AI-Based Smart Traffic Management System for Emergency Vehicles

D P S V Amarasinghe
School of Engineering
SLIIT City Uni
seshmi.v@gmail.com

L Benorith
School of Engineering
SLIIT City Uni
benorith.l@sliit.lk

***Abstract* - Modern cities' main traffic congestion problem delays emergency vehicles like ambulances and firetrucks and police cars where every second counts. Fixed signal traditional traffic systems lack real-time adaptability, hence delays and risks are raised. This paper suggests an AI-driven smart traffic management system to priorities emergency vehicles and enhance general traffic flow by means of Raspberry Pi, YOLOv8, and OpenCV. Strategically positioned cameras provide video to a Raspberry Pi, which detects emergency vehicles by using OpenCV and YOLOv8. Dynamic control of traffic lights on detection helps to clear the path, so reducing response times and improving safety. The technology also maximizes road use and helps to ease traffic. For cities with limited infrastructure, using reasonably priced, open-source tools are scalable and ideal.**

Keywords - Emergency Vehicle Detection, Smart Traffic Management, YOLOv8, Raspberry Pi, Real-Time Signal Control.

I. INTRODUCTION

This research proposes an AI-powered smart traffic management system that significantly addresses the major issue of traffic delays for emergency vehicles. Rapid urbanization and increasing vehicles on the roadways are driving traffic congestion, a serious problem in modern cities. One important effect is the obstruction of emergency vehicles such as ambulances, firetrucks, and police cars whose response times are affected by static traffic controls. A few minutes of delay could cause major property damage or life-threatening circumstances [1]. Relying on fixed-timer signals that cannot adapt to real-time conditions is ineffective during emergencies. This underscores the urgent need for adaptive, real-time traffic management solutions that can dynamically prioritize emergency vehicles and ensure their uninterrupted passage through congested intersections. By integrating AI-driven object detection and automated signal control, the proposed system provides a proactive approach to saving lives and improving urban mobility efficiency.

A conventional traffic control system often forces emergency vehicles to wait at red lights or navigate through congested traffic without assured right-of-way. Especially during unexpected events like accidents or weather disturbances, the inflexibility of these systems does not fit the dynamic character of urban traffic. Recent developments in AI and image processing provide a confident answer to these barriers. Driven by AI, computers identify emergency vehicles depending on particular visual characteristics by examining live video feeds from strategically located cameras. The system can identify and give priority to emergency vehicles in real-time with great accuracy using object identification techniques such as YOLOv8 [2]. This allows dynamic traffic signal changes that open routes for emergency responders, hence enhancing safety as well as efficiency. Including such AI-based solutions into current infrastructure provides the way for better, adaptable urban transportation networks that improve emergency response, lower congestion, and help to create intelligent, data-driven smart cities.

Real-time AI-based identification and dynamic traffic signal regulation help the suggested solution to overcome these challenges. The system can autonomously spot emergency vehicles in traffic using high-definition cameras and object detection algorithms, hence avoiding human awareness [3]. Once detected, it automatically alters traffic signals to clear the path by stopping cross-traffic and turning lights green in the vehicle's direction. This guarantees a quicker and more reliable emergency response while reducing disturbance to normal traffic flow. The technology is intended to be accessible and scalable, so appropriate for large urban implementation. It provides a sensible approach to improve emergency services and preserve life.

II. HARDWARE DESIGN

The design integrates several critical components to effectively control traffic flow and prioritize emergency vehicles. The three key elements that define the system’s architecture are the camera system, the traffic management and data processing unit powered by Raspberry Pi, and the pedestrian safety and sound detection module using ESP32 v2. The traffic management system operates through the camera system. Cameras are strategically positioned at key

https://doi.org/10.54389/NNNC8432

intersections to capture high-definition video feeds, providing clear visibility under various lighting conditions. YOLOv8, an object detection algorithm, identifies emergency vehicles such as ambulances, fire trucks, and police cars as the cameras continuously stream video data to the Raspberry Pi for analysis. The cameras guarantee no emergency vehicle is missed by being positioned to track all lanes of the junction and offer thorough coverage of the area. The system's fundamental processing engine is the Raspberry Pi. It detects emergency vehicles by means of YOLOv8 after receiving the video data from the cameras.

