CORNER SENTINEL

PROJECT REPORT SUBMITTED BY

SIDHARTH P PAI (ASI20EC061)

SREEDEEP K (ASI20EC065)

SREERAM K R (ASI20EC069)

UDITH G MENON (ASI20EC072)

In partial fulfillment of the requirements

for the award of the degree of

BACHELOR OF TECHNOLOGY

in

ELECTRONICS AND COMMUNICATION ENGINEERING

APJ Abdul Kalam Technological University

Adi Shankara Institute of Engineering

and Technology, Kalady

CERTIFICATE

Certified that this is a bonafide record of the project entitled

CORNER SENTINEL

Submitted by

SIDHARTH P PAI (ASI20EC061)

SREEDEEP K (ASI20EC065)

SREERAM K R (ASI20EC069)

UDITH G MENON (ASI20EC072)

during the year 2023 - 24 *in partial fulfillment of the requirement for the award of the degree of*

BACHELOR OF TECHNOLOGY

in

ELECTRONICS AND COMMUNICATION ENGINEERING

Project Supervisor

Head of the Department

Internal Examiner

External Examiner

DECLARATION

We hereby declare that the project report **CORNER SENTINEL**, submitted for partial fulfillment of the requirements for the award of degree of Bachelor of Technology of the APJ Abdul Kalam Technological University, Kerala is a bonafide work done by us under supervision of **Dr. Bobby Mathews .C**

This submission represents our ideas in our own words with a similarity index of 21% and where ideas or words of others have been included, we have adequately and accurately cited and referenced the original sources.

We also declare that we have adhered to ethics of academic honesty and integrity and have not misrepresented or fabricated any data or idea or fact or source in oury submission. We understand that any violation of the above will be a cause for disciplinary action by the institute and/or the University and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been obtained. This report has not been previously formed the basis for the award of any degree, diploma or similar title of any other University.

SIDHARTH P PAI
SREEDEEP K
SREERAM K R
UDITH G MENON

Kalady
07-05-2024

Acknowledgement

At the very outset we would like to give the first honors to God, who gave the wisdom and knowledge to complete this Project.

Our extreme thanks to **Dr. M S Murali**, Principal, Adi Shankara Institute of Engineering and Technology for providing the necessary facilities for the completion of this Project in our college.

We take this opportunity to express our deepest sense of gratitude and sincere thanks to everyone who helped us to complete this work successfully. We express our sincere thanks to **Dr. Ajay Kumar**, Head of Department, Electronics and Communication Engineering, Adi Shankara Institute of Engineering and Technology for providing us with all the necessary facilities and support.

We would like to express our sincere gratitude to **Ms. Savitha Ragavan**, and **Mr. Albins Paul**, department of Electronics and Communication Engineering, Adi Shankara Institute of Engineering and Technology Kalady for the support and co-operation.

We would like to place on record our sincere gratitude to our project guide **Dr. Bobby Mathews .C**, Professor, Electronics and Communication Engineering, Adi Shankara Institute of Engineering and Technology for the guidance and mentorship throughout this work.

Finally we thank our family, and friends who contributed to the successful fulfilment of this project work.

SIDHARTH P PAI

SREEDEEP K

SREERAM K R

UDITH G MENON

Vision and Mission

VISION

To be recognized at national and international level for excellence in education and research in Electronics and Communication Engineering.

MISSION

- Inculcating leadership qualities, adaptability and ethical values
- Imparting quality education in the field of electronics, communication and related areas to meet the challenges in industry, academia and research
- Nurture the growth of each individual by providing a dynamic and conducive learning environment.

Abstract

This project report outlines the development and deployment of Corner Sentinel, a cutting-edge warning system specifically designed to combat accidents occurring at treacherous blind corners in hilly regions. Corner Sentinel leverages a powerful trifecta of technologies: image processing for visual data analysis, machine learning for intelligent hazard recognition, and Li-Fi communication for ultra-fast, real-time data transmission. This system boasts the capability to detect a range of potential threats, including a vehicle's speed, type, and instances of wrong-lane maneuvers. Furthermore, Corner Sentinel can predict imminent collisions, providing invaluable pre-emptive warnings to approaching vehicles. These real-time alerts, delivered directly to drivers, equip them with the critical information necessary to take evasive action and prevent accidents. By implementing Corner Sentinel on these hazardous road sections, we have the potential to significantly reduce the number of accidents at steep corners, fostering a safer driving experience for all travelers.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Problem Statement

Traffic accidents on sharp corners are a persistent threat, claiming lives due to several factors. Limited visibility around the bend creates blind spots, making it difficult for drivers to gauge oncoming traffic. Challenging road conditions, such as steep inclines or declines, further restrict sight lines and complicate vehicle control. Perhaps the most significant danger is the potential for head-on collisions. When drivers misjudge the distance or presence of vehicles approaching from the opposite direction, the consequences can be catastrophic. This chapter highlights the critical need for a solution that addresses these inherent dangers posed by steep corners.

1.2 Motivation

The alarming number of accidents at blind corners necessitates a reliable and effective warning system to enhance road safety, particularly in hilly terrains. Current safety measures, such as signage and road markings, often prove inadequate in situations with limited visibility. The motivation behind this project stems from the desire to create a proactive approach to accident prevention. By implementing a comprehensive warning system, drivers can be alerted to potential hazards ahead, allowing them to take timely corrective actions and avoid collisions. This not only protects lives but also fosters a sense of security for those navigating challenging road conditions

1.3 Relevance

Corner Sentinel emerges as a critical solution to address the safety concerns associated with steep corners on mountainous or hilly roads. Its primary objective is to significantly reduce the number of accidents at these locations, ultimately saving lives. The relevance of this project lies in its ability to directly impact road safety in geographically diverse regions. Corner Sentinel has the potential to be implemented on existing infrastructure, making it a practical and cost-effective solution for transportation authorities worldwide. By prioritizing safety and promoting responsible driving habits, Corner Sentinel aims to create a more secure environment for all road users.

1.4 Objectives

The core functionalities of Corner Sentinel address several key aspects to enhance road safety at blind corners. First, a real-time vehicle detection and tracking system, powered by the latest YOLOv8 object detection algorithms, will continuously monitor traffic flow and identify approaching vehicles. Secondly, the system will integrate road segmentation with YOLO to detect vehicles that have mistakenly entered oncoming traffic lanes, triggering immediate warnings. Corner Sentinel will then leverage its accident/crash detection capabilities to further respond to critical situations. To effectively communicate with approaching vehicles, a Li-Fi communication system will be implemented for real-time alerts. Finally, the entire system will be optimized to ensure reliable functionality under various weather and environmental conditions for real-world deployments. This comprehensive set of functionalities aims to significantly reduce the risk of accidents at blind corners.

1.5 Organisation of the report

The report explores the relevant background and literature review. This section examines existing solutions and research related to corner safety and accident prevention systems. A detailed explanation of the chosen methodology and the overall system architecture of Corner Sentinel is presented next. This section details the technical aspects of the system. Following the technical explanation, the report presents the results achieved during the project's development. This section discusses the functionalities and performance of Corner Sentinel in detail. Finally, the report concludes by summarizing the key findings of the project and outlining potential future directions for further development and refinement of Corner Sentinel.

Chapter 2

Background Theory

This chapter delves into the fundamental theoretical concepts that underpin the Corner Sentinel system, encompassing the domains of image processing, machine learning, and Li-Fi communication. These technologies form the bedrock upon which Corner Sentinel's functionality is built, enabling it to effectively detect and mitigate potential accidents at steep corners.

