // Connect to Wi-Fi
 WiFi.begin(ssid, password);
  Serial.println("Connecting to WiFi...");
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.println("\nWiFi connected");

  // Print ESP32 IP address
  //Serial.println("");
```

```
  Serial.print("Connected to WiFi network with IP address: ");
  Serial.println(WiFi.localIP());

  // Start server
  server.begin();
  Serial.println("Server started");
}

void loop() {  // Check if a client has connected
  WiFiClient client = server.available();
  if (client) {
    while (client.connected()) {
      if (client.available()) {
        String data = client.readStringUntil('\n');
        Serial.println(data);
      }
    }
  }
}
```

Listing 3.4: Code for server

The code facilitates connection to a Wi-Fi network and starting a basic web server. Initially, it defines the Wi-Fi credentials and initializes a server object `server on port 80 using `WiFiServer` which is the default port for HTTP communication. Although the code doesn't explicitly differentiate between HTTP methods like GET, POST, etc., it does implicitly handle the reception of HTTP requests in a basic manner In the `setup` function, after setting up serial communication for debugging, it attempts to connect to the specified Wi-Fi network until successful, upon which it prints the ESP32's assigned IP address. Once connected, it starts the server and confirms its initialization. In the `loop` function, it checks for incoming client connections using `server.available()`. Upon receiving a connection, it reads any incoming data from the client until it disconnects, printing the data to the Serial Monitor. This setup effectively turns the ESP32 into a simple web server that can handle basic communication with clients over Wi-Fi.

3.2 Road Segments

We gathered the obtained traffic data in the vicinity of Padova City, specifically at key arterial roads, intersections, and various urban and suburban zones. By focusing on these strategic

areas, we aimed to capture a wide range of traffic conditions and patterns.

Various road segments were considered based on criteria that influence traffic incidents and risk levels. The following road segments were selected to comprehensively understand traffic dynamics and the potential risks VRUs may face. Each chosen segment represents a distinct traffic environment where interactions between scooters, vehicles, and pedestrians vary significantly, impacting safety outcomes. By analyzing the data collected from these segments, this study seeks to identify high-risk areas prone to traffic incidents and evaluate factors contributing to safer road environments for VRUs. The following road segments illustrate diverse scenarios crucial for informing strategies to lessen risks and improve situational awareness for scooter riders.

3.2.1 Urban street with multiple lanes

This road segment has a dense traffic flow and a diverse mix of vehicles, including cars, buses, and trucks. It is a crucial segment connecting major residential, commercial, and recreational areas within the city. The street is characterized by multiple lanes in each direction, traffic lights at intersections, pedestrian crossings, and varying vehicle speeds.

Figure 3.1: Busy Urban street

The street experiences constant traffic throughout the day, with peak congestion during morning and evening rush hours. Vehicles frequently change lanes to navigate through the dense urban environment, posing challenges for scooter riders who must maneuver safely in

faster-moving traffic. Intersections along this street are critical points where vehicles converge from multiple directions, increasing the likelihood of collisions and close encounters with VRUs.

3.2.2 Residential Street

Figure 3.2: Residential street

This road segment is situated mostly in quiet neighborhoods characterized by single-family homes, apartment buildings, parks, and local shops. This type of street typically experiences moderate traffic flow with lower vehicle speeds. It serves as a connector between residential zones and main roads, making it an essential part of daily commuting for residents, This residential street comprises a single or dual-lane configuration with designated speed limits, often accompanied by traffic calming measures such as speed bumps, stop signs and pedestrian crossings. The street sees a mix of vehicular traffic, including cars, bicycles, and scooters, with peak traffic periods coinciding with school start and end times, as well as morning and evening rush hours.

3.2.3 Commercial Street Area

Figure 3.3: Commercial Street

This road segment features high pedestrian activity, bus stops, and dense vehicular traffic. This area serves as a hub for shopping, dining, banking and public transportation, attracting a diverse mix of commuters, shoppers, and tourists throughout the day. The street layout typically includes wide sidewalks, designated bus lanes, and multiple pedestrian crossings.

The traffic dynamics in this road segment are characterized by private vehicles navigating through congested streets, a mix of slow-moving buses, trams, delivery trucks, and taxis. Pedestrians frequently cross at designated crossings and intersections, often near bus stops and storefronts. This environment poses significant challenges for scooter riders due to the unpredictable movements of pedestrians and vehicles.

3.2.4 Suburban Road with School Zones

This road segment connectsresidential areas to local schools and community facilities. These roads typically exhibit varying traffic dynamics influenced by school hours, residential traffic patterns, and local events. They are characterized by designated speed limits, school crossings, and occasional traffic calming measures to ensure safety for pedestrians and school children.

Figure 3.4: Suburban areas with school zones

The traffic dynamics of this road segment features increased traffic from parents dropping off or picking up children During peak hours, school buses, and commuters passing through residential neighborhoods. The traffic flow can be moderate to heavy, with fluctuations throughout the day depending on school schedules and community activities. The presence of school zones necessitates adherence to reduced speed limits and heightened awareness of pedestrian activity.

3.2.5 Industrial Street Area

This road segment is characterized by warehouses, manufacturing facilities, and commercial loading docks. These areas are typically located on the outskirts of urban centers or in designated industrial zones, away from residential and commercial districts. The layout features wide roads, expansive parking lots, and specialized infrastructure for heavy vehicles and equipment. The Traffic dynamics in industrial areas are influenced by the operations of large trucks, delivery vehicles, and machinery used for transporting goods. The movement of vehicles is often dictated by scheduled deliveries and pickups, leading to fluctuating traffic volumes throughout the day. Speed limits may be higher compared to residential or school zone areas, reflecting the need for logistical operations.

Figure 3.5: Industrial Area with Loading Docks

3.3 Data Representation and Formatting

The output from the serial monitor had the data being represented in the following way:

========= **PuTTY Log** =========
Distance: 206 cm, Speed: 10.83 km/h
Distance: 174 cm, Speed: 10.19 km/h
Distance: 171 cm, Speed: 0.97 km/h
Distance: 204 cm, Speed: 10.42 km/h
Distance: 203 cm, Speed: 0.32 km/h
Distance: 204 cm, Speed: 0.32 km/h

Table 3.1: sample serial monitor output

To effectively analyze the readings from an ultrasonic sensor, which records distance and speed, a Python script was utilized to extract the numerical values and clean the data. The script uses the `pandas` library for data manipulation. Initially, the script reads the data file line by line, employing regular expressions to extract the numeric values from each string entry. the `re` module's `search` function identifies patterns corresponding to "Distance" and "Speed" values. Once extracted, these values are stored in separate lists, which are then combined into a `pandas` data frame for structured analysis. The data frame is subsequently used to generate visual plots: one subplot for distance readings and another for speed readings, both plotted against time.

The pseudocode algorithm used to implement this transformation is as follows:

```
BEGIN
    // Step 1: Initialize variables
    distances ← empty list
    speeds ← empty list

    // Step 2: Open the data file
    file ← open('data.txt', 'r')

    // Step 3: Read the file line by line
    FOR each line IN file
        // Use regular expressions to find the numeric values
        distance_match ← regex_search('Distance:\s*(\d+)\s*cm', line)
        speed_match ← regex_search('Speed:\s*([\d\.]+)\s*km/h', line)

        IF distance_match IS NOT NULL AND speed_match IS NOT NULL THEN
            // Extract and convert the numeric values
            distance ← integer(distance_match.group(1))
            speed ← float(speed_match.group(1))

            // Append the values to the lists
            distances.append(distance)
            speeds.append(speed)
        END IF
    END FOR
```

Listing 3.5: Data cleaning pseudocode

The resulting output after cleaning the data and applying regular expressions had the following structure;

Time (s)	Distance (cm)	Speed (cm/s)
0	300	50
1	300	40
2	270	40
3	250	30
4	280	45
...	310	25

Table 3.2: Sample Data Points.

3.3.1 EXPERIMENTAL DATASET

A short-run dataset experiment was conducted on a random road segment to assess the functionality and reliability of the ultrasonic sensor in real-world conditions. The primary goal was to collect distance and speed readings to analyze the sensor's output consistency and understand its operational characteristics. The chosen road segment for the short-run experiment was selected to reflect typical traffic conditions, which included minor delays and vehicles traveling at an average speed. This ensured that the experiment results would be representative of normal traffic flow. Obtaining speed data in non-controlled scenarios, such as real road conditions, presented challenges. Factors like varying traffic densities, unpredictable driver behaviors, and environmental conditions can influence vehicle speeds, making data collection more complex.

In attempting to mitigate the challenges of obtaining real-world speed data, this study leverages TomTom [19], a leading provider of traffic index and mapping technologies. TomTom aggregates global data on travel time, fuel costs, and CO_2 emissions. This data aids city planners, policymakers, and drivers in predicting traffic patterns, identify congestion hotspots, and optimizing urban mobility strategies [20]. By utilizing TomTom's real-time traffic information, which accumulates trillions of data points annually, this research aims to validate the accuracy of sensor-based vehicle speed detection on e-scooters. This comparison provides insights into improving sensor technology's reliability in urban environments.

Our study leveraged TomTom's traffic report on Padova for real-time insights [21]. The traffic data was collected using ultrasonic sensors deployed on a well-surveilled highway during weekday traffic hours. The data collection period was from approximately 11 am to 12 pm TomTom's traffic report for that period indicated that the average traffic speed on the highway

ranged between 55 km/h and 60 km/h.

