2025 Volume: 01 Issue: 07 www.ijrpa.com ISSN 2456-9995

Research Article

International Journal Research Publication Analysis

Page: 01-12

A REAL-TIME AI MODEL FOR EMERGENCY VEHICLE DETECTION AND PRIORITY ALLOCATION

Basila P K*[1] , Basil Varghese[2] , Geo Mathew[3] , Jaijith A[4] , Mr.Surendranath Gowda D C[5], Dr Krishna Kumar P R[6]

[1,2,3,4]Students , Dept of CSE , S.E.A College of engineering and Technology, India.

[5,6] Faculty, Dept of CSE , S.E.A College of engineering and Technology, India.

Article Received: 2 November 2025

Article Revised: 22 November 2025

Published on: 12 December 2025

***Corresponding Author: Basila P K**

Students , Dept of CSE , S.E.A College of engineering and Technology, India..

DOI: https://doi-doi.org/101555/ijrpa.3233

ABSTRACT

Efficient emergency vehicle movement is crucial for saving lives, yet urban traffic congestion often delays response times. This study presents **A Real-Time AI Model for Emergency Vehicle Detection and Priority Allocation**, designed to enhance emergency mobility in smart city environments. The proposed system employs deep learning–based computer vision techniques to accurately detect emergency vehicles from live traffic camera feeds and classify them in real time. Upon detection, an intelligent traffic signal controller dynamically allocates priority through adaptive signal timing, route optimization, and immediate clearance of traffic lanes.The model integrates convolutional neural networks (CNNs) for vehicle recognition, along with a rule-based decision engine for traffic signal preemption. Experimental results demonstrate high detection accuracy and significant reductions in emergency vehicle waiting time at intersections. By automating the identification and prioritization process, the proposed AI-driven system improves emergency response efficiency, reduces congestion, and supports the development of smarter and safer urban traffic infrastructures. This work highlights the potential of real-time AI systems to transform traditional traffic management and enhance emergency service delivery

KEYWORDS: Adaptive traffic system, Emergency vehicles, Real-time traffic management, Intelligent traffic control, Traffic signal automation.

1. INTRODUCTION

One of the numerous issues we deal with in modern life is the increasing hazard of traffic congestion. Several issues, including traffic accidents and congestion, emerged because of the increase in traffic. Traffic load delays are caused by a variety of variables, such as the weather, the time of day, the season, and unanticipated events like accidents or construction projects [12]. The maximum number of vehicles that can pass on a road without getting into an accident or blocking traffic is limited bya traffic control system [2]. There is a set time delay pattern for traffic signals in our nation, Sultanate of Oman. Only light traffic is appropriate for the timer-based traffic system, which processes data at predetermined intervals. To reduce traffic congestion, a longer green light is needed in a busier lane than the others. An adaptive system with traffic density management capabilities is necessary for systems with high traffic volumes [1,11]. It may be difficult for a general traffic control system to identify emergency circumstances and high-priority situations. This highlights the requirement for an intelligent traffic management system that can manage every task and make decisions on its own.

Government authorities used to monitor and regulate traffic management systems manually, but as artificial intelligence and machine learning have grown in popularity, these systems are increasingly becoming automated. Finds trends in the given data, lowers the likelihood of accidents happening again, and manages traffic signal systems. One of the most useful solutions provided by artificial intelligence is smart city technology, which includes traffic control and parking management systems. Artificial intelligence is quickly changing the world around us. Artificial Intelligence (AI) may be used to correctly predict the movement of people, vehicles, and items at different sites in interconnected transportation networks [10].

This project proposes an AI-based smart adaptive traffic system that prioritizes emergency vehicles. It gives emergency vehicles top priority in a signal unit. This project's completion will ensure that no lives are lost because of traffic. The main goal is to use artificial intelligence and image processing to monitor ambulances so that they can cross-traffic signals first when there is congestion at the junction of signals. The system counts the number of vehicles at a signaling point and adapts the timing automatically.