2.1 Image Processing

Image processing is a multifaceted discipline that revolves around the manipulation and analysis of digital images. Its applications span a wide spectrum, from enhancing image quality for human viewing to extracting meaningful information for automated systems. In the context of Corner Sentinel, image processing plays a pivotal role in several key tasks:

2.1.1 Object Detection and Classification

Object detection involves identifying and locating objects of interest within an image or video stream. Corner Sentinel relies on object detection to identify vehicles, categorize them into different types (e.g., cars, trucks, motorcycles, ambulances), and track their movement over time. This information is crucial for assessing potential risks and generating appropriate warnings.

2.1.2 Lane Detection

Lane detection is the process of identifying the boundaries of lanes on a road. This is essential for determining whether a vehicle is veering out of its lane or driving on the wrong side of the road. Corner Sentinel employs lane detection to provide warnings for lane departure and wrong-way driving, contributing to accident prevention.

2.1.3 Optical Flow

Optical flow is a technique used to estimate the apparent motion of objects, surfaces, and edges in a visual scene caused by the relative movement between the observer (camera) and the scene. In Corner Sentinel, optical flow is employed to calculate the speed of vehicles by analyzing the movement of pixels between consecutive frames in a video stream. This allows for real-time speed estimation and the generation of alerts for speeding vehicles.

2.2 Machine Learning

Machine learning is a sub-field of artificial intelligence that empowers computers to learn from data without being explicitly programmed. It encompasses a wide range of algorithms and techniques that enable systems to improve their performance on a specific task through experience. Corner Sentinel leverages machine learning, specifically deep learning, for object detection and classification tasks.

2.2.1 Deep Learning and Convolutional Neural Networks

Deep learning is a subset of machine learning that employs artificial neural networks with multiple layers to progressively extract higher-level features from raw input data. Convolutional Neural Networks (CNNs) are a type of deep learning architecture that excels at image recognition and classification. CNNs consist of multiple layers, including convolutional layers that extract features from images, pooling layers that reduce dimensionality, and fully connected layers that perform classification.

2.2.2 YOLOv8 Object Detection

YOLO (You Only Look Once) is a state-of-the-art real-time object detection algorithm that has gained significant popularity in recent years. YOLOv8 is the latest iteration of the YOLO family, offering improved accuracy, speed, and versatility. Corner Sentinel utilizes YOLOv8 for its real-time object detection capabilities, enabling it to identify and classify vehicles with high precision and efficiency. YOLOv8's architecture is based on a single-stage detector, meaning it performs object detection and classification in a single pass through the network, contributing to its fast inference speed.

2.3 Li-Fi Communication

Li-Fi (Light Fidelity) is a wireless communication technology that utilizes light to transmit data. It offers several advantages over traditional radio frequency (RF) based communication, including:

- **High bandwidth:** Li-Fi operates in the visible light spectrum, which has a much wider bandwidth compared to RF, enabling higher data rates.
- **Low latency:** Li-Fi signals experience minimal delays, making it suitable for real-time applications like Corner Sentinel.
- **Security:** Light signals cannot penetrate walls, providing inherent security and preventing eavesdropping.
- **Immunity to electromagnetic interference:** Li-Fi is not susceptible to interference from other electronic devices, ensuring reliable communication in environments with high electromagnetic activity.

2.3.1 Li-Fi Modulation and Encoding

Corner Sentinel employs On-Off Keying (OOK) modulation with Manchester encoding for Li-Fi communication. OOK is a simple modulation scheme where the presence or absence of light represents binary 1s and 0s. Manchester encoding ensures that each bit period contains a transition (high to low or low to high), preventing long sequences of 0s or 1s that could cause LED flicker and making it easier for the receiver to synchronize with the transmitter.

2.3.2 Li-Fi Transceiver Design

Corner Sentinel's Li-Fi transceivers consist of LEDs for transmission and a solar panels for reception. The transmitter circuit modulates the LED's intensity according to the encoded data, while the receiver circuit detects the light intensity variations and decodes the information.

By combining these fundamental technologies, Corner Sentinel effectively addresses the challenge of preventing accidents at steep corners, offering a robust and reliable solution for enhancing road safety.

Chapter 3

Literature Review

This chapter delves into the existing research and technological advancements relevant to the development of Corner Sentinel. It explores various approaches to vehicle detection, tracking, wrong-lane identification, accident/crash detection, and Li-Fi communication, providing a comprehensive understanding of the current state-of-the-art.

3.1 Vehicle Detection and Tracking

Vehicle detection and tracking are fundamental tasks in computer vision with numerous applications in intelligent transportation systems, autonomous driving, and traffic monitoring. Various algorithms have been proposed and implemented over the years, with deep learning-based methods achieving significant breakthroughs in recent times. [1, 2, 4, 6]

- **YOLO (You Only Look Once):** YOLO is a family of single-shot object detectors known for their speed and accuracy. Corner Sentinel utilizes YOLO v8, which boasts improved performance and efficiency compared to its predecessors. YOLO algorithms work by dividing an image into a grid and predicting bounding boxes and class probabilities for each grid cell, enabling real-time object detection.

- **Faster R-CNN (Region-based Convolutional Neural Network):** Faster R-CNN is a two-stage object detector that first generates region proposals and then classifies and refines bounding boxes for each proposal. While generally slower than YOLO, Faster R-CNN often achieves higher accuracy, especially for small or occluded objects [2].

- **SSD (Single Shot MultiBox Detector):** SSD is another single-shot detector that predicts bounding boxes at multiple scales and aspect ratios, making it effective for detecting objects of various sizes. SSD offers a good balance between speed and accuracy [1].

The choice of object detector depends on the specific requirements of the application. Corner Sentinel prioritizes real-time performance, making YOLO v8 a suitable choice for vehicle detection and tracking.

3.2 Wrong-Lane Detection

Wrong-lane driving is a dangerous traffic violation that can lead to head-on collisions. Detecting such incidents requires a combination of road segmentation and lane detection techniques. [3, 5, 7]

- **Road Segmentation:** This involves partitioning an image into semantically meaningful regions, such as road surface, lane markings, and surrounding areas. Deep learning-based semantic segmentation models, such as U-Net and DeepLab, have demonstrated high accuracy in road segmentation tasks [5].

- **Lane Detection:** Once the road is segmented, lane markings can be identified and tracked using various algorithms. Hough transform and its variants are commonly used for detecting straight or curved lanes, while more advanced methods like LaneNet utilize deep learning for robust lane detection [3].

Corner Sentinel employs a combination of YOLO and a road segmentation model to identify vehicles and lane markings, enabling accurate detection of wrong-lane driving instances.

3.3 Accident/Crash Detection

Accidents or crashes can be detected using a variety of sensors and algorithms. Corner Sentinel focuses on vision-based approaches due to their non-intrusive nature and ability to capture rich visual information.

- **Change Detection:** By analyzing consecutive video frames, sudden changes in vehicle speed, position, or orientation can indicate a potential collision. Optical flow analysis and background subtraction techniques are commonly used for change detection [8].

- **Motion Analysis:** Examining the trajectory and acceleration of vehicles can help identify abnormal behaviors indicative of accidents. Kalmann filtering and particle filtering are popular methods for tracking and predicting vehicle motion [9].

- **Deep Learning-based Approaches:** Recent research explores the use of deep learning models trained on large datasets of accident footage to automatically detect and classify crashes. These methods offer promising results and may be integrated into Corner Sentinel in the future [8].

Corner Sentinel currently utilizes change detection and motion analysis techniques to identify potential accident scenarios.

3.4 Li-Fi Communication

Li-Fi is a novel wireless communication technology that leverages visible light to transmit data. It offers several advantages over traditional RF-based communication, including higher bandwidth, lower latency, and immunity to electromagnetic interference.