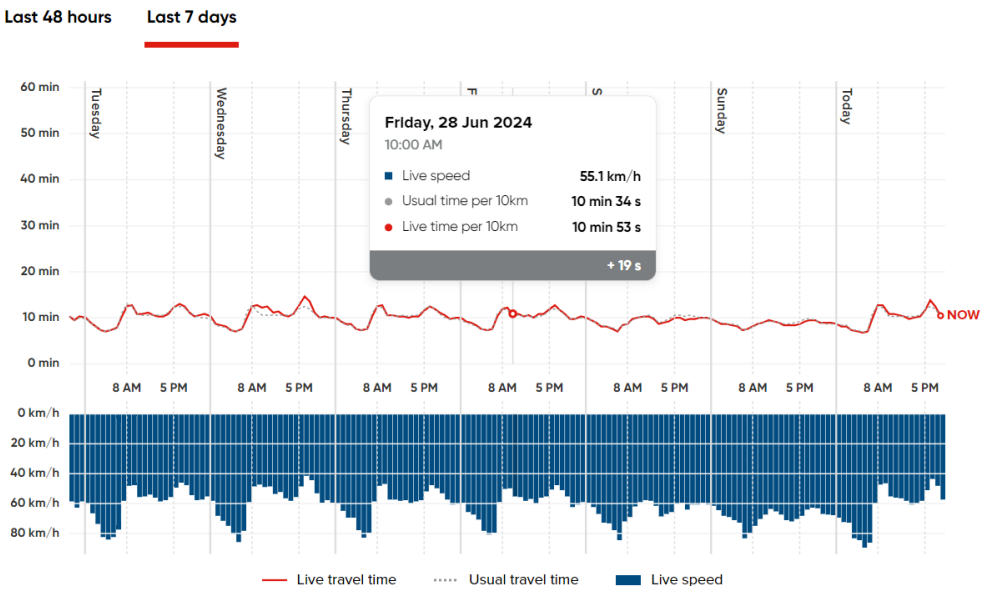


Figure 3.6: Divergent chart of hourly speed and travel time

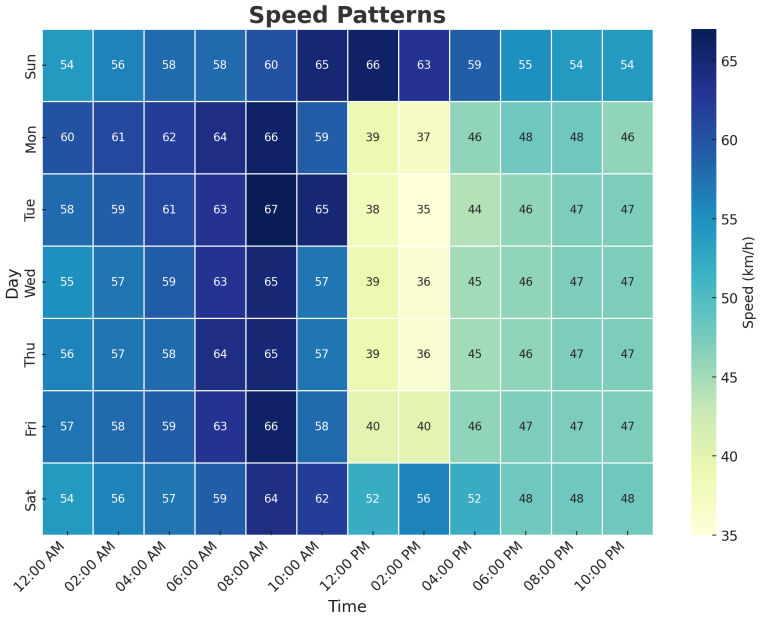


Figure 3.7: Padova speed heat map

The highway selected for the study is regularly monitored and accurately recorded using

traffic cameras and GPS tracking which would contribute to accurate and reliable traffic data provided by TomTom's traffic index, this time slot was selected to capture typical mid-morning traffic conditions, aiming to provide a representative sample of vehicle speeds during this part of the day.

Another significant challenge was estimating the vehicle flow on the highway at targeted time frames. It is essential to understand the traffic flow principles. The principles involve the study of the movement of vehicles on roadways, focusing on the relationships between traffic flow (vehicles per hour), traffic density (vehicles per kilometer), and vehicle speed (kilometers per hour). Understanding these relationships, often depicted in the fundamental diagram of traffic flow, is crucial for estimating and managing traffic conditions [22], [23].

Traffic density refers to the number of vehicles occupying a specific length of road at a given time. It varies based on road conditions and traffic levels. Direct measurement in the field is challenging, as it requires a vantage point for photographing, videotaping, or observing significant lengths of highway [24]. Traffic simulation and modeling studies have offered insights into traffic densities under various scenarios, estimating a moderate traffic density at around 20–30 vehicles per kilometer [25]. The traffic flow rate is the equivalent hourly rate at which vehicles pass a point on a highway for 1 hour expressed in vehicles per hour(vph). We can calculate the flow by Using the average speed and the estimated density. For simplicity, we used the average speed of 57.5 km/h (midpoint of 55 km/h and 60 km/h). The flow rate can be estimated using the following relationship:

$$q = k \cdot v, \tag{3.4}$$

Where:

q = Traffic flow rate (vehicles per hour)

k = Traffic density (vehicles per kilometer)

v = Average speed of vehicles (kilometers per hour).

From this, the flow rate can be calculated using the average speed and the estimated density, as described in (3.4). For this calculation, a moderate traffic density of 25 vehicles per kilometer is assumed hence:

$$q = 25 \text{ vehicles/km} \times 57.5 \text{ km/h} = 1437.5 \text{ vph} \tag{3.5}$$

For the experimental runs, the time frame interval was set to approximately five minutes. Consequently, estimating the number of vehicles passing within this interval was necessary.

To achieve this estimation, the hourly vehicle rate was divided by 12, reflecting that one hour comprises twelve five-minute intervals. This method allowed us to determine the number of vehicles passing in five minutes, as represented by: vehicles in a 5-minute interval = $\frac{q}{12} \approx 120$ vehicles. Based on this estimation, under moderate traffic conditions with an average speed of 55 km/h to 60 km/h, approximately 120 vehicles can pass a certain point in one direction within a five-minute interval during mid-morning traffic on a weekday.

The subsequent procedure involved generating synthetic data for approximately 120 vehicles to estimate traffic flow rates. This synthetic dataset was utilized for comparative analysis against empirical measurements obtained from the ultrasonic sensor. Vehicle speed data were simulated using a normal distribution centered around an average speed of 57.5 km/h, with a standard deviation of 5 km/h to introduce variability. The simulation was conducted over a predefined duration of 5 minutes, accommodating 120 vehicles. Timestamps were dynamically generated with non-fixed intervals using an exponential distribution, reflecting realistic variations in vehicle arrival times. Each timestamp was computed by adding a randomly generated time interval to the previous timestamp, starting from an initial fixed time. This approach mirrors real-world traffic scenarios where vehicles arrive at irregular intervals. The simulated speeds and timestamps were aggregated into a pandas DataFrame and saved to a CSV file for subsequent analysis. Visualizations included a time series plot showing the variation in vehicle speeds over time and a histogram illustrating the distribution of speeds, offering insights into the variability and distribution of simulated vehicle speeds under the specified conditions.

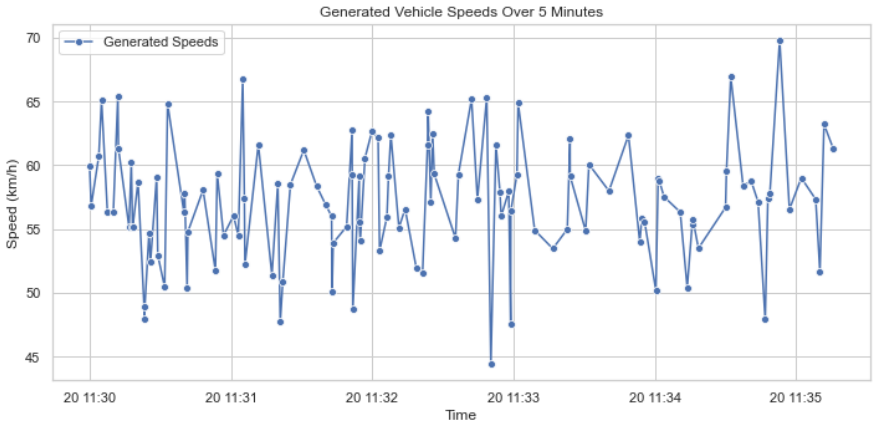


Figure 3.8: speeds time series over time

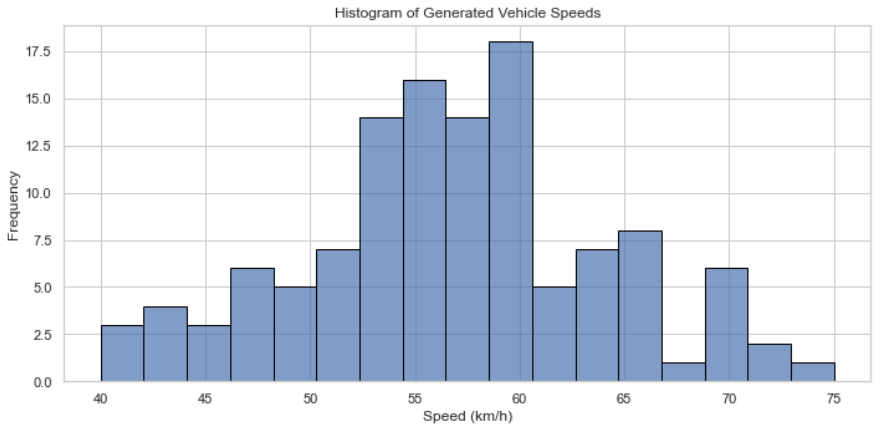


Figure 3.9: Histogram of speed distribution

A stationary ultrasonic sensor was mounted on the back of an e-scooter bike to monitor approaching vehicles. Readings were taken at regular 0.5-second intervals, recording both the distance in centimeters and the speed in cm/s, which was then converted to km/h after processing the measurements. Each data point measured the distance between the sensor and the approaching vehicle, and the approaching vehicle's speed was calculated based on consecutive distance readings.

The experimental duration spanned approximately five minutes, capturing a sufficient number of data points and ensuring variability in vehicle speeds and distances. The collected dataset comprised 500 data points, each representing a snapshot of the distance and speed of an approaching vehicle. This evaluation included assessing the sensor's responsiveness to changes in vehicle speed and its accuracy in calculating speed based on distance intervals.

We generated plots to visualize the data, including histograms and scatter plots for distance and speed measurements. The findings are illustrated below.