The remaining part of this article is structured as follows: *Section 2* details the literature review, *section 3* gives the materials and methods used in the system, and *section 4* provides a detailed analysis of the proposed system performance.

2. LITERATURE REVIEW

Congestion in urban areas has grown to be a serious problem, especially for emergency vehicles like fire engines and ambulances [6]. Reducing delays and saving lives depend on effective traffic management systems that prioritize emergency vehicles. To solve these issues, several research have investigated the application of artificial intelligence (AI), embedded systems, and communication technologies. This section defines the research need that the suggested system attempts to fill evaluates the body of current literature and points out its shortcomings.

For real-time traffic management solutions, the use of embedded systems—like Arduino microcontrollers—has grown in popularity. The viability of employing Arduino for decentralized traffic management was shown by Sharma et al. [3]. Effective signal control was made possible by the inexpensive microcontroller, negating the need for costly infrastructure modifications. The system's versatility was further increased by the successful integration of the SIM808 module for real-time data transfer.

For traffic control, ZigBee technology provides a low-power, short-range communication option. An Arduino-based system with Zigbee modules enabled manual intervention to prioritize emergency cars by turning traffic signals to green in a Projects Factory (2021) project. The system's low communication range, susceptibility to interference, and the difficulty of configuring Zigbee-compliant devices were major drawbacks despite its effectiveness in controlled settings [4].

Gunda (2018) gave an example of how RFID technology may be used for intelligent traffic control. Emergency vehicles equipped with RFID tags were able to modify their signals when they were detected automatically. Although technology successfully reduced the need for human involvement, privacy and security issues were brought up. Widespread adoption was further hampered by high implementation costs and signal interference from metal surfaces [9].Using the SIM808 module, Maleki et al. demonstrated the effectiveness of GPS and GSM-based systems. The technology allowed for smooth real-time communication, allowing traffic signals to be adjusted in response to emergency vehicle locations. This system was appropriate for outside traffic management because of GPS's precise location monitoring and GSM's wide coverage. Nevertheless, the study lacked a thorough evaluation of microcontroller integration and real- time traffic light control techniques [13].

A thorough review points out the importance of a smart emergency vehicle prioritized adaptive traffic control system.

3. MATERIALS AND METHODS

Figure 1 displays the proposed system block diagram. The system consists of a camera-embedded ESP32-CAM that is used for capturing images of different types of vehicles on the road, especially for Emergency Vehicles. On the other hand, we have a traffic light system interfaced with an Arduino UNO microcontroller. The ESP32-CAM captures images of the vehicles and then uses an AI-based model to detect emergency vehicles (ambulances). In the lane that has an ambulance, the signal will change to green automatically, and other lanes will be red. After the emergency vehicles pass the road, the system will return to normal mode.

System Architecture

Hardware and AI-based software are combined in the proposed intelligent traffic management system to prioritize emergency vehicles and dynamically manage traffic flow. The following are its main constituents;

ESP32-CAM Camera Module

The ESP32-CAM is a low-cost microcontroller that has a built- in OV2640 camera module that can take sharp pictures. Because it has extra GPIO pins for peripheral connections, it's perfect for embedded AI and Internet of Things applications. Important features are recording traffic photos in real-time with a maximum pixel resolution of 1600x1200.MicroSD storage is supported for storing model data or pictures that have been taken. Bluetooth and Wi-Fi are built-in for smooth data transfer [6].

The ESP32-CAM will continuously take pictures of the intersection's traffic. Identifies emergency vehicles, such as fire trucks and ambulances, using an AI model installed on the ESP32-CAM and uses serial connectivity to send detection findings to the Arduino UNO traffic management system.

Traffic Light Control System

At a four-way junction, the traffic lights are controlled by the traffic light control system. Four LED-based traffic light modules are controlled by an Arduino UNO microcontroller. To process the results of AI detection, the Arduino UNO Microcontroller coordinates with the ESP32-CAM and controls traffic signal operations. LED modules for miniature traffic lights can replicate red, yellow, and green signals for four lanes [5,8].