- **Modulation Techniques:** Li-Fi systems typically employ intensity modulation and direct detection (IM/DD) schemes, where the intensity of the light source is modulated to represent digital data. Common modulation techniques include On-Off Keying (OOK) and Pulse Position Modulation (PPM) [10]

- **Data Encoding:** To ensure reliable data transmission, error correction codes such as Cyclic Redundancy Check (CRC) are often used. Additionally, encoding schemes like Manchester encoding can mitigate flickering issues and improve signal detection . [11]

- **Transmitter and Receiver Design:** Li-Fi transmitters typically use high-brightness LEDs, while receivers employ solar cells to detect the modulated light signals. The design of the transmitter and receiver circuitry is crucial for achieving high data rates and reliable communication . [12]

Corner Sentinel utilizes a Li-Fi communication system with OOK modulation, Manchester encoding, and CRC error checking to transmit alerts to approaching vehicles.

Chapter 4

Proposed Model

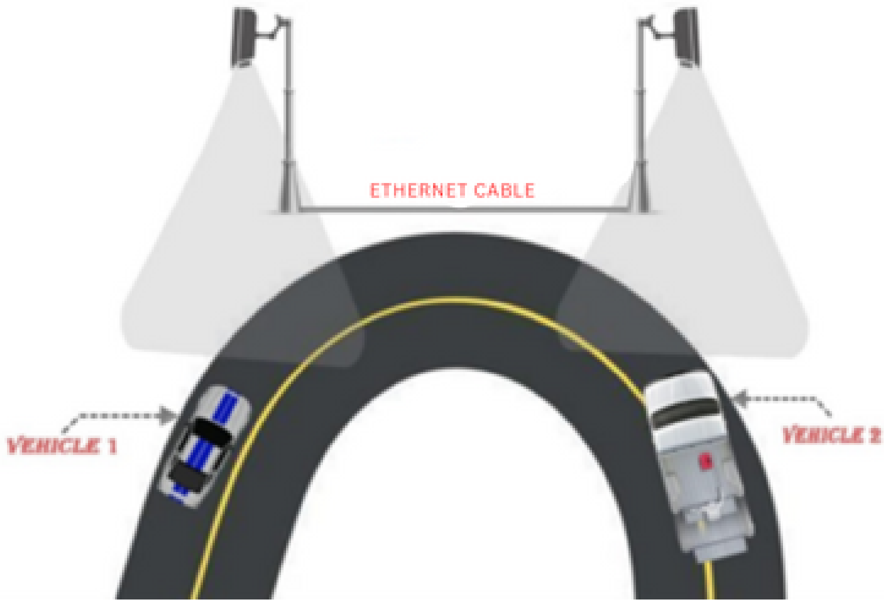

Figure 4.1: Proposed system model.

4.1 System Architecture

Corner Sentinel's design adopts a modular approach, separating the system's functionalities into two core modules: the Machine Learning Module and the Li-Fi Communication Module. This modular design offers several advantages. First, it promotes code re-usability and simplifies maintenance. By encapsulating specific tasks within distinct modules, developers can focus on improving individual components without affecting the entire system. Second, the modular design enhances scalability. If future advancements necessitate the addition of new functionalities, the modular architecture allows for easier integration without major system overhauls. Finally, the modular approach facilitates testing and debugging. By isolating potential issues within specific modules, troubleshooting becomes more efficient.

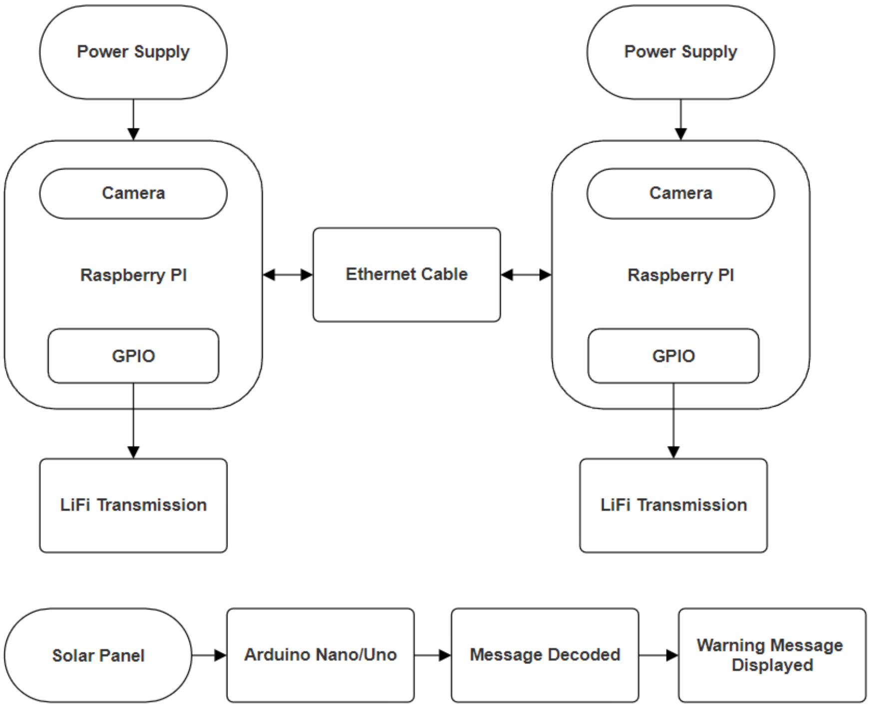

Figure 4.2: System Block Diagram

4.1.1 Machine Learning Module

At the heart of the Machine Learning Module lies the Raspberry Pi 5 single-board computer (SBC). This powerful unit acts as the central processing unit, running the YOLOv8 model. YOLOv8 is responsible for the crucial tasks of object detection, classification, and tracking. It analyzes the real-time video feed captured by the camera, accurately identifying and pinpointing the presence and location of vehicles on the road. Additionally, the system employs the optical flow algorithm to estimate the speed of each detected vehicle. By analyzing the data collected on vehicle types, speeds, and positions relative to lane markings, the system generates binary codes that represent potential hazards on the road.

4.1.2 Li-Fi Module

The Li-Fi Communication Module serves as the bridge between Corner Sentinel and approaching vehicles. The Li-Fi transmitter encodes the binary hazard codes generated by the Machine Learning Module. This encoded information is then modulated onto light signals emitted by LEDs, enabling data transmission via Li-Fi technology. Vehicles equipped with Li-Fi receivers capture these light signals and decode the binary codes. Based on the decoded information, the receiver provides corresponding alerts to the driver, such as warnings about approaching vehicles in the wrong lane or those exceeding speed limits.

4.2 Hardware Components

The successful operation of Corner Sentinel hinges on a carefully chosen selection of hardware components, each playing a vital role in the system's functionality.

4.2.1 Raspberry Pi 5

The Raspberry Pi 5 is the latest and most powerful single-board computer (SBC) in the Raspberry Pi series. It is equipped with a 2.4GHz Broadcom BCM2712 quad-core ARM Cortex-A76 processor, making it up to three times faster than its predecessor, the Raspberry Pi 4. The Pi 5 also boasts an improved 800MHz VideoCore VII GPU that enhances its graphics performance, in addition to dual 4Kp60 display output over HDMI With 8GB of LPDDR4-3200 SDRAM, the Pi 5 delivers ample memory for running demanding applications, such as video editing and machine learning. It also supports dual-channel memory for increased bandwidth. Additionally, the Pi 5 features a Gigabit Ethernet port for wired networking and dual-band Wi-Fi 5 (802.11ac) for wireless connectivity. The storage, the Pi 5 utilizes a microSD card slot that supports up to 2TB of storage. It also includes two USB 3.0 ports and two USB 2.0 ports for connecting peripherals. For power, the Pi 5 uses a USB Type-C connector and requires a 5V/3A power supply. The Raspberry Pi 5 comes pre-installed with Raspberry Pi OS (formerly Raspbian), a free and open-source operating system based on Debian. Raspberry Pi OS includes a wide range of software applications, including web browsers, office suites, and programming tools. Overall, the Raspberry Pi 5 is a powerful and versatile SBC that is suitable for a wide range of applications, from education and hobbyist projects to professional use.