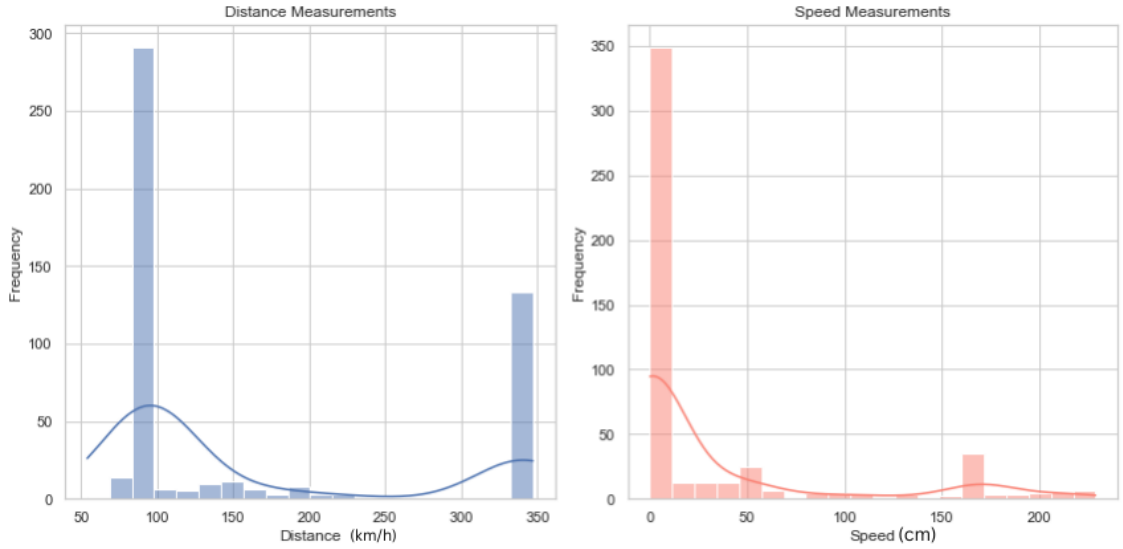


Figure 3.10: Speed and Distance Histograms

The histogram plot illustrates the distribution of distance measurements, revealing a distinct bimodal pattern characterized by prominent peaks approximately at 100 and 340 units. Regarding speed, the histogram depicts a skewed distribution concentrated towards the lower values. This skewness indicates that most recorded speeds are relatively low, while higher speeds are less common and exhibit a rapid frequency decline towards the distribution's tail end.

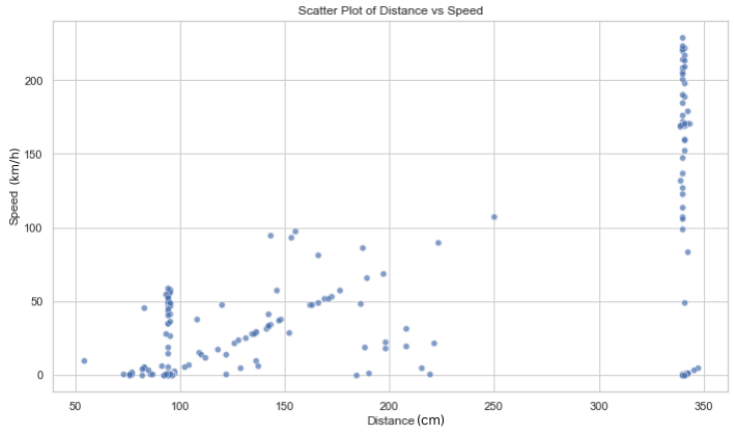


Figure 3.11: scatter plot of distance and speed

We were also able to see a non-linear relationship between the speed and distance variables from the scatter plot, necessitating additional analysis to understand the underlying patterns.

in the context of this study when calculating speed as the ratio of change in distance to time interval, if both the current and previous readings are zero, the change in distance is zero, resulting in a calculated speed of zero. This typically indicates that no object is detected within the sensor's range. Therefore, zero or near-zero readings suggest the absence of approaching vehicles, as the lack of distance data translates to no measurable change in distance, thus yielding a speed calculation of zero.

We observed that the ultrasonic sensor readings become inconsistent when the distance exceeds 100 to 200 cm. This behavior results in miscalculated distances that affect the speed values. This indicates that the sensor has an accuracy limitation in reliably detecting weaker signal reflections beyond 200 cm, where the accuracy starts fading. Based on this insight, we concentrated on speeds greater than 5 km/h and distances under 200 cm to improve data interpretation and focus on actual vehicle detections.

We had the following findings after filtering the data:

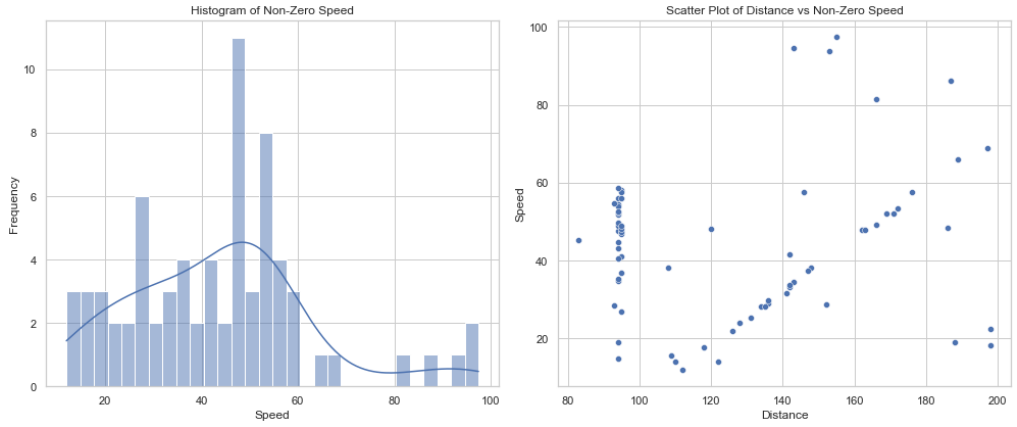


Figure 3.12: Histogram and scatter plot for non-zero speeds

A summary of the statistics of the short-run experimental data is as follows:

Distance (cm)	
Mean	126.10
Standard deviation	34.31
Minimum	83
25th Percentile	94
75th Percentile	149
Maximum	198

Table 3.3: Filtered Distance Data Points summary

Speed (km/h)	
Mean	43.64
Standard deviation	19.01
Minimum	12
25th Percentile	28.96
75th Percentile	52.77
Maximum	97.41

Table 3.4: Filtered Speed Data Points summary

After filtering the ultrasonic sensor readings, we gained clearer insights into the speed and distance measurements for detecting approaching vehicles in traffic. The speed data ranges from 12 to 60 km/h, distributed around the mean, indicating a normal distribution with slight skewness. Additionally, the distance measurements showed a positive skew, suggesting the sensor effectively detects vehicles within its operational range, with a tendency to capture more readings from vehicles farther away but still within detection limits.

3.3.2 Comparison

The next step involved the analysis of the sensor readings by comparing them with synthetic data generated based on TomTom traffic data index. This comparison was essential to validate the sensor's accuracy in capturing traffic speed. By analyzing the average speed values from both datasets, we aimed to assess the reliability and effectiveness of the sensor technology used in this study.

The datasets had different entries because of the filtering applied to remove noise, and therefore we only compared the average speed values from both datasets. The following plot visualizes the comparison:

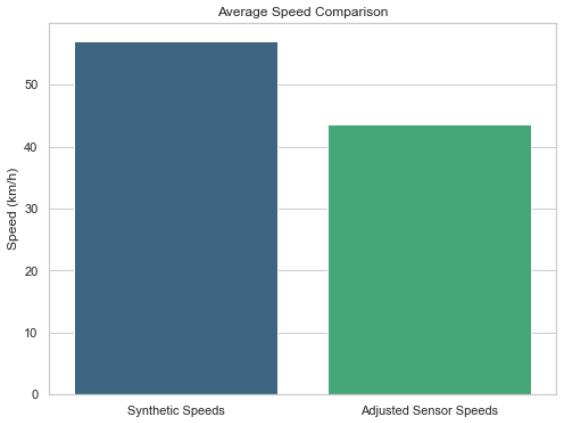


Figure 3.13: Average speed comparison

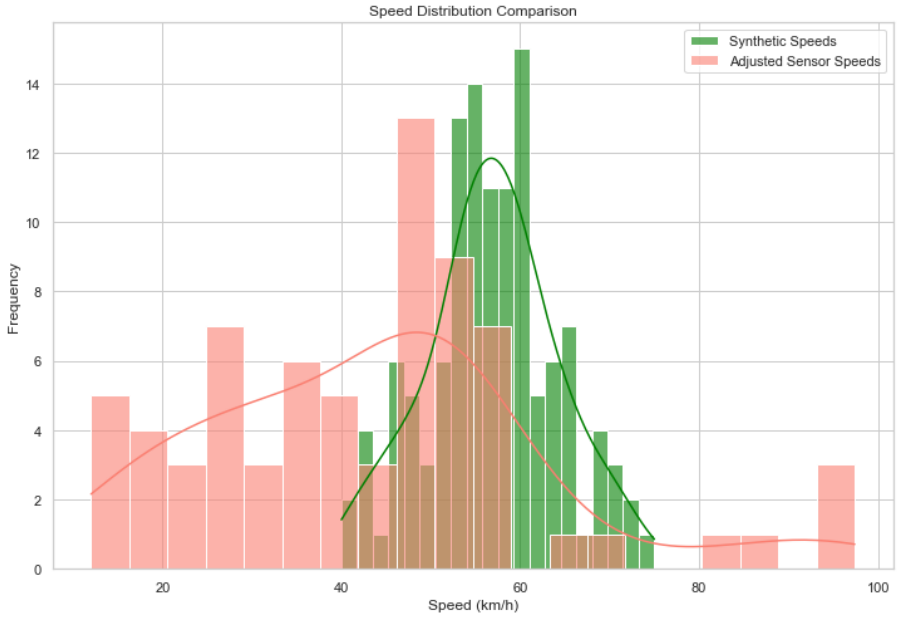


Figure 3.14: Speed distribution comparison

We calculated the absolute speed difference from both datasets and found it to be **13.46 km/h**. The results indicated that the ultrasonic sensor speeds are consistently lower than the synthetic speeds. This discrepancy could be due to the use of synthetic traffic instead of actual traffic data, leading to only approximate estimations. Additionally, the sensor's accuracy deteriorates with increased range, and environmental factors, such as air temperature, can affect the speed of the sensor's pulse [26].

It is important to note that the traffic speed readings were taken from a moving e-scooter, which could significantly affect the sensor accuracy due to the context of **relative motion**. Relative speed refers to the velocity of an object as observed from a moving reference point [27]. In this thesis, the direction of motion is considered the same for both objects. When two objects move in the same direction, each appears to move more slowly relative to the other [28]. To account for the relative motion, we applied some relative speed adjustments. If the scooter is moving at speed V_{scooter} and it detects a vehicle approaching from behind at speed V_{detected} relative to the scooter, the actual speed of the vehicle V_{actual} is:

$$V_{\text{actual}} = V_{\text{detected}} + V_{\text{scooter}} \tag{3.6}$$

We adjusted the sensor data by adding the scooter's speed to the detected values to approximate the true speed of the vehicles. The e-scooter had a maximum speed of 20 km/h, and the experimental speeds varied between 15 and 20 km/h. Therefore, we used the scooter's average speed (V_{scooter}) of 17.5 km/h.