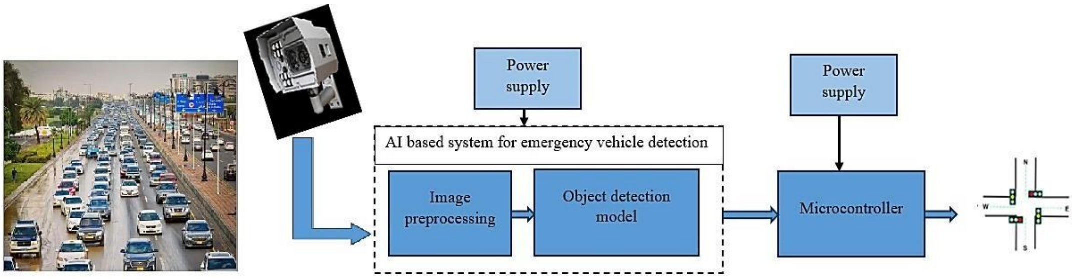

Figure 1. Proposed system process-flow diagram

Edge Impulse AI Model

pins 5,6 and 7 at traffic light 3(Green, Yellow, Red). In Road 4,

Object detection models are trained and deployed on embedded devices using the machine learning platform Edge Impulse. Object detection models are trained and deployed on embedded devices using the machine learning platform Edge Impulse. It enables effective inference on limited hardware, such as the ESP32-CAM, by supporting TinyML models and provides neural network training pipelines and integrated image preprocessing. Enhances models for object recognition in real- time [7,14].

A collection of tagged photos of ambulances, fire engines, and other vehicles is used to train the AI model. The following are part of the training process;

- Data collection: taking pictures of traffic and identifying them.
- Training Models: Developing a MobileNetV2 model for object identification using the FOMO (Faster Objects, More Objects) paradigm.
- Model Deployment: Using the ESP32-CAM to recognize emergency vehicles in real-time after training a model. Gives the Arduino UNO the detection output (the presence of an emergency vehicle) so that it may make decisions.

FTDI Programmer

An essential tool for interacting with devices that employ FTDI (Future Technology Devices International) chips is the FTDI Programmer. These chips are intended to make it easier for microcontrollers, serial communication devices, and other integrated circuits to connect over USB. Between a computer's USB port and a microcontroller or other circuit's serial interface, the FTDI Programmer serves as a bridge. Using a serial bootloader to program microcontrollers [15].

Circuit Diagram

3.2.1Circuit Diagram for Traffic Light system

Figure 2 shows a 4-way traffic light system using an Arduino (UNO) microcontroller. The system uses mini traffic light modules to represent the traffic light system. Attach each traffic light cathode to the Arduino’s GND pin. In Road 1, connect pins 11,12 and 13 at traffic light 1(Green, Yellow, Red). In Road 2, connect pins 8,9 and 10 at traffic light 2(Green, Yellow, Red). In Road 3, connectconnect pins 2,3 and 4 at traffic light 3(Green, Yellow, Red).

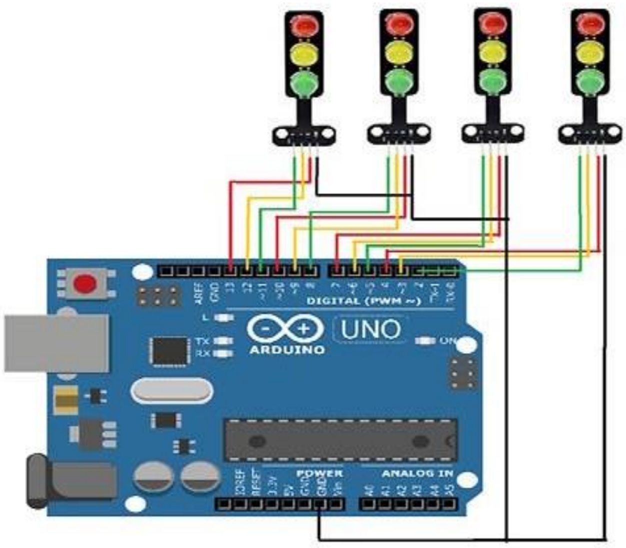

Figure 2. Connection diagram of a 4-way traffic light system.