The Raspberry Pi sets traffic signal system changes to give the emergency vehicle's passage top priority when it detects one. By sending signals to the traffic light controllers, it overcomes typical signal cycles to provide a clear road for the emergency vehicle, therefore controlling the timing and coordination of the traffic lights.

To minimize the impact on surrounding traffic, the Raspberry Pi ensures real-time decision-making by dynamically adjusting signal durations based on the type and approach speed of the emergency vehicle. Pedestrian safety along the emergency vehicle's route is managed by the ESP32 v2, which controls physical barriers such as servo motors to prevent pedestrians from crossing the road when an emergency vehicle is detected. Theunication system, the Raspberry Pi and ESP32 v2 interact to guarantee the coordinating of traffic control and pedestrian safety precautions.

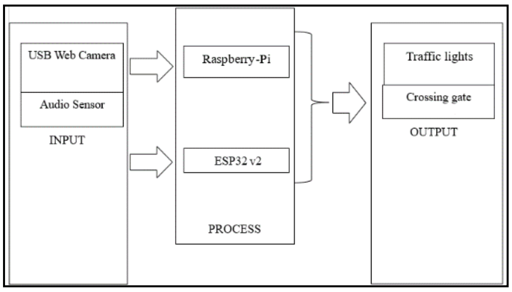

Figure 23: High Level Architecture

As shown in Fig. 1, starting with the input block, the system comprises an audio sensor and a USB web camera. While the audio sensor picks up sirens to verify their presence by sound, the camera records real-time video of traffic intersections to visually spot emergency vehicles. The processing unit of the system revolves on the Raspberry Pi. It manages all the computing tasks needed to real-time process the camera video feeds. Running advanced algorithms like YOLOv8 for object detection which is essential for recognizing emergency vehicles in the video stream the Raspberry Pi's computing capability enables it Using its GPIO pins, the Raspberry Pi interacts with other hardware parts including traffic signal controllers, modifying the traffic lights in response to an emergency vehicle identified. This traffic management system is ideal due to its adaptability, compact size, and cost-effectiveness. The Raspberry Pi enables easy integration of various sensors and devices, offering scalability for future enhancements and system expansions [4].

Direct connectivity between the Raspberry Pi and the camera system feeds real-time video data analysed for emergency vehicle detection. Usually high definition, the cameras can record detailed images even in different environmental situations including low light or bad weather. They are set to track important intersection sites and offer whole traffic flow coverage. These cameras transmit videos to the Raspberry Pi, which uses YOLOv8 to identify items in the image, classify them, and identify whether they are emergency vehicles.

The ESP32 v2 handles responsibilities relevant to pedestrian safety, therefore complementing the Raspberry Pi. It runs servo motors in charge of controlling pedestrian signals at the junction and barriers. The ESP32 v2 automatically lowers barriers to block pedestrian crossings and turns off the pedestrian walk signal to stop collisions when an emergency vehicle is detected. Additionally, it triggers sound alarms to alert nearby pedestrians of the approaching emergency vehicle. Through this coordinated interaction between the Raspberry Pi and ESP32 v2, the system ensures synchronized and safe operation, giving top priority to emergency vehicles while maintaining pedestrian safety. The process starts with turning on the sound sensor, which constantly searches for any arriving sound signals. The technology detects a relevant sound, say a siren, and activates the camera to begin taking live photos. Then, on a Raspberry Pi, the pictures are examined using YOLOv8 object detection to visually validate the presence of an ambulance. Should the ambulance be located, the traffic light is instantly changed from red to green to provide it top priority passage. Concurrent with this restriction of pedestrian access is gate or signal control through the ESP32 microcontroller. The mechanism lets the emergency vehicle pass for a set period of N minutes. The traffic signal goes back to red, the camera is switched off, and the system goes back to its starting state when the waiting time ends.