The following plot visualizes the new adjusted values:

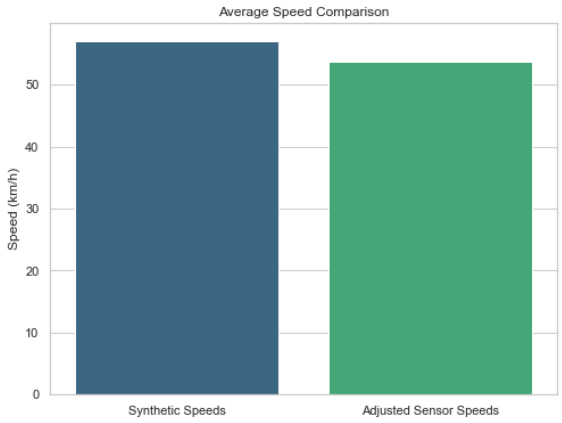


Figure 3.15: Average speed comparison

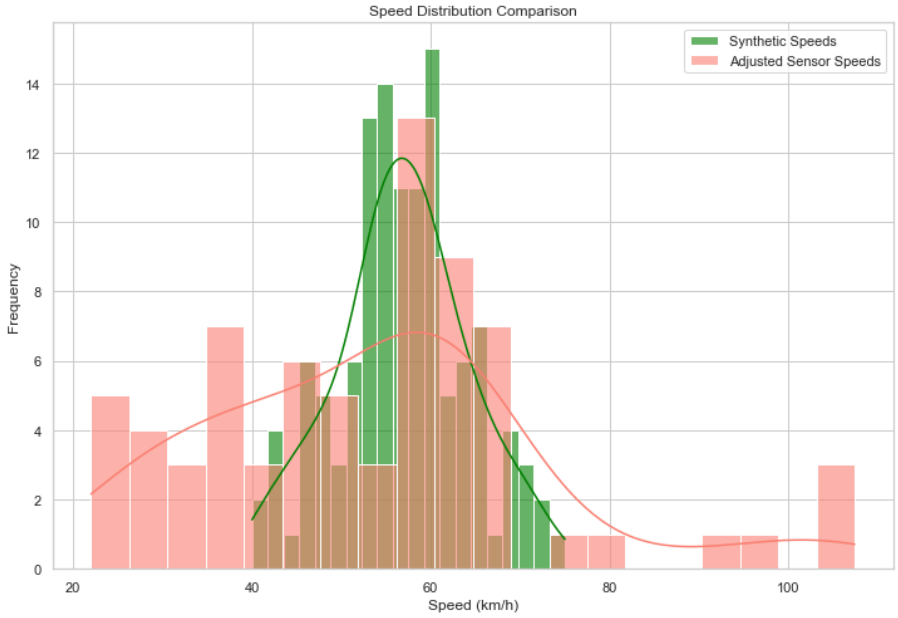


Figure 3.16: Speed distribution comparison

The new readings showed an adjusted speed difference of 3.46 km/h, significantly reducing the discrepancy and closely matching the true speed data from the TomTom traffic index. This indicates that accounting for the e-scooter's speed provided a more accurate representation of actual vehicle speeds, validating both the adjustment's effectiveness and the TomTom traffic index's authenticity.

The short-run dataset experiment provided valuable insights into the operational capabilities of the ultrasonic sensor in a real-world scenario. The experiment involved testing the sensor's ability to detect approaching vehicles and calculate their speeds using distance measurements. The findings suggest that the sensor can reliably perform these tasks, demonstrating its potential for use in traffic monitoring and management systems. Additionally, the experiment highlighted the sensor's limitations, which is crucial for understanding its operational boundaries. The accuracy of the sensor, especially when the data was adjusted for the e-scooter's speed, is closely aligned with the true speed data from the TomTom traffic index. This validates both the sensor's effectiveness and the accuracy of the TomTom traffic index.

3.3.3 Side Sensor Data

This section explains the data captured from the e-scooter's side sensors, specifically designed to detect objects approaching from the side on which the sensor was mounted. These sensors

ensure rider safety by identifying potential hazards and obstacles in the scooter's immediate vicinity.

The following outlines the basic statistics of the data:

Distance (cm)	
Mean	208
Standard deviation	104.91
Minimum	17
25th Percentile	129
75th Percentile	270
Maximum	348

Table 3.5: Side Distance Data Points summary

Speed (km/h)	
Mean	11.77
Standard deviation	4.17
Minimum	2
25th Percentile	10.8
75th Percentile	13.72
Maximum	28.91

Table 3.6: Side Speed Data Points summary

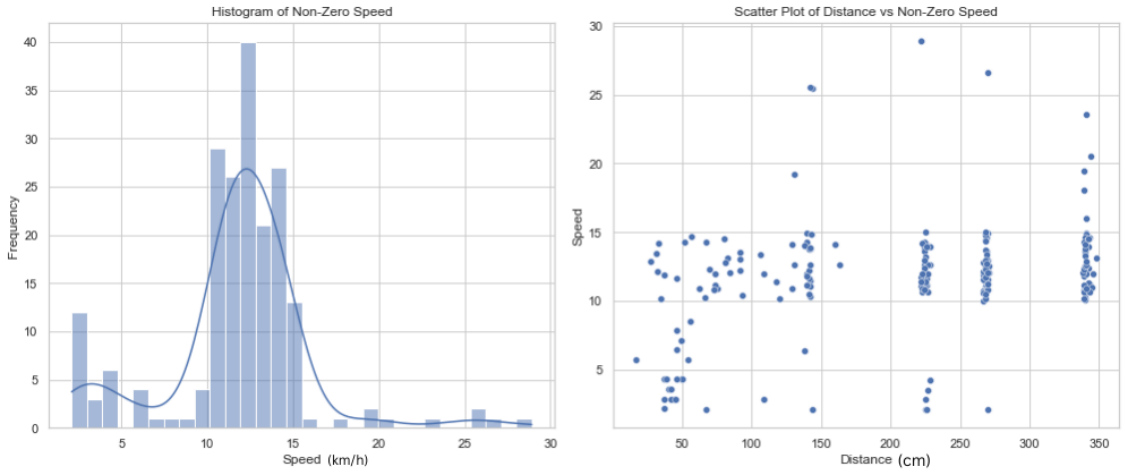


Figure 3.17: Speed and Distance Histograms

The scatter plot and histogram in this analysis shed light on the relationship between distance and speed in traffic monitored by an e-scooter. The scatter plot reveals no significant trend between the two variables, a common scenario in traffic where objects at similar distances exhibit varying speeds. For example, some objects near the e-scooter may move slowly, while others at the same distance might travel much faster.

The data primarily shows speed values ranging from 2 to 15 km/h, typical of objects either moving slowly or overtaking the e-scooter at moderate speeds. The histogram highlights a con-

centration of speeds below 15 km/h, reflecting common traffic conditions. Occasional speeds above 15 km/h likely indicate rare events or data noise from fast-moving objects.

4

Results and Discussion

4.1 Trends and Patterns of Road Segments

In this section, we present and analyze the results obtained from long-run experiments and data collection. The primary objective was to validate the reliability of the radar-based detection system implemented on e-scooters, focusing on key metrics such as distance, speed as well as variations. Additionally, we sought to identify patterns and trends in approaching vehicle behaviors across different road segments, ultimately aiming to enhance road safety for VRU.

Data were collected from various road segments known for their diverse traffic conditions. The selected segments included urban streets with multiple lanes, residential streets, commercial street areas, suburban streets with school zones, and industrial areas. The radar sensors on the e-scooters continuously monitored the distance and speed of approaching vehicles, capturing real-time data on their proximity and velocity.

The collected data were categorized based on the type of road segment and the time of day to understand the variations in traffic patterns and incident risks. We focused on two times of the day: mid-morning to afternoon hours and late afternoon to evening hours. Each segment's data was further analyzed to determine the frequency of vehicle approaches, as well as the corresponding speed and distance measurements. The datasets featured already cleaned data removing noise as well as relative speed adjustments based on the relative speed of approaching vehicles to give accurate results

4.1.1 Urban Road Segment Patterns

The datasets collected for this road segment contain distance and speed readings captured by the HC-SR04 sensor in the morning and evening hours. Here is a brief comparison based on their data:

Tables 4.1 to 4.4 represent evening data

Distance (cm)	
Mean	124
Standard deviation	124
Minimum	35
25th Percentile	94
75th Percentile	150
Maximum	198

Table 4.1: Urban Road Evening Distance summary

Speed (km/h)	
Mean	68.75
Standard deviation	30
Minimum	8
25th Percentile	47.2
75th Percentile	82.6
Maximum	170.7

Table 4.2: Urban Road Evening Speed summary

The following tables represent the morning data

Distance (cm)	
Mean	122
Standard deviation	34
Minimum	83
25th Percentile	94
75th Percentile	147
Maximum	198

Table 4.3: Urban Road Morning Distance summary

Speed (km/h)	
Mean	76.77
Standard deviation	31
Minimum	12
25th Percentile	54
75th Percentile	91
Maximum	205

Table 4.4: Urban Road Morning Speed summary

Next, we visualized the distribution of distance and speed for both datasets and we had the following observations

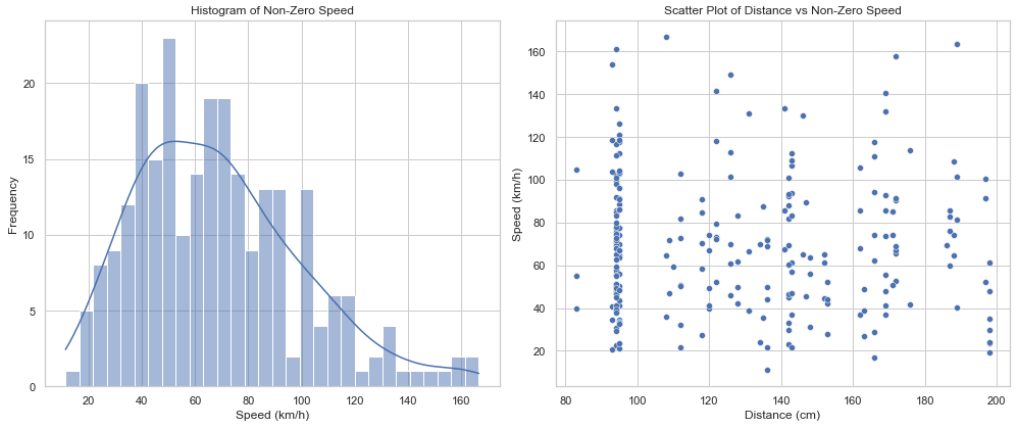


Figure 4.1: Urban road evening distribution

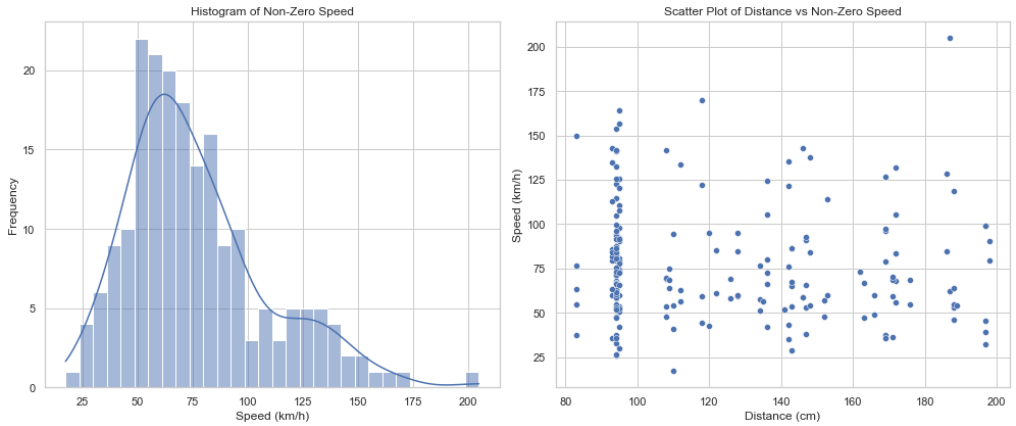


Figure 4.2: urban road morning distribution

The histograms for distance show a similar distribution pattern for both morning and evening. The speed histogram shows that the morning data has a broader distribution with higher maximum speeds, whereas the evening speeds are more concentrated around the mean. The data indicates that while the distances measured by the sensor are relatively consistent between morning and evening, the speeds are significantly higher and more variable in the morning compared to the evening. This could suggest that the VRU encounters more dynamic traffic conditions or faster-moving traffic during the morning hours compared to the evening, possibly due to higher traffic or more obstacles in the evening hours.