3.2.2. Circuit Diagram for ESP32-CAM and Traffic Light System

Figure 3 shows the ESP32-CAM interfaced with a microcontroller.
This circuit includes an ESP32-CAM for AI-based monitoring and links an Arduino Uno to four traffic light modules, each of which has red, yellow, and green LEDs. Arduino controls the LEDs through its digital pins. Serial connections (TX to RX and RX to TX) are used for communication between the Arduino and ESP32-CAM.

3.2.3. ESP32-CAM and FTDI Programmer Connection *Figure 4* shows the programming mode of the FTDI programmer connected with ESP32-CAM to upload the code. Connect VCC of FTDI with 5v of ESP32-CAM, TX of FTDI with RX of ESP32-CAM, RX of FTDI with RX of ESP32-CAM and ground of FTDI with ground of ESP32-CAM. The important thing to upload code, GPIO 0 must be linked to GND.

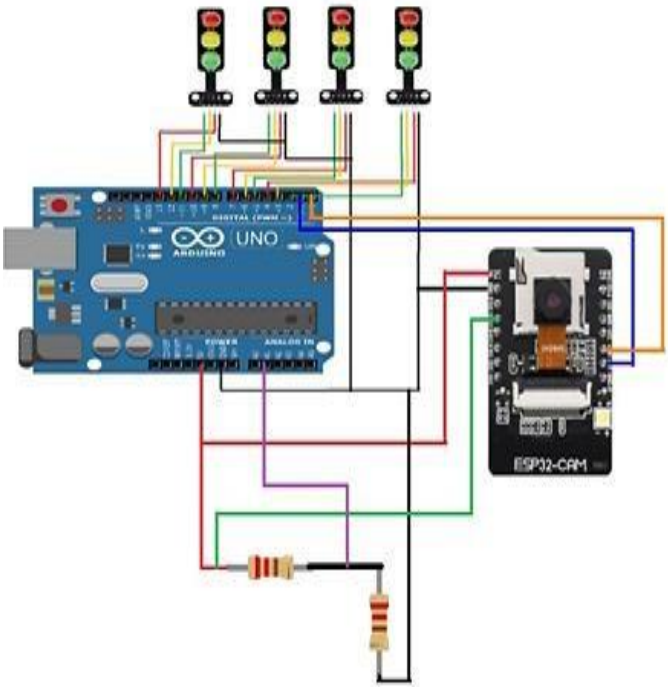

Figure 3. ESP32-CAM interface with Arduino board.

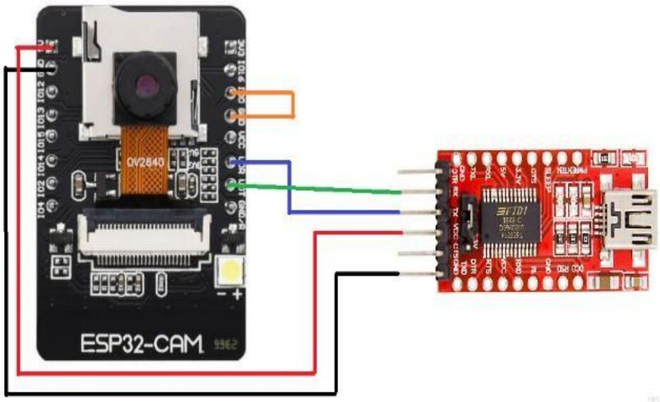

Figure 4. FTDI Programming mode.