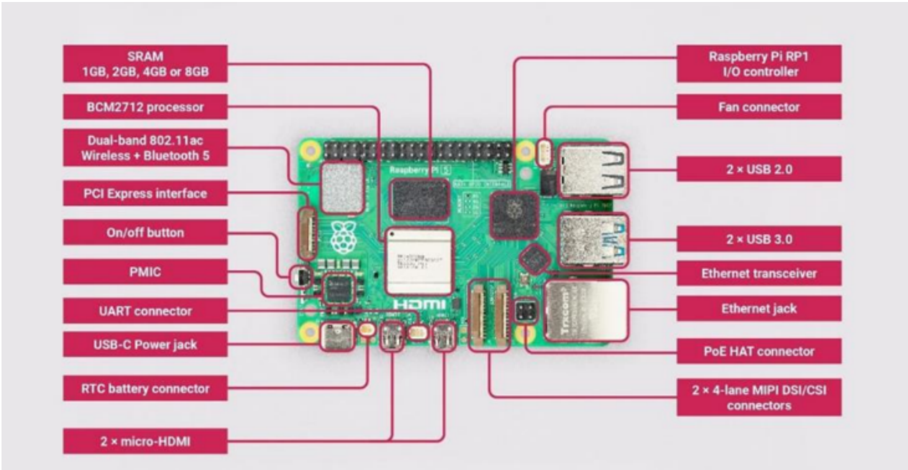

Figure 4.3: Raspberry Pi 5

4.2.2 Camera

Figure 4.4: Raspberry PI Surveillance Camera Module

A high-resolution camera module is crucial for capturing clear and detailed video footage of the road scene. The camera's resolution, frame rate, and low-light performance are critical factors influencing the system's accuracy. A higher resolution camera can capture finer details, improving object detection and classification, particularly for smaller vehicles or those at a distance. A faster frame rate allows the system to analyze changes in the scene more frequently, enabling real-time tracking of vehicles and their movements. Low-light performance becomes essential during dawn, dusk, or poor weather conditions, ensuring the system can still function effectively when traditional visible light cameras might struggle.

4.2.3 Light Emitting Diodes

Light-emitting diodes(LEDs) serve as the workhorses of the Li-Fi communication module, acting as the transmitters that convert electrical signals into light pulses for data transmission. By rapidly turning the LEDs on and off, the system can modulate the light intensity to encode the binary data representing the critical alerts. This modulation technique leverages the inherent ability of LEDs to switch on and off very quickly at rates exceeding human perception. The Li-Fi transmitter carefully controls the timing and duration of these LED pulses to create a specific pattern that corresponds to the binary code. The Li-Fi receiver on the approaching vehicle is designed to detect these rapid fluctuations in light intensity and decode the embedded binary data, ultimately translating them into meaningful warnings for the driver.

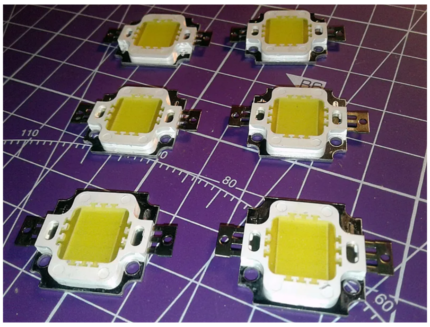

Figure 4.5: 10W Light Emitting Diode

4.3 Software Components

The software components are the invisible forces governing Corner Sentinel's intelligent behavior. These Software Components play a crucial role in detecting, and sending warning signals, monitoring incoming and outgoing traffic , constant vehicle surveillance many more functions.

4.3.1 Yolo V8

YOLO v8, a state-of-the-art machine learning algorithm, forms the core of the system's object detection and classification capabilities. Leveraging deep learning techniques, YOLO v8 is trained on massive datasets of images and labels, enabling it to recognize and distinguish various types of vehicles (cars, trucks, motorcycles, bicycles, ambulances) with exceptional accuracy in real-time video streams.

4.3.2 Open CV

OpenCV (Open Source Computer Vision Library) is a comprehensive open-source library that provides a rich set of functions and routines for computer vision and image processing. Corner Sentinel utilizes OpenCV's functionalities for various tasks essential for YOLO v8's operation. These tasks include video capture and frame acquisition, image pre-processing (noise reduction, filtering), and object detection/recognition. By leveraging OpenCV's capabilities, Corner Sentinel can efficiently process the raw video data captured by the camera and prepare it for YOLO v8's object detection and classification algorithms.

4.3.3 Google Colab

Google Colab is a cloud based Jupyter notebook environment that allows users to write and execute Python code in their web browser. This makes it a convenient platform for developing and testing Machine Learning applications. There are several benefits to using Google Colab for vehicular communication, including:

- Accessibility: Google Colab is accessible from any web browser, so users do not need to install any software or download any data.
- Scalability: Google Colab can scale to handle large datasets and complex computations.
- Cost-effectiveness: Google Colab is free to use, making it an affordable option for researchers and developers.

Overall, Google Colab is a valuable tool for researchers and developers working on vehicular communication. It provides a convenient, scalable, and cost-effective platform for developing, testing, and analysing Computer vision applications.

4.3.4 Raspbian OS

Raspbian OS is a lightweight operating system based on Linux that is designed for the Raspberry Pi minicomputer. It includes several features that make it well-suited for vehicular communication applications, including:

- **L**ow power consumption: Raspbian OS is very lightweight, which means that it can run on the Raspberry Pi's low-power processor. This makes it ideal for use in vehicles, where power consumption is a major concern.
- **S**upport for a variety of communication protocols: Raspbian OS supports several communication protocols that are commonly used in vehicular communication.
- **S**everal pre-installed software packages for vehicular communication: Visual Raspbian OS comes with several pre-installed software packages for vehicular communication. These features make Raspbian OS a powerful and versatile platform for vehicular communication applications.

4.3.5 Visual Studio Code

Visual Studio Code (VS Code) is a popular code editor that can be used for a variety of tasks, including developing vehicular communication applications. Here are some specific uses of VS Code in the context of vehicular communication:

- **P**rototyping and testing vehicular communication systems: VS Code can be used to quickly prototype and test vehicular communication systems. VS Code's support for extensions allows developers to easily add features to VS Code that are specific to vehicular communication.
- **V**isualizing vehicular communication data: VS Code can be used to visualize vehicular communication data, such as traffic data and sensor data. VS Code's support for extensions allows developers to create custom visualizations for vehicular communication data.

- Collaborating on vehicular communication projects: VS Code's built-in support for collaboration features, such as Git and live coding, can be used to help teams of developers collaborate on vehicular communication projects.

In addition to these specific applications, Raspbian OS can also be used for a wide variety of other vehicular communication applications. For example, it can be used to develop new applications for in-vehicle infotainment, autonomous driving, and connected cars. Overall, Raspbian OS is a powerful and versatile platform for vehicular communication applications. Its low power consumption, support for a variety of communication protocols, and pre-installed software packages make it an ideal choice for developers who are looking to create innovative vehicular communication solutions.

4.4 Working Principle

Corner Sentinel's operation can be segmented into two key stages: the Machine Learning Module's processing and the Li-Fi Communication Module's data exchange.
Both of them work hand in hand for real-time visualization of vehicular information and quick data transfer to targeted vehicles. This collaborative operation ensures that relevant data is swiftly communicated and visually represented, enabling efficient decision-making and enhancing overall road safety measures.