4.1.2 Residential Road Segment Patterns

A brief comparison of the initial road segment characterized by quiet neighborhoods, single-family homes, apartment buildings, parks, and local shops is as follows in tables 4.5 to 4.8:

Distance (cm)	
Mean	130
Standard deviation	34
Minimum	93
25th Percentile	95
75th Percentile	162
Maximum	198

Table 4.5: Residential Street Evening Distance summary

Speed (km/h)	
Mean	45.77
Standard deviation	22.2
Minimum	15.65
25th Percentile	28.03
75th Percentile	96.08
Maximum	225.28

Table 4.6: Residential street Evening speed summary

Distance (cm)	
Mean	129
Standard deviation	37
Minimum	83
25th Percentile	94
75th Percentile	103
Maximum	198

Table 4.7: Residential street morning Distance summary

Speed (km/h)	
Mean	51.12
Standard deviation	23.18
Minimum	13.48
25th Percentile	32.95
75th Percentile	62.43
Maximum	144

Table 4.8: Residential street morning speed summary

The distribution of distance and speed for both datasets had the following observations

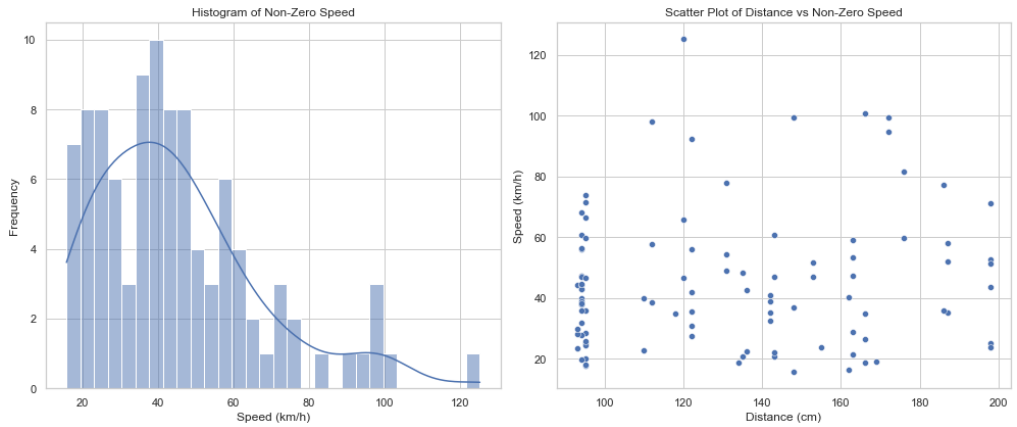


Figure 4.3: residential street road evening distribution

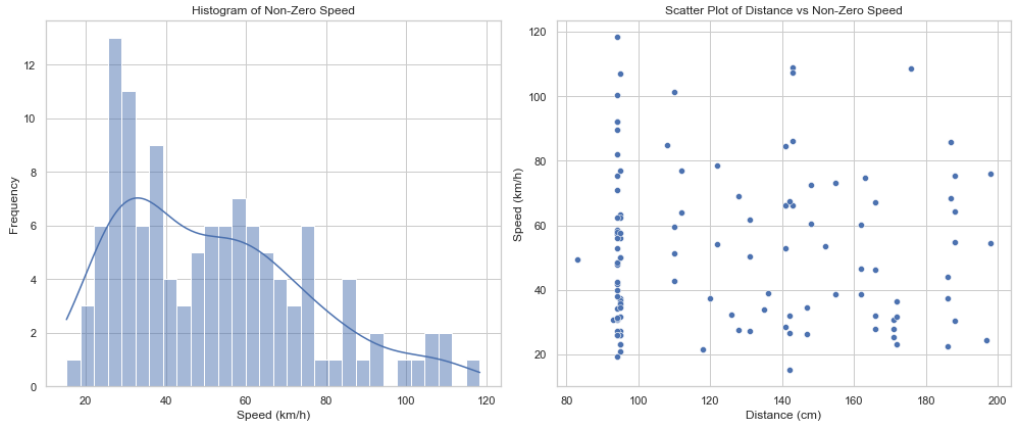


Figure 4.4: residential street road morning distribution

The histograms indicate that both the distance and speed readings have a similar distribution pattern in the morning and evening, although the morning readings show slightly higher variability, particularly in speed. The evening speeds are more concentrated around the lower to mid-range values. The distance measurements do not show significant differences between the two time periods. The data shows higher average speeds and greater variability in morning hours on a residential street, possibly due to drivers using it as a shortcut or through route. In the evening, the average speed drops, suggesting more consistent traffic flow, possibly due to increased congestion as residents return home. This pattern of higher, more variable speeds in the morning suggests traffic and safety risks, while slower, more consistent traffic suggests higher residential use and potential congestion.

4.1.3 Commercial Street Patterns

This section outlines a comparison of the road segment, characterized by a busy commercial street, and the understanding of traffic patterns and speed distributions on such streets in tables 4.9 to 4.12.

Distance (cm)	
Mean	128
Standard deviation	31
Minimum	66
25th Percentile	108
75th Percentile	150
Maximum	199

Table 4.9: Commercial Street Evening Distance summary

Speed (km/h)	
Mean	60
Standard deviation	12
Minimum	17
25th Percentile	50
75th Percentile	59
Maximum	103

Table 4.10: Commercial Street Evening speed summary

Distance (cm)	
Mean	246
Standard deviation	31
Minimum	48
25th Percentile	96
75th Percentile	139
Maximum	196

Table 4.11: Commercial Street Morning Distance summary

Speed (km/h)	
Mean	62.05
Standard deviation	12.44
Minimum	40.49
25th Percentile	52.07
75th Percentile	69.53
Maximum	106

Table 4.12: Commercial Street Morning speed summary

The trend and distribution of the distance and speed values are visualized as follows

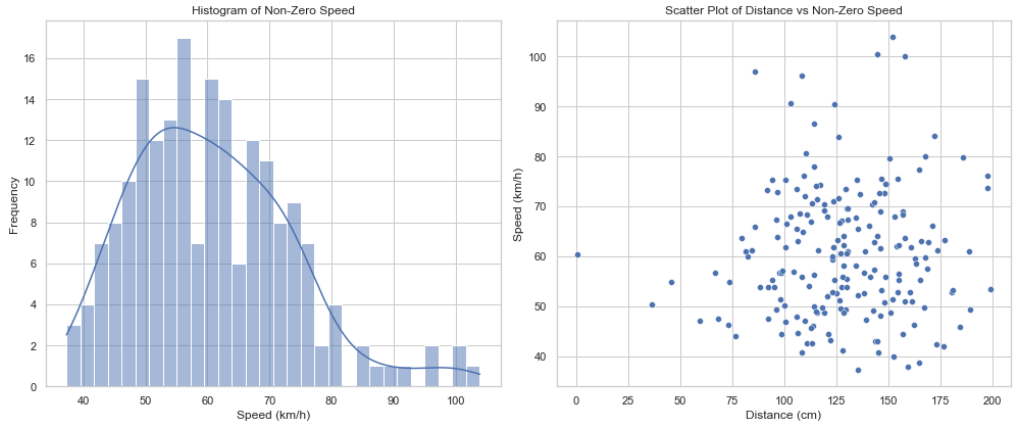


Figure 4.5: commercial street evening distribution

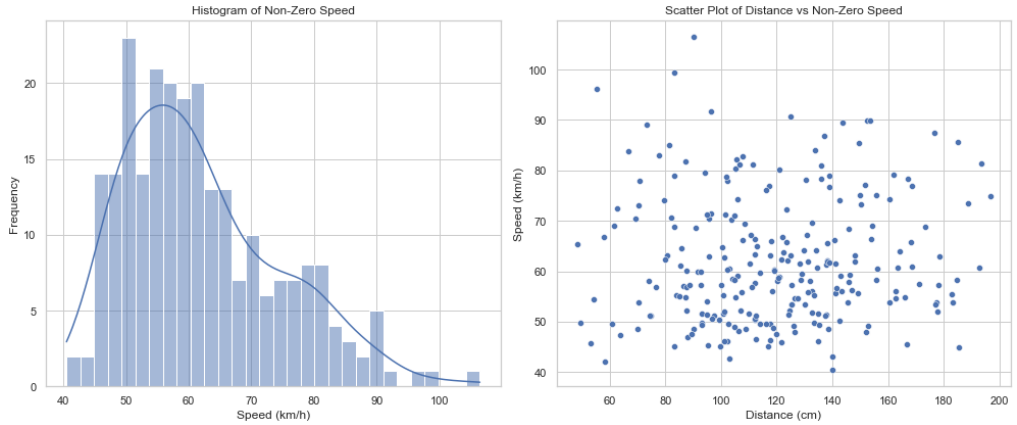


Figure 4.6: commercial street morning distribution

The study analyzed vehicle speed patterns for this road segment, revealing distinct trends in morning and evening traffic. In the morning, average vehicle speeds are slightly higher, with speeds clustering around the median, indicating a consistent and uniform traffic flow. This consistency is likely due to routine commuting patterns where vehicles maintain stable speeds. Morning traffic shows more consistent distances and speeds, reflecting regular commuting behavior with steady driving patterns.

