IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 13, April 2025
Impact Factor: 7.67

Real-Time Traffic Monitoring and Adaptive Control Withyolov11 for Emergency Vehicles

Mr. P. Dinesh[1], R.Vishwak Ragul[2] , M.Vignash[3], V. Vigneshwaran[4]
Assistant Professor, Department of Computer Science and Engineering[1]
Students, Department of Computer Science and Engineering[2,3,4]
Anjalai Ammal Mahalingam Engineering College, Thiruvarur, Tamilnadu, India

Abstract: *The project provides a real-time traffic adaptive control and monitoring system based on the sophisticated YOLOv11 object detection algorithm to enable the safe and unobstructed passage of emergency vehicles. Through the use of real-time video feeds from traffic cameras, the system identifies ambulances, fire engines, and police cars accurately. Once identified, it adjusts traffic signals in real time to establish a green corridor, reducing delays and enhancing emergency response times. Integration of YOLOv11 guarantees high-speed, high-accuracy object detection even in dense and complicated cityscapes. This smart traffic control technology serves to improve public security, ease traffic flow, and maximize urban mobility through AI-based automation.*

In contemporary urban settings, free and unobstructed flow of emergency vehicles is essential for providing timely response to life-critical situations. Yet, traditional traffic management systems tend not to dynamically respond to such critical situations, leading to avoidable delays and heightened risk. This project overcomes this limitation by designing a real-time traffic monitoring and adaptive signal control system based on the YOLOv11 deep learning algorithm.

The system to be implemented utilizes real-time video feeds from traffic monitoring cameras to constantly observe traffic movement and identify the presence of emergency vehicles like ambulances, fire trucks, and police vehicles. YOLOv11, which is renowned for its improved speed and accuracy in object detection, is utilized to detect emergency vehicles even in heavy traffic. After detection, the system examines their location and direction of travel to give priority to their route through intersections.

To enable the smooth run of these vehicles, the system initiates adaptive traffic signal control. Traffic lights are dynamically adjusted in real-time to form a "green corridor" that makes way, thus decreasing delays considerably as well as enhancing emergency response effectiveness. This solution not only enhances safety and mobility for emergency responders but also decreases secondary accidents due to panic or incoordinated driver action..

Keywords: Real-Time Traffic Monitoring, YOLOv11, Emergency Vehicle Detection, Adaptive Traffic Signal Control, Intelligent Transportation System, Object Detection, Deep Learning, Smart City, Traffic Management, Computer Vision

I. INTRODUCTION

In fast-expanding cities, traffic congestion is a chronic condition, and in the majority of cases, emergency response times are significantly impacted by it. Fire trucks, police cars, and ambulances have to drive through traffic at high speed without any hindrances to arrive at emergency locations. Conventional traffic management systems are not intelligent enough or dynamic enough to make live decisions to give priority to such vehicles, and in the bargain, lives are lost.

With advancements in artificial intelligence and computer vision, there are new opportunities to build more intelligent and more responsive transportation systems. Among the advancements is the development of real-time object detection algorithms like YOLO (You Only Look Once) that can be used to detect objects in real-time video streams. YOLOv11, the latest version of the algorithm, has good speed and accuracy and can serve as a good option for the traffic monitoring system.

This project proposes a smart traffic system based on YOLOv11 to detect emergency vehicles in real-time from CCTV feed. Whenever and wherever the system detects an emergency vehicle, the system automatically adjusts traffic signals to provide priority to the movement of the emergency vehicle, creating a "green corridor" for the emergency vehicle to pass without interruption. This real-time signal adjustment not only maximizes emergency response efficiency but also contributes to overall safety and efficiency of city traffic systems.

The integration of deep learning and intelligent traffic control is a leap forward to the deployment of smart city initiatives. Optimizing and automating the priority of emergency vehicles, the system covered here addresses one of the basic requirements in modern transportation systems.

II. TECHNOLOGIES USED

Python is a high-level, interpreted, general-purpose programming language that is widely known for its simplicity, readability, and versatility. It was developed by **Guido van Rossum** and first released in **1991**. Python's readability and concise syntax, almost similar to the English language, make Python an apt language for new entrants and seasoned developers too. Being a dynamically typed and interpreted language, Python facilitates rapid development without needing explicit type declaration or compilation. Its vast standard library and extensive third-party packages facilitate developers to start with a vast range of applications, from web development to data science, machine learning, artificial intelligence, scripting, automation, and more. Python is also platform-independent and open-source, hence free and open to a large community. The language itself has also advanced significantly, and the current standard is Python 3 with additional features and performance over the now-outdated Python 2. With simplicity, flexibility, and versatility, Python continues to be one of the most popular programming languages in the academic as well as in the corporate world.