4.4.1 Machine Learning Model

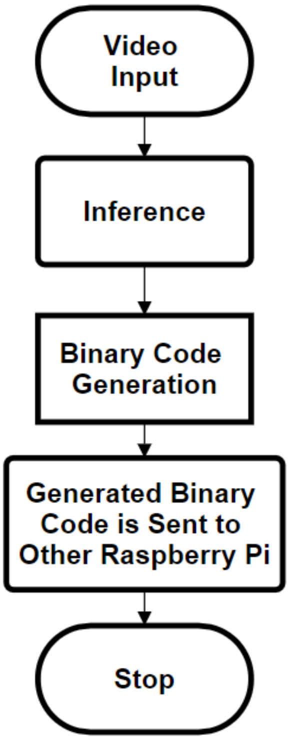

Figure 4.6: ML Module Working

In this system, a camera captures real-time video footage of the road, which is then processed by the YOLO v8 algorithm. This algorithm is capable of analyzing each video frame, identifying vehicles, classifying them, and tracking their movements. Additionally, an optical flow algorithm is employed to calculate the speed of each detected vehicle based on the video data. By combining these functionalities, the system can effectively monitor and analyze the traffic situation.The output of the system is a unique binary code that represents potential hazards, such as wrong-lane driving or high-speed vehicles. This code is generated based on factors such as the type of vehicle, its speed, and its position relative to the road and lane markings. Overall, this system represents a significant advancement in road safety technology, as it provides real-time monitoring and analysis of traffic conditions, enabling authorities to take proactive measures to prevent accidents and ensure the safety of road users.

4.4.2 Li-Fi Module Implementation

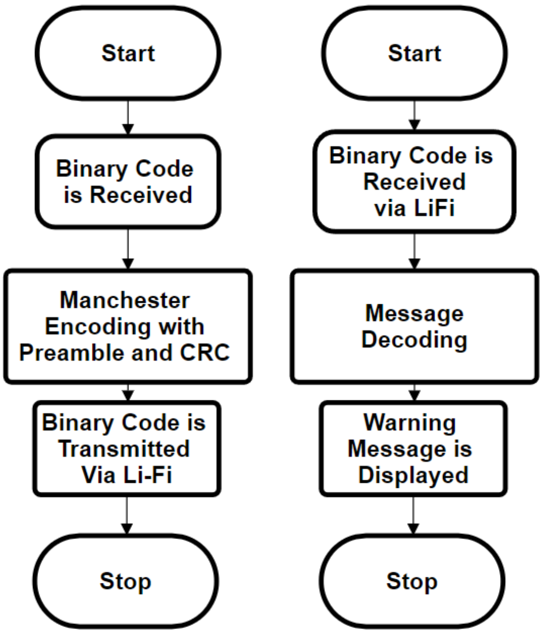

Figure 4.7: Li-Fi Module Working

In the context of road safety, the use of Light Fidelity (Li-Fi) technology for transmitting potential hazards to approaching vehicles represents a significant advancement. This system involves encoding binary codes representing potential hazards onto LEDs, which are then modulated for transmission. Li-Fi receivers installed in approaching vehicles detect these modulated light signals and decode the binary codes to extract the hazard information. This approach offers several advantages over traditional communication methods. Firstly, Li-Fi enables high-speed data transmission, which is crucial for real-time communication of hazards. Additionally, Li-Fi is secure and immune to electromagnetic interference, making it ideal for use in vehicular communication systems. The decoded information can be used to alert drivers about potential hazards ahead, such as accidents, roadblocks, or adverse weather conditions, allowing them to take necessary precautions and avoid potential accidents. Moreover, the system can be integrated with existing vehicle safety systems, enhancing overall road safety. However, challenges such as signal interference and the need for standardized protocols must be addressed for the widespread adoption of Li-Fi-based hazard communication systems on roads. Overall, Li-Fi technology shows great promise in improving road safety by providing timely and accurate hazard information to drivers.

4.5 Binary Code Technology

The following image shows the bit allocation for the 8-bit binary data, which is sent into the Li-Fi transmitter. The first bit indicates whether there is a stationary vehicle in the scene or not. The next bit signifies whether an accident has occurred on the road. The next three bits are assigned for specifying the vehicle type, such as cars, HMVs, ambulances, bikes, etc. Following these bits, there is a bit assigned for detecting whether the approaching vehicle is in the wrong lane of the road. Finally, there are two bits that classify the speed of the vehicle into below 40 km/h, 40 to 60 km/h, and above 60 km/h. A detailed representation of the bitwise map is explained in the following section.

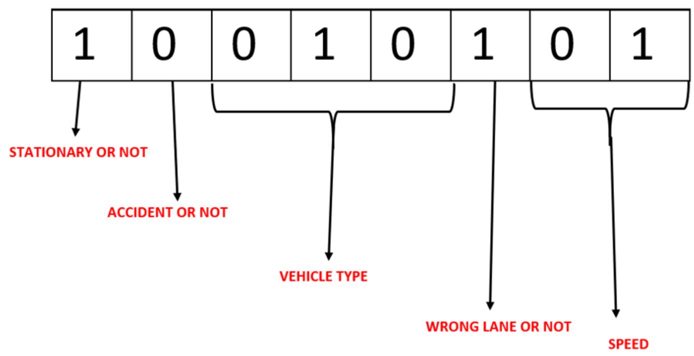

Figure 4.8: Data bit allocation

4.5.1 Bit wise Binary code table

Bit	7	6	5	4	3	2	1	0	Specification
Val	St	A	C2	C1	C0	W	S1	S0	
St	0								No Stationary Vehicle
St	1								Stationary Vehicle Detected
A		0							No Accident/Crash
A		1							Accident/Crash Detected
C			1	0	0				Ambulance
C			0	1	0				Car
C			0	0	1				Bike
C			0	1	1				Truck/Bus (HMV)
W						0			Not Wrong Lane
W						1			Wrong Lane
S							0	1	1.5kmph < Speed < 40kmph
S							1	0	40kmph<Speed<60kmph
S							1	1	Speed > 60kmph

Figure 4.9: Bit wise Binary Table

The binary representation is a compact way to convey multiple pieces of information about the detected object, such as its class, whether it is on the wrong side, and its speed condition. The exact interpretation of the bits will depend on the specific mapping chosen for the class ID and speed conditions.

This table details the specific bit assignments and their corresponding meanings within the binary code used by CornerSentinel for Li-Fi communication. Each bit represents a specific attribute or hazard, allowing for efficient and concise transmission of critical information to approaching vehicles.

- Stationary Vehicle Detection (Bit 7): This bit indicates the presence or absence of a stationary vehicle on the road ahead. A value of 'St 1' alerts drivers to the potential obstacle, prompting them to exercise caution and adjust their speed accordingly.

- Accident/Crash Detection (Bit 6): This bit signifies whether an accident or crash has been detected in the vicinity. A value of 'A 1' indicates a potential hazard, urging drivers to be extra vigilant and prepared to stop or take evasive action.

- Vehicle Classification (Bits 5-3): These three bits categorize the type of vehicle detected, allowing for differentiated warnings based on the potential risks associated with different vehicle types. For instance, the system might prioritize warnings for larger vehicles like trucks or buses due to their increased stopping distance and potential for causing more severe damage in collisions.

- **W**rong Lane Detection (Bit 2): A value of 'W 1' indicates that a vehicle is traveling in the wrong lane, representing a significant hazard that requires immediate attention from both the wrong-way driver and oncoming vehicles.

- **S**peed Condition (Bits 1-0): These two bits provide information about the speed range of the detected vehicle, enabling the system to alert drivers to potential risks associated with excessive speed or sudden braking. The specific speed thresholds for each range are determined based on the road conditions, traffic regulations, and safety considerations. The lowest speed range (1.5 kmph to 40 kmph) might indicate a slow-moving vehicle or potential congestion, while the highest speed range (above 60 kmph) could signify a speeding vehicle or a situation requiring increased vigilance.