In the evening, average speeds decrease, with a similar distribution spread but slightly lower averages. This decrease is attributed to evening rush hour congestion, leading to slower and more variable speeds due to frequent stops and starts. Evening traffic exhibits greater distance variability and lower average speeds, indicative of typical rush hour conditions with more stop-and-go situations. This increased variability suggests that vehicles are often further apart due to

congestion and delays. Overall, morning traffic is more predictable and steady, while evening traffic experiences increased variability and congestion, characteristic of urban traffic dynamics during peak hours.

4.1.4 Suburban Road with School Zones Patterns

outlined below are the analysis and comparison of morning and evening data summary statistics in tables 4.13 to 4.16.

Distance (cm)	
Mean	126.69
Standard deviation	31.58
Minimum	83
25th Percentile	94
75th Percentile	150
Maximum	198

Table 4.13: Suburban Road Evening Distance summary

Speed (km/h)	
Mean	42.11
Standard deviation	12
Minimum	11
25th Percentile	26
75th Percentile	54
Maximum	104

Table 4.14: Suburban Road Evening speed summary

Distance (cm)	
Mean	12.36
Standard deviation	31.85
Minimum	93
25th Percentile	94
75th Percentile	95
Maximum	196

Table 4.15: Suburban Road Morning Distance summary

Speed (km/h)	
Mean	38.77
Standard deviation	19
Minimum	10
25th Percentile	23
75th Percentile	45
Maximum	101

Table 4.16: Suburban Road Morning speed summary

Outlined are the visualizations to better understand the pattern and trends

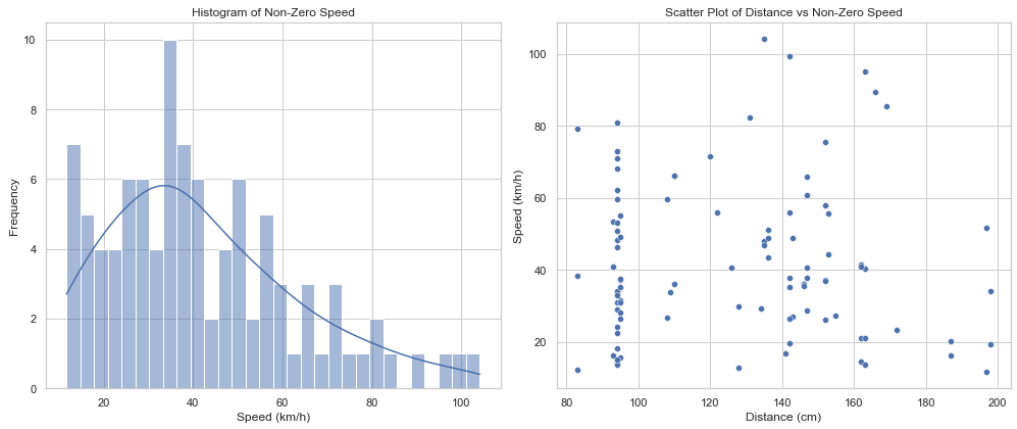


Figure 4.7: Suburban street evening distribution

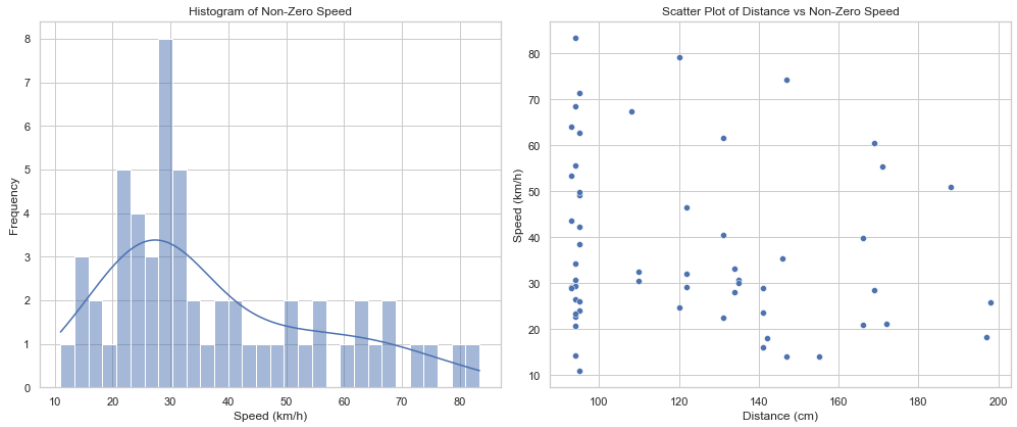


Figure 4.8: suburban street morning distribution

The analysis of distance and speed readings shows that vehicles tend to be detected at slightly greater distances in the evening. Additionally, there is higher variability in morning distances, indicating more variation in the distances at which vehicles are detected. Speed analysis shows that vehicles travel faster during the evening compared to the morning. The evening data also reveals increased speed variability and higher maximum speeds, indicating a greater tendency towards extreme speed values.

These observations point to a pattern of higher and more varied vehicle speeds during the evening, likely due to the evening rush hour when people are returning home from work or other activities. In contrast, morning traffic exhibits more consistent speeds and distances, possibly due to school zone regulations and morning routines. Evening hours experience higher

and more variable speeds, while morning traffic tends to be more consistent but still requires careful management by VRU.

4.1.5 Industrial Area Patterns

outlined below are the analysis and comparison of morning and evening data summary statistics of industrial street road segments in tables 4.16 to 4.19

Distance (cm)	
Mean	124.69
Standard deviation	31.5
Minimum	28
25th Percentile	107
75th Percentile	141
Maximum	196

Table 4.17: Evening Industrial street Distance summary

Speed (km/h)	
Mean	36.71
Standard deviation	18.6
Minimum	15
25th Percentile	29
75th Percentile	44
Maximum	75.36

Table 4.18: Evening Industrial street speed summary

Distance (cm)	
Mean	12.36
Standard deviation	31.85
Minimum	93
25th Percentile	94
75th Percentile	95
Maximum	196

Table 4.19: Morning Industrial Street Distance summary

Speed (km/h)	
Mean	38.77
Standard deviation	19
Minimum	10
25th Percentile	23
75th Percentile	45
Maximum	101

Table 4.20: Morning Industrial street speed summary

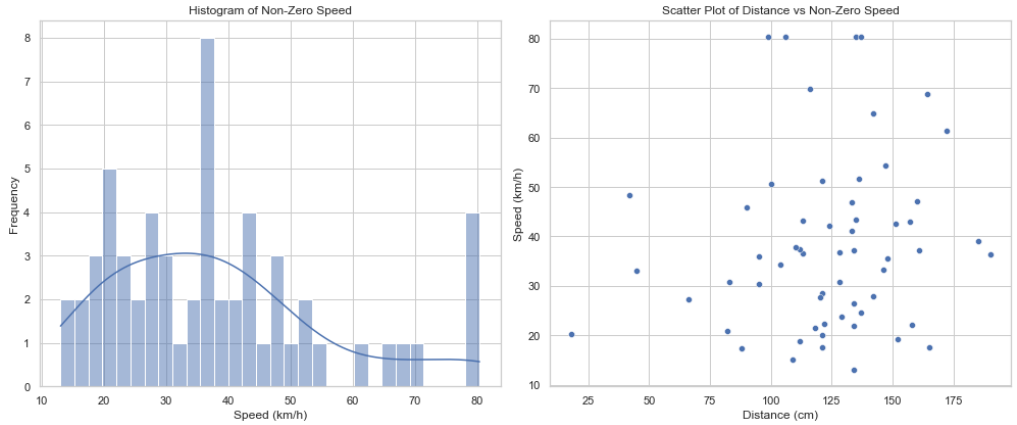


Figure 4.9: Industrial street evening distribution

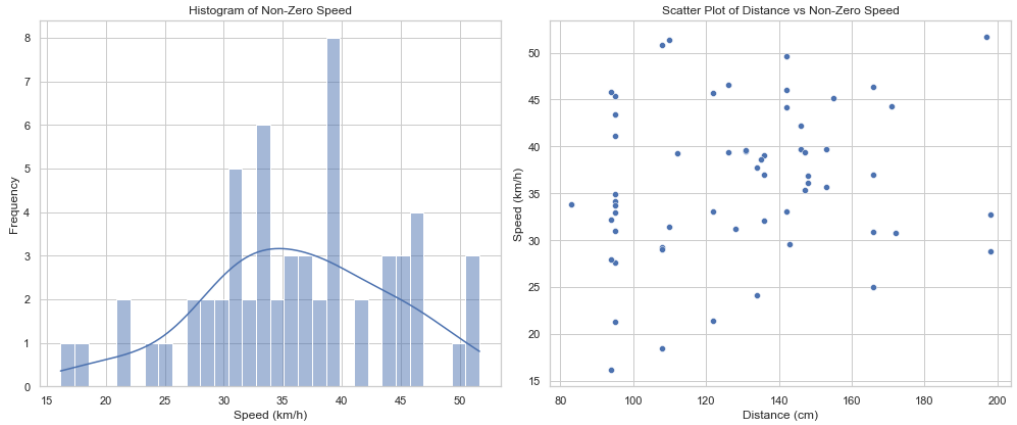


Figure 4.10: industrial street morning distribution

The average vehicle speed is slightly higher in the evening compared to the morning, but the speed measurements are close to identical in both periods. Given the industrial setting, it's expected to see lower speeds and potentially fewer vehicles moving at high speeds. To some extent, the data reflects this, especially in the morning, when speeds are more consistent and within a narrower range. The evening variability may be caused by shift changes, increased activity, or more diverse types of vehicles using the street during these hours, suggesting an increase in risk during these hours.

4.2 Discussion

The analysis of the collected data from various road segments: commercial, industrial, residential, suburban, and urban, reveals distinct patterns in the speed and distance of approaching vehicles. Box plots were created to visualize these differences effectively.

4.2.1 Evening Patterns

The analysis of the collected data from various road segments—commercial, industrial, residential, suburban, and urban—reveals distinct patterns in the speed and distance of approaching vehicles. Box plots were created to visualize these differences effectively.