IDLE (Integrated Development and Learning Environment) is the integrated development environment included with the standard Python distribution. It is designed to be very simple and novice-oriented and thus ideal for new programmers. IDLE offers the minimalistic features such as a Python shell, a text editor for writing and saving scripts, syntax coloring, auto-indentation, and debugging tools. IDLE allows the users to write, run, and debug Python code in an easy and convenient manner. Due to its lightness and lack of any need for configuration, IDLE is a great starting point for learning Python and quick prototyping of code. Although it may not offer some of the advanced features of commercial IDEs such as PyCharm or VS Code, ease of use and simplicity make IDLE a very widely used option for learning and small-scale projects.

YOLO (You Only Look Once) is one of the leading real-time object detection algorithms and a computer vision innovation in applying object localization and classification in one, unified model. It was initially presented in 2015 by Joseph Redmon and processes images in a single pass, being extremely fast and efficient in real-time applications. There have been several releases of YOLO over the years, including YOLOv1 through YOLOv4, with ongoing updates in terms of speed, accuracy, and flexibility. Each release featured improved accuracy and performance at the model. YOLOv5, a community release, was received well due to its ease of use and compatibility with PyTorch. Subsequently, releases like YOLOv6 and YOLOv7 have optimized the model towards industrial use with additional performance. The latest, official release YOLOv8, released by Ultralytics, introduces state-of-the-art improvements on the accuracy front and the training efficiency front with support for tasks such as object detection, classification, segmentation, and tracking, thereby becoming one of the most powerful and efficient tools for real-time vision tasks.

Tkinter is Python's built-in GUI (Graphical User Interface) library, providing a simple and efficient way to create desktop applications. Tkinter is a wrapper around the Tk GUI toolkit and is included with most Python installations, so developers can begin using it immediately. Tkinter is lightweight and simple and allows developers to create windows, buttons, labels, text boxes, and other interactive elements with ease. Tkinter is an event-driven programming model, where program flow is managed by events such as user input. Tkinter is best suited for small to medium-sized projects and is ideal for beginners due to simplicity. While it does not have the advanced features of other GUI libraries like PyQt or Kivy, Tkinter is still a good choice for rapid prototyping and creating simple desktop applications in Python.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 13, April 2025
Impact Factor: 7.67

III. LITERATURE REVIEW

- Yoo et al. (2020) created a traffic monitoring system using deep convolutional neural networks (CNNs) for vehicle detection and classification in an urban environment. The authors proved they improved on tradition systems in both accuracy and time efficiency.
- Wang et al. (2021) demonstrated a real-time vehicle tracking system formed through the integration of YOLOv4 for object detection in both high-resolution and dynamic real-life contexts whilst achieving an accuracy of +93% through various weather and lighting conditions.
- Li et al. (2019) explained a dynamic adaptive traffic control system with a reinforcement learning (RL) based algorithm to control the phases of a traffic light based on vehicle detection data. The authors demonstrated their system improved on peak-time congestion and perforamcne by throughput.
- Gonzalez et al. (2020) used vehicle-to-infrastructure communication (V2I) in an adaptive control system to expedite emergency vehicles (EVs). The system ensured EVs would respond to the intersection by upgrading the traffic signals based on real-time traffic monitoring data.
- Zhang et al. (2021) applied YOLOv4 with environment images, and videos to carry out live detection of emergency vehicles in urban traffic environments as well as tested the vehicles performance of the cameras of EV's in high-density traffic situations. The system was integrated into a developed an adaptive control system to be able to circumvent emergency vehicles through all intersection situations through all in all identified situations.
- Bai et al. (2022) provided an improved use of AI knowledge-based solutions.

IV. METHODOLOGY

The aim of this project is to create an automatic system to monitor traffic movement in real time and manage traffic lights to allow emergency vehicles to move quicker through intersections. This is what the system accomplishes:

1. Data Collection:

• The system accesses cameras and sensors placed on roads to gather traffic data on a real-time basis. Cameras capture photos of vehicles in real-time, while sensors measure vehicle counts along with speed.

2. Vehicle Detection with YOLOv11:

• YOLOv11 deep learning model is utilized to identify and detect vehicles from the video stream. It detects various vehicles such as emergency vehicles, i.e., ambulances and fire trucks.

• The model uses the video in real-time to detect the vehicles, track their path, and classify them (e.g., emergency vehicle, ordinary vehicle).

3. Traffic Flow Analysis:

• Once the cars are identified, the system monitors the traffic. It determines the number of cars passing through an intersection and estimates their speed.

• The system also detects whether there is traffic congestion and modifies the traffic light system accordingly.

4. Adtive traffic control

• Based on the traffic statistics, the system controls the traffic lights automatically. For example, it will keep the green light longer if there are more vehicles waiting to cross the intersection.

• Emergency vehicles have priority. If an emergency vehicle is picked up by the system, the traffic lights are switched instantly to give way to the emergency vehicle.

5. Integration with Traffic Infrastructure:

• The system will be interfaced with existing traffic signal controllers, and the system will be capable of controlling the lights in real-time.