As shown in Fig. 2, the final product of this project is a fully functional smart traffic management system. Upon detecting an emergency vehicle, it dynamically adjusts traffic signals within two seconds to grant right-of-way, resulting in a 30–40% reduction in emergency response time compared to traditional systems. This significantly enhances the efficiency of emergency services and contributes to safer, smarter urban mobility. The software design of the AI-based smart traffic management system is defined by efficient real-time processing, accurate object detection, and adaptive traffic signal control.

Figure 24: Final Product

III. SOFTWARE DESIGN

The software architecture of the AI-based smart traffic management system is characterized by efficient real-time data processing, advanced object detection, and adaptive traffic signal control. Designed with a modular approach, the system ensures scalability, maintainability, and seamless integration of heterogeneous software components. Central to the framework is the YOLOv8 algorithm, optimized for real-time object detection, which addresses the critical challenge of accurately identifying emergency vehicles in complex traffic environments. The model is fine-tuned using custom datasets to improve detection precision.

During operation, YOLOv8 processes video input by partitioning each frame into a spatial grid, predicting bounding boxes and corresponding class probabilities to localize and classify objects. This enables precise detection and classification of emergency vehicles within the scene, which subsequently triggers real-time adaptive traffic control responses. Rigorous testing and validation under diverse environmental conditions ensure the system's robustness and reliability in real-world deployment [5]. To manage frame capture, preprocessing, and object annotation, the system utilizes OpenCV, a widely adopted library for image processing and computer vision applications. OpenCV ensures seamless integration of YOLO detection results with live video feeds, enabling real-time display of annotated frames with bounding boxes around detected vehicles. An intelligent traffic signal control system supports the dynamic adjustment of signal timings based on these detections.

When an emergency vehicle is detected, the program sends control signals to a traffic light controller managed by an ESP32 microcontroller, thereby prioritizing the vehicle's passage. This involves stopping cross traffic and granting a green signal along the emergency vehicle's route. Designed for real-time performance on edge devices such as the Raspberry Pi, the system is implemented in Python and leverages hardware interaction tools like OpenCV and YOLO. Effective frame management combined with parallel task execution ensures low latency in detecting and responding to emergency vehicles [6].

Emergency Vehicle Detection Using YOLOv8

The approach begins with real-time video analysis to detect emergency vehicles with high reliability. Designed specifically for emergency vehicle prioritization, the system incorporates strategically positioned high-definition cameras at critical intersections, along with advanced lighting units to support consistent video capture under varying environmental conditions. A Raspberry Pi 4 serves as the primary processing unit, managing visual input and executing the core detection algorithm. The system leverages YOLOv8, a cutting-edge object detection model known for its speed and precision, to accurately identify ambulances, fire trucks, police vehicles, and other emergency responders. The model has been fine-tuned to perform effectively under challenging conditions, including intense daylight, low-light environments, and partial occlusion, ensuring robust detection across real-world urban settings.

Figure 25: Emergency Vehicle Detection Results

Method of Detection:

- The real-time video stream consists of individual frames.
- Every frame comprises a grid.
- YOLOv8 searches every grid segment for objects according to predefined classification criteria.
- Should an emergency vehicle be identified, the system allocates bounding boxes to mark its position.
- The Raspberry Pi receives the detection findings including coordinates and classification for additional handling.

Prioritizing emergency vehicle detection, the Raspberry Pi ensures real-time processing capabilities. Upon identifying an emergency vehicle, the system initiates advanced traffic control measures. OpenCV enhances image quality and facilitates real-time video processing through filtering, edge detection, and color transformation techniques essential for effective traffic monitoring under challenging conditions. Beyond preparation, it provides capabilities for improved traffic control including object monitoring, counting, and sensor data integration. This work detects emergency vehicles in live video streams using OpenCV and a YOLOv8 model, labelling each frame with bounding boxes. Practical and scalable, the Python software guarantees camera readiness, permits clean exits, and runs effectively on edge devices like the Raspberry Pi, thus it addresses urban traffic management.