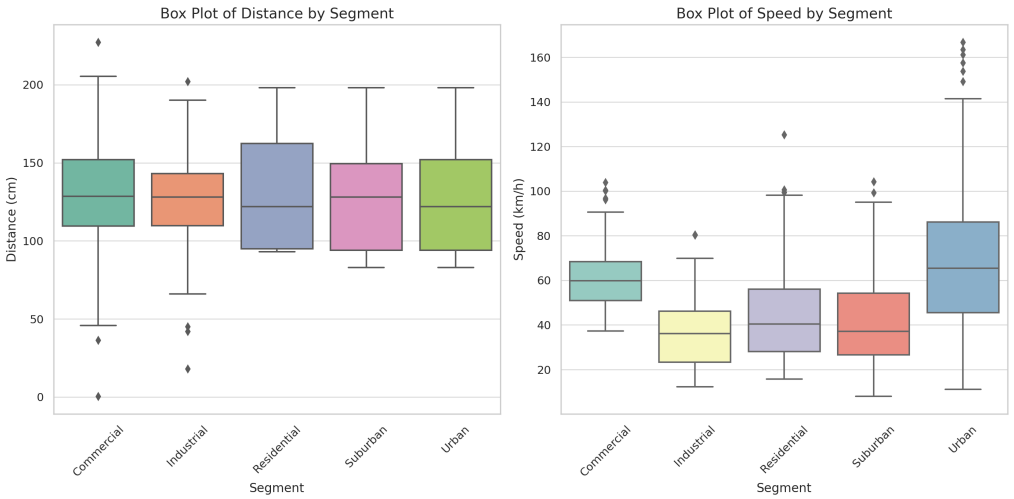


Figure 4.11: Box Plots by road segment in evening traffic

- **Distance by Segment**:

  Commercial Areas: The distance of approaching vehicles shows a wide range with several outliers. This variability could be attributed to the mixed traffic conditions typical of commercial areas, including both fast-moving and slow-moving vehicles, frequent stops for pedestrians, and varying road widths.

  Industrial Areas: Distances in industrial areas are relatively consistent with fewer outliers. The controlled environment, dominated by trucks and commercial vehicles with predictable routes and speeds, might explain this stability. Residential Areas: The data from residential areas show significant variability with a wide interquartile range. The presence of pedestrians, children, and strict speed limits likely contributes to this dispersion.

Suburban Areas: Suburban distances are moderately variable, indicating a blend of residential and commercial traffic characteristics. These areas often have longer stretches of road with fewer stops, allowing vehicles to maintain more consistent distances.

Urban Areas: Urban areas exhibit high variability with numerous outliers. This can be linked to dense traffic, frequent traffic signals, and diverse vehicle types, resulting in unpredictable vehicle distances.

- **Speed by Segment**:

  Commercial Areas: Vehicle speeds in commercial areas show high variability with several extreme values. The mixed-use nature of these roads, frequent stopping, and varying traffic flow contribute to the observed speed range.

  Industrial Areas: Speeds in industrial areas are relatively high and consistent, with a few outliers. The predominance of commercial vehicles adhering to regulated speeds and routes explains the observed trend.

  Residential Areas: Residential speeds are generally lower, with some variability. The strict enforcement of speed limits and the presence of pedestrians necessitate slower vehicle speeds.

  Suburban Areas: Speeds in suburban areas vary widely, reflecting the mixed traffic conditions. The blend of residential and commercial characteristics, along with longer road stretches, allows for higher speeds.

  Urban Areas: Urban speeds are highly variable, similar to the distance data. The dense traffic, frequent stops, and diverse vehicle types in urban environments lead to significant speed variations.

The box plots reveal notable differences in vehicle speeds and distances across various road segments, highlighting the traffic behavior. Commercial Areas exhibit the highest variability in both speed and distance, indicating a dynamic and unpredictable traffic environment. The frequent stopping and starting, coupled with a mix of vehicle types, contribute to this variability. Industrial Areas show more consistency, particularly in distance, which can be attributed to the controlled nature of these environments dominated by commercial vehicles. However, speeds are relatively high, reflecting the efficiency of these routes for commercial activities. Residential Areas demonstrate significant variability in both metrics, underscoring the impact of speed regulations and the presence of pedestrians. The lower speeds are a direct result of these safety measures. Suburban Areas offer a middle ground with moderate variability, reflecting their mixed-use nature. These areas combine residential and commercial traffic characteristics, leading to diverse speed and distance patterns. Urban Areas present high variability in both speed and distance, influenced by dense traffic conditions, frequent traffic signals, and diverse

vehicle types. The complex urban environment necessitates frequent speed adjustments and results in unpredictable vehicle behavior.

The variability in vehicle speeds and distances across different road segments significantly challenges the safety of VRUs. In commercial and urban areas, the dynamic and unpredictable traffic conditions increase risks for VRUs, who must navigate frequent stops and the presence of various vehicle types. This environment heightens the probability of unexpected encounters with fast-moving vehicles, complicating safe navigation. In residential areas, lower speeds and the presence of pedestrians, particularly children, require enhanced vigilance. Although more predictable, industrial areas still present safety hazards due to the higher speeds of commercial vehicles. Therefore, understanding and raising awareness of these traffic patterns is crucial for improving road safety to VRUs.

4.2.2 Morning Patterns

The box plots provide a comprehensive visualization of the speed and distance data across different road segments, highlighting key patterns and trends. By examining the distribution, median, and range of these variables, we gain insights into the typical behavior and variability of vehicle movements in various environment

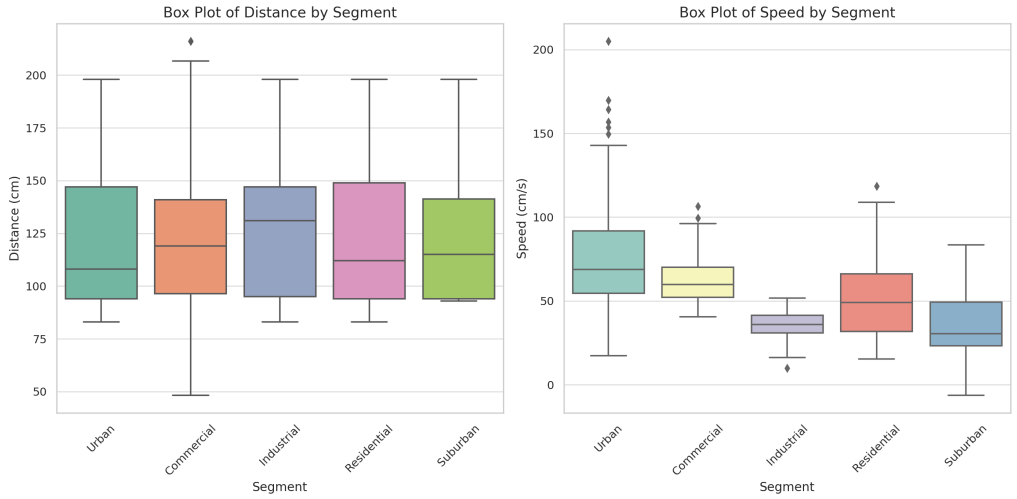


Figure 4.12: Box Plots by road segment in morning traffic

- **Distance by segment**: Urban Areas: The distance data for urban areas shows a moderate range with some outliers, indicating variability in how close vehicles approach the scooter. The median distance is relatively low, reflecting the dense traffic environment where vehicles frequently come closer to each other.

Commercial Areas: Commercial areas display a similar pattern to urban areas, with a moderate range of distances and fewer outliers. The median distance is slightly higher, indicating that vehicles tend to maintain a bit more space compared to urban traffic.

Industrial Areas: The distance distribution in industrial areas is narrow, with a higher median distance. This suggests that vehicles in industrial zones typically maintain greater distances, likely due to safety regulations and lower traffic density.

Residential Areas: Residential areas show a wider range of distances, with a lower median than industrial areas but higher than urban and commercial areas. This reflects the mixed-use nature of residential traffic, where vehicles might need to navigate around pedestrians and parked cars, leading to varied distances.

Suburban Areas: Suburban areas exhibit the broadest range of distances with many outliers, indicating significant variability. The median distance is similar to industrial areas, suggesting that while some vehicles maintain a safe distance, others come much closer, reflecting diverse traffic interactions.

- **Speed by segment**: Urban Areas: The box plot for urban areas indicates a wide range of speeds with numerous outliers, reflecting high variability. The median speed is relatively high, consistent with the fast-paced nature of city traffic. The outliers suggest occasional instances of very slow or very fast vehicles, likely due to traffic congestion or diverse vehicle types such as bicycles, motorcycles, and cars.

  Commercial Areas: The speed distribution in commercial areas is more concentrated, with fewer outliers and a lower median speed compared to urban areas. This suggests a more regulated traffic flow, influenced by commercial traffic rules and traffic signals.

  Industrial Areas: Industrial areas exhibit a narrow range of speeds with few outliers, indicating consistent vehicle speeds. The lower median speed reflects the controlled environment typical of industrial zones, where traffic is often regulated to ensure safety.

  Residential Areas: Residential areas show moderate variability in speeds, with a median speed lower than urban and commercial areas but higher than industrial and suburban areas. The wider interquartile range and outliers suggest occasional speeding, which may be a concern for road safety in these areas.

  Suburban Areas: Suburban areas have a broad range of speeds, with a median similar to industrial areas but a wider interquartile range. This reflects the mixed nature of suburban traffic, including both local and through traffic, resulting in varied speed patterns.

The presence of outliers, particularly in residential and suburban areas, indicates occasional instances of speeding and close vehicle approaches, posing safety risks. Identifications of these patterns are very useful to enhance road safety for VRUs

4.3 Limitations

During the course of this thesis project on improving safety for vulnerable road users, specifically e-scooter riders, several limitations were identified:

- Sensor Detection Range: The HCSR04 sensors used in this study have a limited detection range. They were unable to detect objects at longer distances, which restricts their effectiveness in providing comprehensive situational awareness in certain traffic conditions.

- Data Collection Challenges: Collecting measurements in real road traffic conditions presented significant challenges. Factors such as varying traffic volumes, environmental conditions, and the dynamic nature of road traffic made consistent and accurate data collection difficult. Additionally, positioning the sensors optimally on e-scooters to avoid obstruction and ensure accurate readings proved to be complex.

- Device Malfunctions: Occasional malfunctions of the HCSR04 sensors occurred during the data collection phase. These malfunctions included intermittent loss of signal and inaccurate distance measurements. Such technical issues required additional troubleshooting and recalibration, which impacted the overall efficiency of the data collection process.