Flowchart

Figure 5 shows the procedure for deploying an AI model for real-time inference on an ESP32-CAM using edge impulse. The first step is data collection using the attached camera module. Data preprocessing, which entails cleaning and getting the data ready for model training, comes next when sufficient data has been gathered. The next step is model training and validation, during which the pre-processed data is used to train the AI model, and its performance is verified. The model goes through an optimization process to improve performance if the required accuracy is not achieved. Real-time inference is made possible by the model's deployment to the ESP32-CAM once it has been tuned. This allows the model to interpret real-time data and provide predictions. Once the model is working on the device, this step is over.

The system captures the live images in the signal unit. Under normal conditions, the system follows a fixed delay system. If the system detects the presence of any emergency vehicle, the signal in that lane change to green automatically, and other lanes will be red. After the Emergency Vehicles pass the road, the system will return to normal mode.

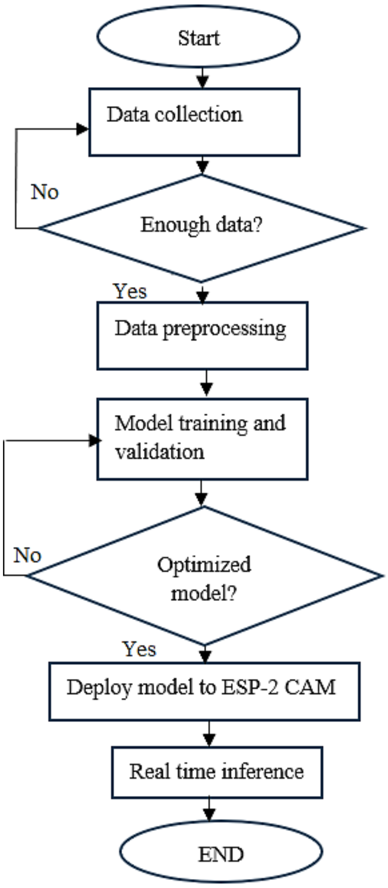

Figure 5. Data flow diagram of deploying an AI on an ESP32-CAM.

4.RESULT AND DISCUSSION

The effectiveness of the recommended intelligent traffic management system in detecting emergency vehicles and dynamically managing traffic signals was tested in a variety of scenarios, and the results are evaluated based on system performance, emergency vehicle detection, and traffic light control efficiency.

Traffic Light System

Before integrating AI-based emergency detection, the traffic lights were tested in a standard fixed-time sequence. The Arduino UNO microcontroller was programmed to control the 4-

way traffic light system based on emergency vehicle recognition. *Figure 6* shows the traffic light output. The system was set up as follows:

- Each lane was sequentially assigned a green light for 30 seconds.
- The yellow light duration is set to 5 seconds before turning red.
- The red lights in the other three lanes remained active for 30 seconds, while the green light was in one lane.

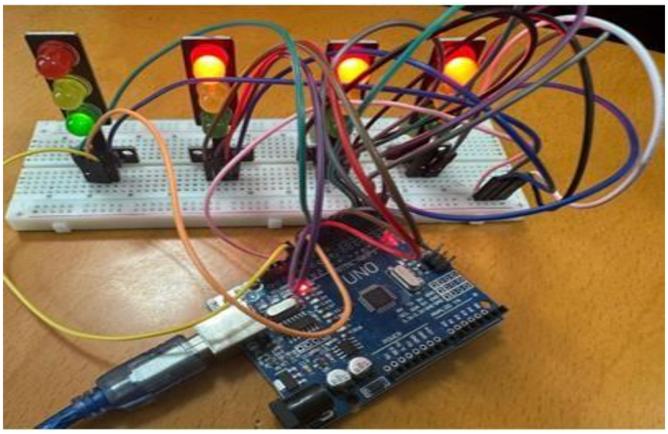

Figure 6. Fixed time delay traffic system.

When operating normally, the Arduino-controlled traffic lights operated according to the pre-programmed time without any problems.

4.2 ESP32-CAM Detecting Vehicles

Figure 7 shows data collection set up for the proposed system. Then save all the images on a Micro-SD Card and insert the Card in ESP32-CAM.