4.5.2 16-Bit Message code table

5-Bit Preamble	8-Bit Code	3-Bit CRC
10101	10010101	110
Sychronization	Binary Data Bits	Error Checking Code

Table 4.1: 16 Bit Message Code Table

This table illustrates the structure of the data packet transmitted by CornerSentinel via Li-Fi. Each packet consists of three main components: a preamble for synchronization, the actual data payload containing the binary code, and a cyclic redundancy check (CRC) for error detection.

- **P**reamble (5 bits): The preamble consists of a predefined sequence of bits (10101 in this case) that serves as a synchronization marker. The receiver detects this pattern to identify the start of a new data packet and synchronize its internal timing with the transmitter. This ensures that the subsequent data bits are correctly sampled and interpreted.

- **D**ata Payload (8 bits): This section contains the actual binary code generated by the ML module based on the identified hazards. The 8-bit data payload encodes information about the type of hazard, vehicle class, and speed condition, as described in Table 1.

- **C**RC (3 bits): The cyclic redundancy check (CRC) is an error detection code appended to the data packet. The transmitter calculates the CRC based on the data payload and appends it to the packet. The receiver performs the same CRC calculation upon receiving the packet and compares it with the received CRC. If the two values match, it indicates that the data was transmitted without errors. If a mismatch occurs, it suggests that errors were introduced during transmission, and the receiver can request re-transmission or take appropriate error handling measures.

This structured approach to data packet transmission ensures reliable and accurate communication between CornerSentinel and approaching vehicles, enabling timely delivery of safety warnings, and contributing to improved road safety at critical road sections.

Chapter 5

Results and Discussion

The initial phase of Corner Sentinel's development yielded promising results, successfully establishing the core functionalities of the system. The system demonstrated accurate vehicle detection and tracking, enabling it to pinpoint the presence and location of vehicles on the road. Classification capabilities allowed Corner Sentinel to distinguish between different vehicle types, such as cars, trucks, motorcycles, and ambulances. Furthermore, the system successfully implemented speed estimation using the optical flow method, providing valuable data on vehicle movement. To lay the groundwork for future communication with approaching vehicles, basic Li-Fi communication was established, showcasing the potential for data transmission using light signals.

5.1 Result

5.1.1 Finalised Circuits

At the transmitter side, 10W power LEDs are utilized for lightning the test room. There are cold-white and warm-white options available for power LEDs, however, our system is independent of color temperature for information. For switching power LEDs, a driver MOSFET is required which costs more than a basic NPN transistor. An NMOS with low RDS-on resistance is needed for minimal power loss due to heating which removes the need of a heat sink. To drive the power MOSFET we need 5V digital output. But since the Raspberry Pi only has a 3.3V digital output, we are again using a BC547 transistor to provide the 5V for switching. [11]

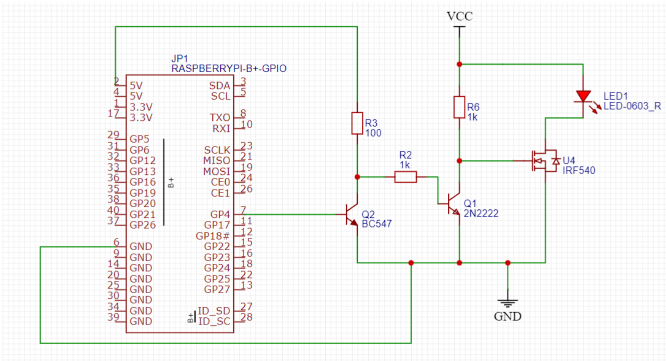

Figure 5.1: Transmitter Circuit.

At the receiver side, we have a solar panel used for receiving the message. This is connected to an Arduino which then decodes and displays the respective warning message on an LCD.

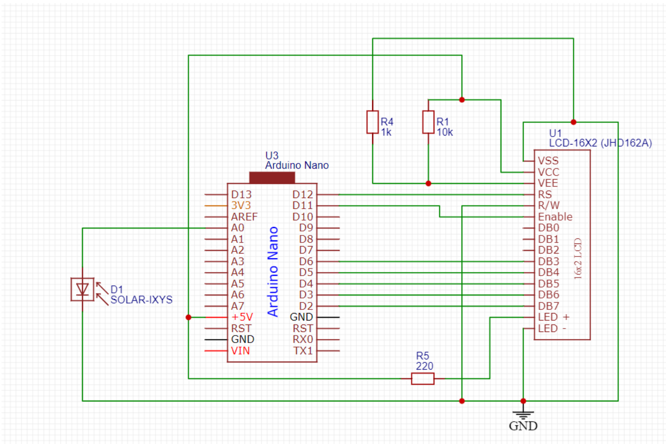

Figure 5.2: Receiver Circuit.

5.1.2 Output

The initial development phase successfully integrated YOLO into a real-time traffic monitoring system. Leveraging camera feeds, it identified and classified vehicles while estimating speeds. This paves the way for future advancements in traffic management and autonomous vehicles. Also a basic Li-Fi circuit was implemented.

In the second phase of development, Corner Sentinel enhanced its capabilities with features like wrong-lane detection, improved accident/crash detection, and robust Li-Fi communication. Wrong-lane detection combines road segmentation and YOLO object detection, crucial for safety at steep corners. The accident/crash detection algorithm was refined for better analysis of vehicle behavior. Enhancements in Li-Fi communication included preamble synchronization, CRC error checking, and Manchester encoding. Full-duplex communication between Raspberry Pi-5 SBCs allows real-time information exchange among units, and multi-threading optimizes system performance. Dynamic thresholding in the Li-Fi receiver adjusts the OOK threshold based on ambient light, maintaining signal accuracy in varying lighting. Below we have attached some of the frames from out ML model detecting and generating said binary codes. The first image is of the system detecting a vehicle coming at wrong lane and generating it's respective binary code.

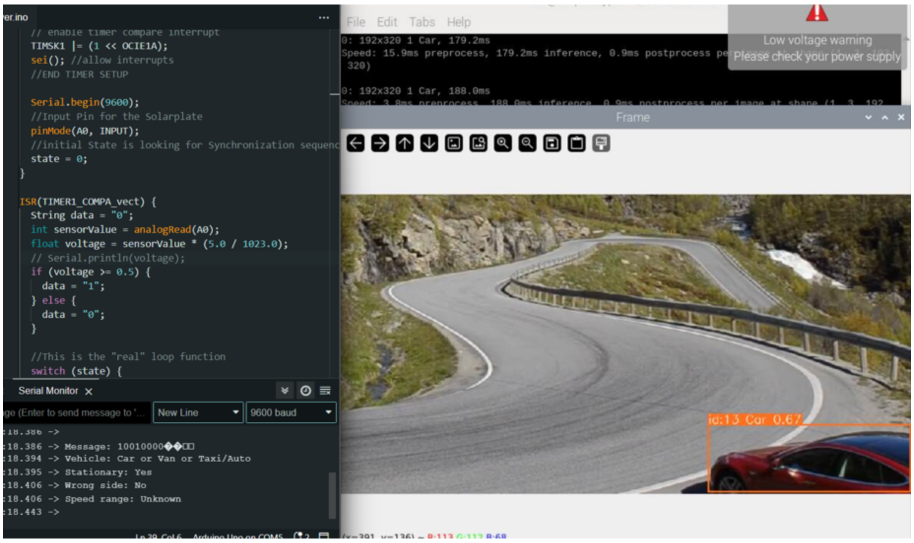

Figure 5.3: ML Output Frame - Stationary Vehicle Detection

The second image is of the system detecting a stationary vehicle and generating it's respective binary code.