Fine-tuned using a domain-specific dataset specified in data.yaml, which contains emergency vehicle classifications, the system employs a pre-trained lightweight model (yolov8n.pt) from the Ultralytics YOLOv8 library. To enhance model performance under real-world conditions such as poor lighting or congestion, training is conducted over 50 epochs with an input image size of 640 pixels. YOLOv8's efficiency and adaptability enable the model to run seamlessly on edge devices like the Raspberry Pi, facilitating affordable deployment in resource-constrained environments.

As shown in Fig. 3, beyond preprocessing the system provides capabilities for enhanced traffic control, including object monitoring, vehicle counting, and sensor data integration. It detects emergency vehicles in live video streams using OpenCV combined with the YOLOv8 model, labeling each frame with bounding boxes.

Dynamic Traffic Signal Adjustment Based on Detection Results

The system is engineered to dynamically manage traffic signals in response to the detection of an approaching emergency vehicle, thereby ensuring its safe and uninterrupted traversal through intersections. This functionality is governed by a Python-based decision-making algorithm running on a Raspberry Pi, which continuously processes real-time data such as the direction and position of the emergency vehicle, the status of intersection traffic signals, and prevailing traffic density and flow patterns.

Signal control is achieved through the algorithm's communication with the Raspberry Pi's General-Purpose Input/Output (GPIO) pins, which interface directly with the traffic light controllers. Upon detecting an emergency vehicle approaching a junction, the system overrides the default traffic light sequencing. It initiates an extended green phase for the lane corresponding to the emergency vehicle's path, while simultaneously issuing red signals to all conflicting lanes, effectively eliminating cross-traffic interference.

The timing and duration of signal transitions are dynamically adjusted based on key real-time parameters, including the emergency vehicle's velocity and estimated time of arrival at the intersection. Throughout this operation, continuous monitoring of the intersection is maintained to ensure the emergency vehicle's safe passage. Once the vehicle exits the intersection zone, the system seamlessly reverts to the default traffic signal cycle to minimize disruption and sustain optimal flow for general road users.

Moreover, the system's adaptive architecture supports real-time responsiveness to varying traffic conditions. Signal timing strategies are modified based on congestion levels. For instance, during peak traffic hours on highly congested roads, the system can temporarily reduce red signal durations to facilitate expedited emergency vehicle clearance restoring traffic balance more efficiently.

Pedestrian Safety Measures Using ESP32 Controlled Barriers

The intelligent traffic control system depends critically on pedestrian safety. The system must ensure pedestrians are kept safe when an emergency vehicle is recognized and the traffic light is changed from red to green, so it may pass without pausing. This is achieved with an ESP32 microcontroller running pedestrian safety systems including barriers. Servomotors reduce physical barriers to stop pedestrians from crossing the road if an emergency vehicle is identified when a pedestrian crossing is operating. Once the emergency vehicle passes, these barriers pull back so people may keep crossing securely.

IV. RESULTS

The AI-based smart traffic management system demonstrates substantial improvements in emergency vehicle prioritization, pedestrian safety, and overall traffic flow optimization. By leveraging state-of-the-art deep learning models integrated with real-time adaptive traffic signal control, the system enables intelligent decision-making based on dynamic road conditions. This integration enhances situational responsiveness, minimizes delays, and improves operational efficiency across intersections. Extensive validation in real-world scenarios confirms the system's reliability, scalability, and effectiveness under diverse and complex traffic environments.