- Lack of Real Traffic Data for Comparison: Obtaining real traffic data for accurate comparisons with the sensor data proved to be difficult. Due to this limitation, the study had to rely on estimates based on the TomTom Traffic Index. While the TomTom Index provided a reasonably reliable benchmark, it did not capture the specific nuances of the local traffic conditions encountered during the study period. This reliance on external data may have introduced some discrepancies in the analysis.

Despite these challenges, the insights gained from the collected data provide a valuable foundation for enhancing the safety of e-scooter riders.

4.4 Recommendations for Future work

- Enhanced Sensor Technology: Future research should focus on the development and integration of advanced sensor technologies with a greater detection range. Sensors with the capability to detect objects at longer distances would provide e-scooter riders with more comprehensive situational awareness, enabling them to respond more effectively to potential hazards in traffic.

- Integration of GPS Technology: Incorporating GPS technology into the sensor system can significantly enhance data collection and analysis. GPS can help identify specific locations where high speeds are frequently observed, thereby providing valuable insights into traffic hotspots. This information can be used to implement targeted safety measures and inform riders about areas where extra caution is needed.
- Robust Data Collection Methodologies: Future studies should explore more effective methodologies for collecting and analyzing traffic data in dynamic environments. This includes optimizing sensor placement on e-scooters to ensure unobstructed and accurate readings, as well as developing protocols to minimize data loss due to sensor malfunctions.

By addressing these recommendations, future work can build on the foundation laid by this study and make significant strides in improving the safety and situational awareness of e-scooter riders in urban traffic environments.

5 Conclusion

This thesis has explored the use of ultrasonic sensors to enhance the safety of vulnerable road users (VRUs), particularly e-scooter riders, by detecting approaching vehicles and providing real-time alerts. Employing a combination of HC-SR04 ultrasonic sensors, Arduino MKR WiFi 1010, and ESP32-WROOM-32 boards, a system capable of real-time data transmission and analysis was developed. Data were collected from various road segments, including urban, residential, commercial, suburban with school zones, and industrial areas, revealing distinct patterns in vehicle speed and distance, which provide valuable insights for improving road safety.

The HC-SR04 sensors demonstrated effectiveness in measuring distances and speeds of approaching vehicles within their detection range, though limitations in detection range and occasional malfunctions were noted. Real-time data transmission and processing were achieved using a client-server network setup, ensuring timely alerts for the e-scooter riders. Analyzing the collected data across different road segments highlighted the variability in traffic conditions.

Urban areas exhibited high variability in vehicle speeds and distances, indicative of dense and dynamic traffic conditions. Residential areas reflected the mixed-use nature of these zones, with pedestrians, parked cars, and strict speed limits contributing to significant variability in both speed and distance. Commercial areas had more regulated traffic flow with fewer outliers in speed and distance due to traffic signals and pedestrian activities. Suburban areas with school zones displayed higher and more varied vehicle speeds during evening rush hours compared to mornings, influenced by school schedules and residential traffic. Industrial areas demonstrated consistent vehicle speeds and distances, with lower speeds reflective of the controlled environ-

ment typical of these zones.

The study observed that evening traffic generally exhibited higher variability in speeds and distances across all segments, likely due to rush hour conditions and the presence of diverse vehicle types. Morning traffic was more predictable, with lower variability but still requiring careful management due to school zones and morning routines. However, several limitations impacted the comprehensiveness of the findings. The limited range of HC-SR04 sensors restricted the ability to detect objects at longer distances. Inconsistent data collection occurred due to varying traffic volumes, environmental conditions, and sensor positioning complexities. Occasional sensor malfunctions necessitated additional troubleshooting and recalibration. Moreover, the reliance on external data sources like the TomTom Traffic Index introduced potential discrepancies in the analysis.

To address these limitations and build on the findings, several recommendations are proposed for future research. Developing and integrating advanced sensors with greater detection ranges would provide more comprehensive situational awareness. Incorporating GPS technology could enhance data collection and identify specific high-risk locations for targeted safety measures. Additionally, optimizing sensor placement and developing protocols to minimize data loss and improve accuracy in dynamic environments are essential steps forward.

In conclusion, this thesis has demonstrated the potential of using ultrasonic sensor technology to improve the safety of VRUs. Despite the challenges encountered, the insights gained provide a valuable foundation for future advancements in this field. Implementing the recommended improvements can further enhance the safety and situational awareness of e-scooter riders, contributing to safer urban mobility environments. The findings and recommendations of this research align with global urban development goals, promoting safer and more sustainable urban mobility options and paving the way for more effective safety solutions in urban traffic environments.

References

[1] E. . Young, "Micromobility: moving cities into a sustainable future." [Online]. Available: https://assets.ey.com/content/dam/ey-sites/ey-com/en_gl/topics/automotive-and-transportation/automotive-transportation-pdfs/ey-micromobility-moving-cities-into-a-sustainable-future.pdf

[2] N. R. Safety, "Fact sheet: Vulnerable road users," 2021. [Online]. Available: https://www.roadsafety.gov.au/nrss/fact-sheets/vulnerable-road-users

[3] Y. Zhang, J. D. Nelson, and C. Mulley, "Learning from the evidence: Insights for regulating e-scooters," *Transport Policy*, vol. 151, pp. 63–74, 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0967070X24000982

[4] E. T. S. Council, "E-scooter safety: new study reveals key factors to prevent serious injuries," 2023. [Online]. Available: https://etsc.eu/e-scooter-safety-new-study-reveals-key-factors-to-prevent-serious-injuries/

[5] R. D. and H. koya, "Iot based smart vehicle parking system using rfid," vol. 10, pp. 53–60, 02 2024.

[6] H. Ahmad, A. Khan, W. Noor, G. Sikander, and S. Anwar, "Ultrasonic sensors based autonomous car parking system," 04 2018.

[7] A. Ziębiński, R. Cupek, D. Grzechca, and L. Chruszczyk, "Review of advanced driver assistance systems (adas)," vol. 1906, 11 2017, p. 120002.

[8] A. A. Karthik Ramasubramanian, Kishore Ramaiah, "Moving from legacy 24 ghz to state-of-the-art 77 ghz radar," 2017. [Online]. Available: https://www.ti.com/lit/wp/spry312/spry312.pdf

[9] M. Sadaf, Z. Iqbal, A. R. Javed, I. Saba, M. Krichen, S. Majeed, and A. Raza, "Connected and automated vehicles: Infrastructure, applications, security, critical challenges, and future aspects," *Technologies*, vol. 11, no. 5, 2023. [Online]. Available: https://www.mdpi.com/2227-7080/11/5/117

[10] A. Jenkins, "Remote sensing technology for automotive safety," 2007. [Online]. Available: https://www.microwavejournal.com/articles/5689-remote-sensing-technology-for-automotive-safety

[11] B. Yadav, "radar-technology-automotive," 01 2024.

[12] A. F. Santamaria, C. Sottile, F. D. Rango, and M. Voznak, "Road safety alerting system with radar and GPS cooperation in a VANET environment," in *Wireless Sensing, Localization, and Processing IX*, S. A. Dianat and M. D. Zoltowski, Eds., vol. 9103, International Society for Optics and Photonics. SPIE, 2014, p. 91030G. [Online]. Available: https://doi.org/10.1117/12.2053299

[13] A. francesco, C. sottile, A. lupia, and pierfrancesco, "An efficient traffic management protocol based on ieee802.11p standard," 07 2014.

[14] M. motro Taewan kim, "Communications and radar-supported transportation operations and planning: Final report," 2016. [Online]. Available: https://library.ctr.utexas.edu/ctr-publications/0-6877-1.pdf

[15] F. awais Haider naqv, "Communications and radar-supported transportation operations and planning: Final report," 2022. [Online]. Available: https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=9690855

[16] E. Hargis, "Blind spot accidents," 2024. [Online]. Available: https://www.lilawyer.com/blog/blind-spot-accidents/

[17] S. S. Gale Bagi, H. G. Garakani, B. Moshiri, and M. Khoshnevisan, "Sensing structure for blind spot detection system in vehicles," in *2019 International Conference on Control, Automation and Information Sciences (ICCAIS)*, 2019, pp. 1–6.

[18] S. bawankar Abhishek bhamare Somesh bhamre Harsh batheja Ganesh korwar, "Driver safety system," 2023. [Online]. Available: https://www.ijraset.com/best-journal/driver-safety-system

[19] T. I. BV, "Maps and location technology to keep the world moving," 2024. [Online]. Available: https://www.tomtom.com/

[20] W. Secretariat, "Traffic index, global coverage," 2024. [Online]. Available: https://wgicouncil.org/traffic-index-global-coverage/#:~:text=The%20TomTom%20Traffic%20Index,city%2Dby%2Dcity%20information.

[21] T. I. BV, "Padua traffic report," 2024. [Online]. Available: https://www.tomtom.com/traffic-index/padua-traffic/

[22] L. Elefteriadou, *An Introduction to Traffic Flow Theory*, 01 2014, vol. 84.

[23] WebAssign, "Fundamental principles of traffic flow," 2024. [Online]. Available: https://demo.webassign.net/ghtraffiche5/Garber5e_Chapter6_US.pdf

[24] transport research board of the national academics, "Highway-capacity-manual," 2010. [Online]. Available: https://www.jpautoceste.ba/wp-content/uploads/2022/05/Highway-Capacity-Manual-2010-PDFDrive-.pdf

[25] M. treiber Arne kesting, *Traffic Flow Dynamics*, 01 2013.

[26] maxbotix, "How noise and temperature can affect sensor operation," 2019. [Online]. Available: https://maxbotix.com/blogs/blog/noise-temperature-sensor-operation

[27] D. L. Gerlough and M. J. Huber, "Fundamental principles of traffic flow," 1975. [Online]. Available: https://onlinepubs.trb.org/onlinepubs/sr/sr165/165.pdf

[28] testbook, "Relative speed: Learn its definition, formula, special cases, examples," 2024. [Online]. Available: https://testbook.com/physics/relative-speed#:~:text=If%20two%20objects%20are%20moving,object%20which%20is%20being%20observed.

Acknowledgments

This thesis acknowledgment is a tribute to all the individuals who have contributed to accomplishing my academic endeavors. I thank my supervisor, Proffesor Andrea Zanella for allowing me to work on this project and providing valuable guidance and feedback. I am especially grateful to my family for their unwavering support, which served as an inspiration for my perseverance throughout this dissertation. Lastly, my peers' encouragement and the intellectual environment at the University of Padova have been vital throughout my study