Figure 5.4: ML Output Frame - Wrong lane Detection

Practical Implementation of the Circuits:

Here the transmitter circuit has been in a ABS plastic casing. the 12V needed is supplied by a custom battery pack consisting of three 3.7V rechargeable batteries connected in series.

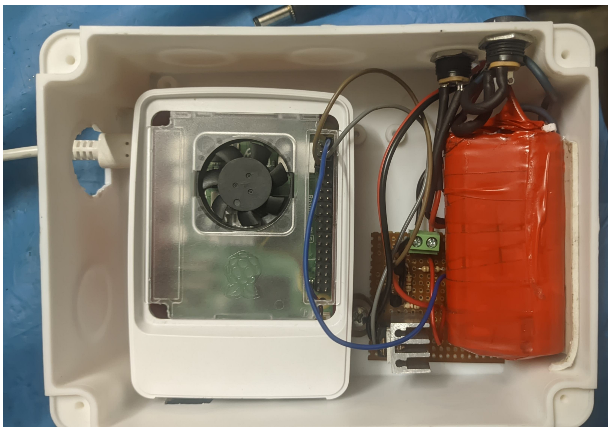

Figure 5.5: Realised Transmitter Circuit

Here the receiver module has been implemented on a printout zero layer PCB including the LCD display. It is powered by a typical 9V cell.

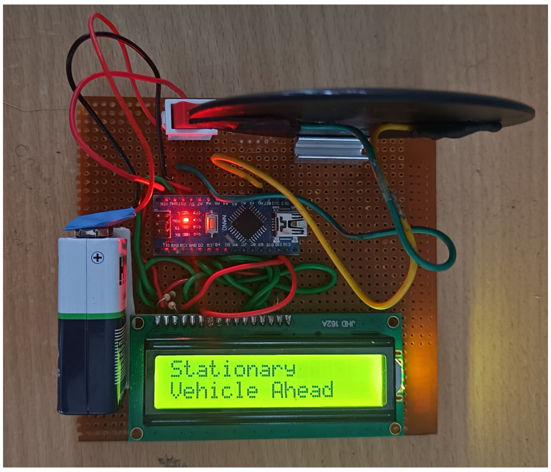

Figure 5.6: Realised Receiver Circuit

5.2 Limitations

While Corner Sentinel has achieved significant progress, it is essential to acknowledge certain limitations that will guide future development efforts. The system's effectiveness may be influenced by adverse weather conditions such as heavy rain or fog. These conditions can potentially impair camera vision and disrupt Li-Fi signal transmission. Another limitation is the inherent range constraint of Li-Fi communication compared to traditional radio frequency (RF) based systems. While Li-Fi offers several advantages, its effective range is shorter, necessitating strategic placement of Corner Sentinel units to ensure consistent coverage at blind corners. The computational demands of the YOLO v8 model also warrant consideration. For wider deployment on resource-constrained devices with lower processing power, the model might require further optimization to ensure efficient performance without compromising accuracy. By acknowledging these limitations, the Corner Sentinel project can strive for continuous improvement, addressing these challenges through innovative solutions and paving the way for a more robust and universally deployable road safety system.

Chapter 6

Conclusions and Future Scope

6.1 Conclusions

CornerSentinel emerges as a groundbreaking innovation in road safety technology. It leverages the combined power of Artificial Intelligence (AI) and Li-Fi communication to tackle the challenges posed by blind corners and limited visibility on winding roads. This system acts as a guardian angel, proactively detecting potential hazards, assessing the severity of risks, and transmitting critical warnings to approaching vehicles in real-time. By empowering drivers with this vital information, CornerSentinel has the potential to significantly reduce the number of accidents and prevent countless tragedies.By addressing these areas for improvement, CornerSentinel can evolve into an even more robust and reliable safety system. Imagine a future where steep corners no longer hold the threat of unseen dangers. CornerSentinel, through its unwavering vigilance and timely warnings, can pave the way for safer roads and a future where every journey is a journey home.

6.2 Future Scope

The potential for further optimization opens a path towards even greater effectiveness. Future advancements can focus on refining the system's core functionalities. Utilizing newer, more optimized versions of YOLO, the AI engine driving object detection, can enhance accuracy and efficiency. A complete rewrite of the program code in languages like Rust or C++

offers opportunities for improved performance. While Python offers its advantages, languages like Rust and C++ can provide a more streamlined foundation for real-time applications. Additionally, the current Li-Fi circuitry presents an exciting opportunity for improvement. Implementing advanced filtering and noise reduction techniques on the receiver side will ensure clearer signal reception, especially in challenging weather conditions. Furthermore, exploring more optimized message encoding algorithms can further enhance data transmission efficiency.

References

[1] Z. Wang, J. Zhan, C. Duan, X. Guan, P. Lu, and K. Yang, “A Review of Vehicle Detection Techniques for Intelligent Vehicles,” IEEE Transactions on Neural Networks and Learning Systems, vol. 34, no. 8, pp. 3811–3831, Aug. 2023, doi: 10.1109/tnnls.2021.3128968.

[2] A. Gomaa, T. Minematsu, M. M. Abdelwahab, M. Abo-Zahhad, and R. Taniguchi, “Faster CNN- based vehicle detection and counting strategy for fixed camera scenes,” Multimedia Tools and Applications, vol. 81, no. 18, pp. 25443–25471, Mar. 2022, doi: 10.1007/s11042-022-12370-9.

[3] Goswami, P. and Shukla, M.K. (2017) ”Design of a Li-Fi Transceiver” . Wireless Engineering and Technology, 8, 71-86. https://doi.org/10.4236/wet.2017.84006

[4] S. Saranya, M. Sriram, K. Varadharaj, and K. Yugesh, “Vehicle to Vehicle Communication Using LI-FI,” 2023 2nd International Conference on Advancements in Electrical, Electronics, Communication, Computing and Automation (ICAECA), Jun.2023, doi: 10.1109/icaeca56562.2023.10199736.

[5] R. R. Samantaray, S. Banu, A. Azeez, B. CN, Kannan, and J. D, “Vehicle-to-Vehicle Communication Using RF Technology,” 2023 International Conference on Smart Systems for applications in Electrical Sciences (ICSSES), Jul. 2023, doi: 10.1109/icsses58299.2023.10199726.

[6] M. Letavay, M. Bažant, and P. Tuček, “Object Detection Algorithms - A Review,” 2023 International Conference on Control, Artificial Intelligence, Robotics & Optimization (ICCAIRO), Apr. 2023, doi: 10.1109/iccairo58903.2023.00014.

[7] J. Terven, D.-M. Córdova-Esparza, and J.-A. Romero-González, “A Comprehensive Review of YOLO Architectures in Computer Vision: From YOLOv1 to YOLOv8 and YOLO-NAS,” Machine Learning and Knowledge Extraction, vol. 5, no. 4, pp. 1680–1716, Nov. 2023, doi: 10.3390/make5040083.

[8] L. Nkuzo, M. Sibiya, and E. Markus, “Computer Vision-based Applications in Modern Cars for safety purposes: A Systematic Literature Review,” 2023 Conference on Information Communications Technology and Society (ICTAS), Mar. 2023, doi: 10.1109/ictas56421.2023.10082722.

[9] S. Maity, A. Bhattacharyya, P. K. Singh, M. Kumar, and R. Sarkar, “Last Decade in Vehicle Detection and Classification: A Comprehensive Survey,” Archives of Computational Methods in Engineering, vol. 29, no. 7, pp. 5259–5296, Jun. 2022, doi:10.1007/s11831-022-09764-1.