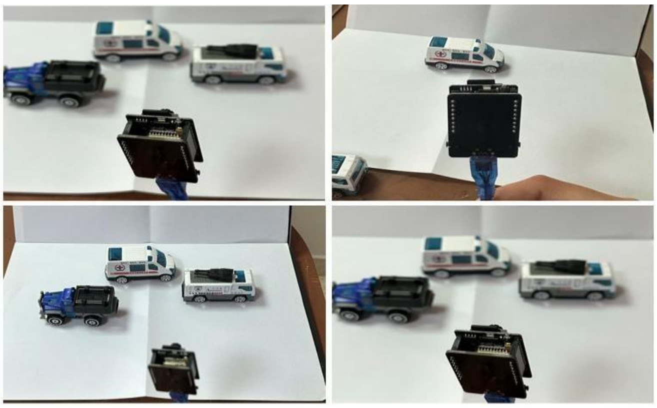

Figure 7. Image data collection.

Figure 8 shows the labelled data . Here, we put the labels for each photo of the vehicles after uploading all photos in the edge impulse.

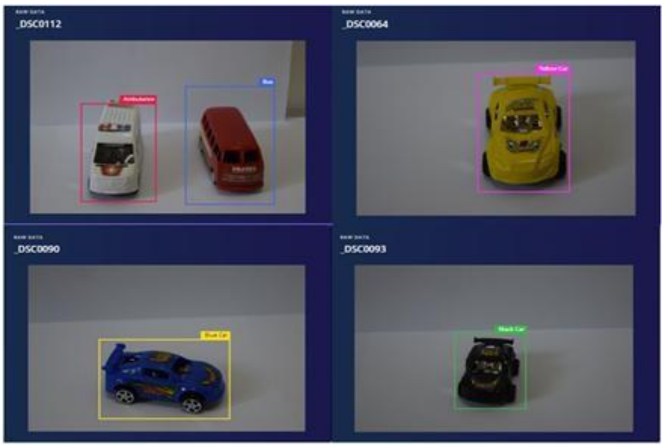

Figure 8. Data labelling.

In Neural Network settings, we put the number of training cycles as 30, the learning rate is 0.005 and the Training processor is CPU.

Figure 9 shows the results of detecting and counting different types of vehicles, such as police cars, ambulances, fire & rescue vehicles, and cars.

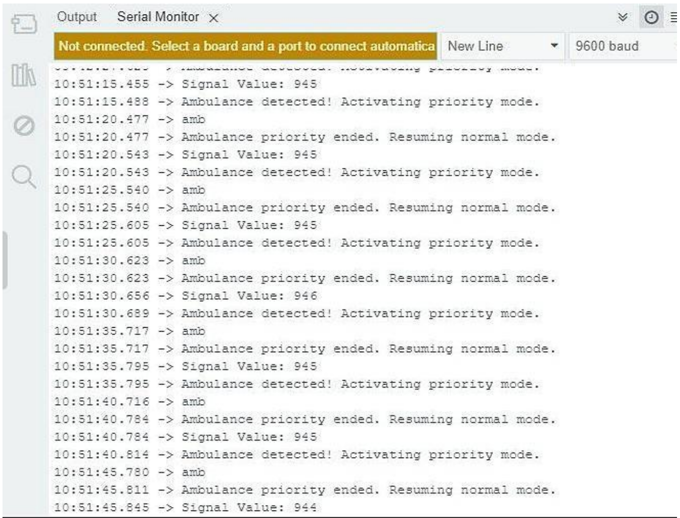

Figure 9. Outcomes of the proposed method on Serial monitor.

5. CONCLUSION

Using AI and image processing, the intelligent traffic management system effectively illustrated a sophisticated, automated method of controlling urban traffic congestion while giving priority to emergency vehicles. The project successfully solved the drawbacks of traditional fixed-timer traffic control systems by combining an Arduino-controlled traffic light system, an AI-based object identification model developed with Edge Impulse, and an ESP32-CAM for real-time picture collection.