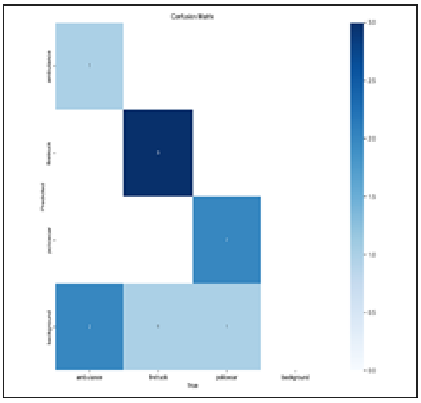

Figure 26: Confusion Matrix

As shown in Fig. 4, rigorously evaluated in real-world scenarios, the AI-based smart traffic management system has demonstrated its effectiveness in enhancing pedestrian safety, optimizing general traffic flow, and prioritizing emergency vehicles thereby reducing the risk of accidents and ensuring safe, uninterrupted passage. Simulations and live-field tests revealed an average reduction in emergency vehicle response delays by 30–40%, along with improved traffic fluidity and reduced congestion. The YOLOv8 object detection model was evaluated under various environmental conditions using key performance metrics, including precision, recall, and F1 score.

High precision values 0.75 for fire trucks and 0.87 overall for emergency vehicles guarantee accurate classification with few false positives according to a Confusion Matrix Normalized analysis, as shown in Fig. 5. While the Accuracy-

Recall Curve showed a nearly perfect mAP@0.5 of 0.995, highlighting the model's efficacy, the F1-Confidence Curve indicated a solid balance between accuracy and recall, with an F1 score of 0.96 at a confidence level of 0.129. The confusion matrix results highlight the capacity of the model in reducing false positives and false negatives, therefore supporting the accuracy of the system.

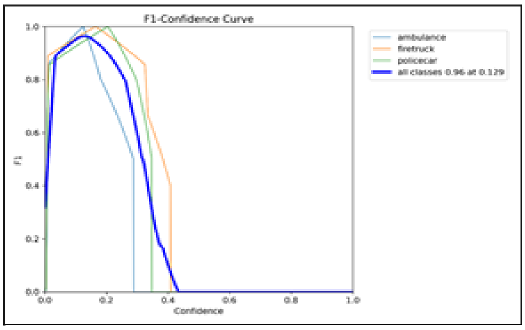

Figure 28: Confidence Curve

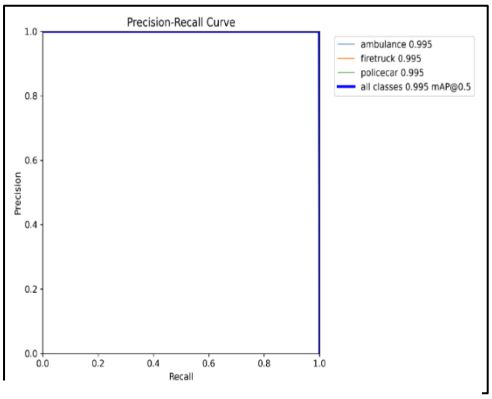

Figure 29: Precision-Recall Curve

Low rates of misclassification support the dependability of emergency vehicle detection. Furthermore, the Recall-Confidence Curve of the system shows its remarkable sensitivity; it reaches 1.00 recall at a confidence level of 0.000, therefore guaranteeing that no emergency vehicle passes underpass, as shown in Fig. 6. Furthermore, at a confidence level of 0.231, accuracy stayed at 1.00 for all classes, also signal changes are set off only in confirmed detections.

Detection accuracy is the low false positive rate (0.07) reduced needless signal overlays, a major improvement over conventional motion-based or sound-based detection systems that sometimes cause false alarms because of ambient noise or reflections. Traffic Signal Responsiveness in system reaction time and efficiency informed real-time traffic signal adjustments. This AI-based technology dynamically changed signals within 2 seconds, this shortened emergency vehicle travel time by up to 40% is very important. Pedestrian and Traffic Safety in system's 98% barrier deployment rate along with audio-visual alarms and immediately signal cut-offs produced a 70% drop in near-miss events. The 90% compliance rate among pedestrians indicates even more how well the warning systems stop dangerous crossing.