[10] K. H. Rahouma, M. Mohammed, and K. S. Salih, “Design and Implementation of Light Fidelity Communication Systems,” Journal of Advanced Engineering and Technology, vol. 42, no. 1, pp. 173-180, Jul. 2023, doi: 10.21608/jaet.2021.146260

[11] Güray Yıldırım, Özgür Özen, Heba Yüksel, M Naci İnci, “A Low-Cost Li-Fi Communication Setup”, Bogazici University, Physics Department, Istanbul, Turkey.

[12] Wang, Z., Tsonev, D., Videv, S., Haas, H., & Xu, Z. (2015, September). On the design of a solar-panel receiver for optical wireless communications with simultaneous energy harvesting. In 2015 IEEE International Conference on Communications (ICC) (pp. 3684-3689). IEEE.

APPENDIX

Appendix A

Achievements

Preliminary Results has been presented at RTMSEE-24 CONFERENCE (AIP Conference Proceeding), held at Kristu Jayanti College, Bangalore and paper publication is pending.

Major results has been submitted to the following IEEE conferences:

- 4th IEEE International Conference on Signal Processing, Informatics, Communication and Energy Systems 2024 (IEEE SPICES 2024), organised by Indian Institute of Information Technology Kottayam, Kerala
- 7th International Conference on Circuit Power and Computing Technologies (ICCPCT-2024), organized by Baselios Mathews II College of Engineering, Kollam, Kerala.

Appendix B

Source Code

The Generalised Main Inference Code has been given below:

```
from collections import defaultdict
import cv2
import time
from ultralytics import YOLO
from lanedetector import *
import numpy as np
from send import *
model = YOLO('train3/weights/best.pt')
video_path = "test_images/1.mp4"
cap = cv2.VideoCapture(video_path)
VIDEO_FPS = int(cap.get(cv2.CAP_PROP_FPS))
FACTOR_KM = 3.6
LATENCY_FPS = VIDEO_FPS / 2
ld = LineDrawerGUI(video_path)
line_coords = ld.line_coords
option_val = ld.option_val
ld2 = LaneDetector(video_path, line_coords)
ld2.calculate_all_points()
points = ld2.points
```

```
def calculate_distance(p1, p2):
return np.sqrt((p2[0] - p1[0]) ** 2 + (p2[1] - p1[1]) ** 2)
def calculate_speed(distances, factor_km, latency_fps):
if len(distances) <= 1:
return 0.0
average_speed = (np.mean(distances) * factor_km) / latency_fps
    * 10
return average_speed
def generate_binary_code(class_id, speed, is_stationary,
    is_wrong_side):
binary_code = ['0'] * 8
binary_code[0] = '1' if is_stationary else '0'
binary_code[1] = '0'
if class_id == 0:
binary_code[2:5] = '100'
elif class_id in [2, 6, 4]:
binary_code[2:5] = '010'
elif class_id in [5, 1]:
binary_code[2:5] = '011'
elif class_id == 3:
binary_code[2:5] = '001'
else:
binary_code[2:5] = '000'
binary_code[5] = '1' if is_wrong_side else '0'
if speed > 60:
binary_code[6:8] = '11'
elif 40 <= speed < 60:
binary_code[6:8] = '10'
elif 1.5 <= speed < 40:
binary_code[6:8] = '01'
```

```
else:
binary_code[6:8] = '00'
return ''.join(binary_code)
def display_warning_message(frame, binary_code):
warning_message = f"Warning: {binary_code}"
cv2.putText(frame, warning_message, (10, 30), cv2.
    FONT_HERSHEY_SIMPLEX, 1, (255, 0, 0), 2)
track_history = defaultdict(list)
stationary_timers = defaultdict(float)
frame_counter = 0
prev_frame = None
prev_pts = None
speed = 0
prev_track_id = None
prev_binary_code = None
is_wrong_side = False
is_stationary = False
while cap.isOpened():
ret = cap.grab()
if ret:
success, frame = cap.retrieve()
if frame_counter % 2 == 0:
results = model.track(frame, persist=True, tracker='bytetrack.
    yaml', imgsz=320, conf=0.20, int8=True)
annotated_frame = results[0].plot()
if results[0].boxes.id is not None:
boxes = results[0].boxes.xywh.cpu().numpy().astype(int)
class_id = results[0].boxes.cls.cpu().numpy().astype(int)
track_ids = results[0].boxes.id.cpu().numpy().astype(int)
for i, box in enumerate(boxes):
```

```
x, y, w, h = box
xmin, ymin, xmax, ymax = x, y, x + w, y + h
track_id = i if (len(track_ids) == 0 or frame_counter == 0)
   else track_ids[i]
track = track_history[track_ids[i]]
track.append((float(x + w / 2), float(y + h / 2)))
if len(track) >= 2 and track[-2][1] < track[-1][1]:
distances = [calculate_distance(track[j], track[j + 1]) for j
   in range(len(track) - 1)]
is_stationary = speed < 0.5
stationary_timers[track_ids[i]] = time.time() if not
   is_stationary else stationary_timers[
track_ids[i]]
if time.time() - stationary_timers[track_ids[i]] > 10.0:
is_stationary = True
binary_code = generate_binary_code(class_id[i], speed,
   is_stationary, is_wrong_side)
vehicle_pos = calculate_centroid(xmin, ymin, xmax, ymax)
correct_lane = lane_detector(points, vehicle_pos, int(
   option_val))
is_wrong_side = correct_lane != 1.0 and correct_lane != 0
speed = calculate_speed(distances, FACTOR_KM, LATENCY_FPS)
binary_code = generate_binary_code(class_id[i], speed,
   is_stationary, is_wrong_side)
if track_ids[i] != prev_track_id and binary_code !=
   prev_binary_code:
transmit_message(binary_code)
prev_binary_code = binary_code
display_warning_message(annotated_frame, binary_code)
```

```
cv2.putText(annotated_frame, f"Speed: {speed:.2f} km/h", (int(x
    ), int(y) + 10),
cv2.FONT_HERSHEY_SIMPLEX, 0.5, (255, 0, 0), 2)
roi = frame[int(y):int(y + h), int(x):int(x + w)]
if prev_frame is not None and prev_pts is not None:
prev_frame_resized = cv2.resize(prev_frame, (roi.shape[1], roi.
    shape[0]))
flow = cv2.calcOpticalFlowPyrLK(prev_frame_resized, roi,
    prev_pts, None, winSize=(15, 15), maxLevel=2)
flow_distances = np.sqrt(np.sum((prev_pts - flow[0]) ** 2, axis
    =2))
good_pts = flow_distances > 0.5
for j, is_good in enumerate(good_pts):
if is_good:
x1, y1 = prev_pts[j].astype(int).ravel()
x2, y2 = (x + flow[0][j][0], y + flow[0][j][1]).astype(int)
cv2.line(annotated_frame, (x1, y1), (x2, y2), (0, 255, 0), 2)
prev_frame = roi
prev_pts = np.array([[(int(w / 2), int(h / 2))]], dtype=np.
    float32)
frame_counter += 1
cv2.imshow("Frame", annotated_frame)
if cv2.waitKey(1) & 0xFF == ord("q"):
break
cap.release()
cv2.destroyAllWindows()
```

Rest of the Code is available at:

https://github.com/roguehunter7/CornerSentinal

Appendix C

Similarity Report

turnitin

Similarity Report ID: oid:10159:58886373

PAPER NAME	AUTHOR
CornerSentinel_Project_Report (5).pdf	sidharth

WORD COUNT	CHARACTER COUNT
7516 Words	44823 Characters
PAGE COUNT	**FILE SIZE**
47 Pages	4.3MB
SUBMISSION DATE	**REPORT DATE**
May 8, 2024 2:42 PM GMT+5:30	May 8, 2024 2:42 PM GMT+5:30

21% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

- 19% Internet database
- 9% Publications database
- Crossref database
- Crossref Posted Content database
- 7% Submitted Works database