Nevertheless, many difficulties and restrictions were noted, such as the requirement for further real-world validation, the ESP32-CAM's limited processing capability, and the sporadic misclassification in low light. Future improvements involve using IoT and cloud-based monitoring for centralized traffic control, deploying multiple cameras for greater coverage, incorporating traffic density analysis for dynamic signal timing, and installing smart road display panels to warn drivers of oncoming emergency vehicles. This system may be the foundation of the future generation of smart city traffic infrastructure with additional improvements and practical implementation, offering a productive, AI-powered substitute for conventional traffic management systems.

REFERENCES

1. X. Zou et al., “The effects of weather factors on road traffic casualties:X. Zou et al., “The effects of weather factors on road traffic casualties: Analysis on provincial panel data of China from 2006 to 2021,” Heliyon, vol. 10, no. 17, p. e36788, 2024, doi: https://doi.org/10.1016/j.heliyon.2024.e36788.
2. P. Rosayyan, J. Paul, S. Subramaniam, and S. I. Ganesan, “An optimal control strategy for emergency vehicle priority system in smart cities using edge computing and IOT sensors,” Meas. Sensors, vol. 26, p. 100697, 2023, doi: https://doi.org/10.1016/j.measen.2023.100697.
3. Sharma, R., & Gupta, K. (2022). Embedded AI for Traffic Monitoring Using ESP32-CAM. Journal of IoT and Embedded Systems, 6(2).
4. Projects Factory. (2021). Zigbee-Based Traffic Ambulance Signal Control. Retrieved from https://projectsfactory.in/product/zigbee-based- traffic-ambulance-signal-control.
5. Arduino. (2021). Arduino UNO Rev3 Specifications. Retrieved from https://store.arduino.cc/products/arduino-uno-rev3
6. Santos, S., & Santos, S. (2020). ESP32-CAM Video Streaming and Face Recognition with Arduino IDE. Random Nerd Tutorials. Retrieved from https://randomnerdtutorials.com/esp32-cam-video-streaming-face- recognition-arduino-ide/.
7. Rust, E. (2020). Getting Started with Edge Impulse. Edge Impulse Blog. Retrieved from https://www.edgeimpulse.com/blog/getting-started-with-edge-impulse
8. Mybotic. (2019). Mini Traffic Light LED Display Module Tutorial. Instructables. Retrieved from https://www.instructables.com/LED-
9. Traffic-Light-Module-Using-Arduino-UNO/ Gunda, H. (2018). Smart Traffic Management

System Using Arduino and RFID Tags. Journal of Current Research in Technology, 6(2).

10. Y. -R. Chen, Keng-Pin Chen and Pao-Ann Hsiungy, "Dynamic traffic light optimization and Control System using model-predictive control method," 2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC), Rio de Janeiro, Brazil, 2016, pp. 2366- 2371, doi: 10.1109/ITSC.2016.7795937.
11. R. Sundar, S. Hebbar and V. Golla, "Implementing Intelligent Traffic Control System for Congestion Control, Ambulance Clearance, and Stolen Vehicle Detection," in IEEE Sensors Journal, vol. 15, no. 2, pp. 1109-1113, Feb. 2015, doi: 10.1109/JSEN.2014.2360288.
12. M. F. Rachmadi et al., "Adaptive traffic signal control system using camera sensor and embedded system," TENCON 2011 - 2011 IEEE Region 10 Conference, Bali, Indonesia, 2011, pp. 1261-1265, doi: 10.1109/TENCON.2011.6129009.
13. Maleki, M. (n.d.). GPS Car Tracker Using Arduino and SIM808. Retrieved from https://electropeak.com/learn/gps-car-tracker-using- arduino-and-sim808.
14. Edge Impulse. (n.d.). Edge Impulse: The leading development platform for machine learning on edge devices. Retrieved from https://www.edgeimpulse.com.
15. Cirkit Designer. (n.d.). FTDI Programmer Tutorial and Pinout. Retrieved from https://docs.cirkitdesigner.com/component/ftdi-programmer.