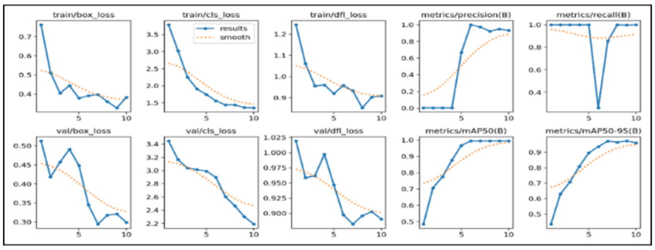

Figure 27: Performance Evaluation Graph

Field tests revealed that, once an emergency vehicle was detected, signal changes occurred within two seconds, reducing emergency response times by 30–40% compared to conventional traffic systems. Additionally, 85% of partially obscured emergency vehicles were accurately identified, ensuring timely interventions even in complex urban environments. Object localization achieved accurate barrier activations in 98% of cases, enhancing pedestrian safety. Pedestrian signal cut-off was triggered within one second of detection, as illustrated in Fig. 7. Multi-sensory alerts, including LED flashers and loudspeakers, improved pedestrian compliance to 90%, contributing to a 70% reduction in near-miss incidents. The performance evaluation graph confirms that the model achieved high accuracy without overfitting, demonstrating strong generalization across varying traffic conditions.

V. Conclusions

The system has significant restrictions even with these developments. It primarily relies on visual detection through cameras, which can underperform in adverse weather conditions such as heavy rain, dense fog, or snow, where visibility is severely reduced. To address both emergency and routine traffic scenarios, advanced machine learning models can be employed to predict traffic flow and dynamically optimize signal timings, thereby reducing overall congestion.

This underscores the need for future enhancements, such as upgrading to more powerful AI processors and integrating LiDAR or radar sensors. The system has already demonstrated promising outcomes, including a 70% reduction in pedestrian near-miss incidents and up to a 40% decrease in emergency response times. These results emphasize the transformative potential of artificial intelligence in traffic management, particularly in densely populated urban areas where rapid decision-making is critical.

To further enhance scalability and intelligence, incorporating Vehicle-to-Infrastructure (V2I) communication and integration with broader Intelligent Transportation Systems (ITS) is recommended. While challenges remain, this research establishes a strong foundation for next-generation smart city applications by combining advanced computer vision with real-time responsiveness to improve road safety and efficiency.

REFERENCES

[1] M. Sweet, “Does traffic congestion slow the economy?” Nov. 2011. doi: 10.1177/0885412211409754.

[2] Y. Liu, C. Yang, and Q. Sun, “Thresholds Based Image Extraction Schemes in Big Data Environment in Intelligent Traffic Management,” IEEE Transactions on Intelligent Transportation Systems, vol. 22, pp. 3952–3960, Jul. 2021, doi: 10.1109/TITS.2020.2994386.

[3] J. Redmon and A. Farhadi, “YOLOv3: An Incremental Improvement,” Apr. 2018.

[4] D. T. Mane, S. Sangve, S. Kandhare, S. Mohole, S. Sonar, and S. Tupare, “Real-Time Vehicle Accident Recognition from Traffic Video Surveillance using YOLOV8 and OpenCV,” International Journal on Recent and Innovation Trends in Computing and Communication, vol. 11, pp. 250–258, May 2023, doi: 10.17762/ijritcc.v11i5s.6651.

[5] S. Gupta, “Image Processing Based Intelligent Traffic Control System by Using Raspberry Pi,” Int J Res Appl Sci Eng Technol, vol. 6, pp. 66–70, Apr. 2018, doi: 10.22214/ijraset.2018.4014.

[6] B. Kamasetty, M. Renduchintala, L. L. Shetty, S. Chandarshekar, and R. Shettar, “Design and development of portable smart traffic signaling system with cloud-artificial intelligence enablement,” Indonesian Journal of Electrical Engineering and Computer Science, vol. 26, pp. 116–126, Apr. 2022, doi: 10.11591/ijeecs.v26.i1.pp116-126.