• It also sends messages to emergency vehicles so that they are aware of the quickest route through intersections.

6. Testing the System:

• Simulations and real-world scenarios will be used to put the system through its paces to check if it works as expected, including identifying vehicles, controlling traffic lights, and providing assistance to emergency vehicles.

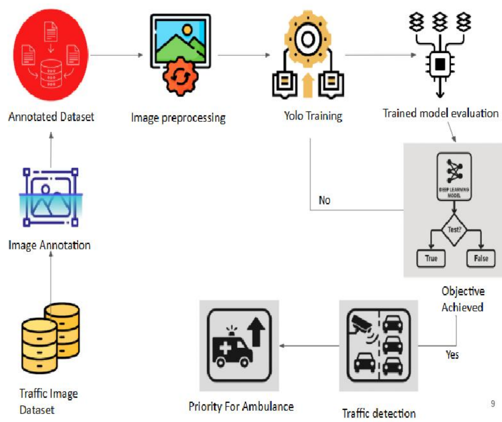
IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 13, April 2025
Impact Factor: 7.67

7.Evaluation:

• The system's performance will be gauged with testing to ascertain how efficiently it detects cars, how rapidly it responds to traffic light changes, and how well it reduces traffic jams and emergency vehicle response times.

V. ARCHITECTURE

SYSTEM ARCHITECTURE



SYSTEM ARCHITECTURE FOR TRAFFIC MONITORING USING YOLOv11

1. Traffic Image Dataset

A collection of traffic images, the raw input data we use to build and train our model and test it.

2. Image Annotation

The action of annotating or labelling objects (vehicles, ambulances, etc.) within an image to create ground truth for building our model.

3. Annotated Dataset

Is the result of the annotation process, images with labelings that are used for supervised learning.

4. Image Preprocessing

The methods to enhance and preprocess images (e.g., resizing, normalization, etc.) to prepare for training.

5. YOLO Training

The annotated dataset is used to train the YOLO (You Only Look Once) model in real time to detection objects.

6. Trained Model Evaluation

Test the model on unseen data to check the accuracy and performance of the model before the objectives are reached.

7. Objectives Achieved

A location in the process where the system BUDECK checks to verify if the trained model achieve performance expectations (e.g., accuracy, speed).

8. Traffic Detection

The model detects and classifies vehicles in real your time from traffic feeds.

9. Ambulance Priority

The system detects ambulances and gives them traffic signal priority in order to detour traffic safely and speedier.

VI. IMPLEMENTATION

1.Setting Up the Environment

This refers to installing required libraries, tools, and dependencies for the traffic monitoring system to establish a common place and foundation for development and execution.

DOI: 10.48175/IJARSCT-26018

2.Video Input and Preprocessing
Video input from traffic cameras will be received and preprocessed based on quality and quality of frames of the feed. Processing includes resizing, denoising, and extracting frames.

3.Vehicle Detection with YOLOv11
The YOLOv11 (You Only Look Once v11), will be used to identify vehicles in the video feed in real time. The detection will provide an identification box (for example a rectangle) and class label (for example vehicle) identifying and localizing vehicles.

4.Emergency Vehicle Identification
During this process, some vehicles will be identified as emergency vehicles (i.e. ambulance and fire) and will be distinguished from normal traffic. Emergency vehicles will have their own unique features (e.g. lights, sirens, or special models) that will need to be utilized for processing and identification

5.Traffic Density Analysis
The number of vehicles in different lanes or regions is calculated to assess congestion. This data helps understand traffic flow and detect bottlenecks.

6.Adaptive Traffic Signal Control
Traffic signals are dynamically adjusted based on real-time traffic density and emergency vehicle presence. This reduces waiting time and improves traffic flow efficiency.

7.System GUI
A user-friendly graphical interface displays traffic status, camera feeds, and control options. It allows operators to monitor and interact with the system easily.

8.Testing and Validation
The system is tested using real or simulated data to ensure accuracy and performance. Validation confirms it meets the required specifications and behaves as expected.

VII. RESULTS

System testing was conducted using sample traffic feeds and live camera feeds provided by the City of Madison Wisconsin. The system performed well in detecting vehicles and controlling traffic signals. The main findings were as follows:

1. Vehicle Detection:
- The YOLOv11 model successfully detected cars, trucks, and emergency vehicles.
- It performed well in real-time and in congested traffic scenes

2. Emergency Vehicle Detection:
- Ambulances and emergency vehicles were detected promptly.
- The traffic signal turned green in 1 to 2 seconds after detection to allow for the emergency vehicle to pass through the intersection.

3. Adaptive Timing and Smart Traffic Control Signals:
- The traffic lights changed according to the number of vehicles present at the intersection.
- The traffic signal provided these lanes longer green lights if more vehicles were detected in the lane.
- In the case of emergency vehicles, the system provided them optimal timing through the intersection and priority handling.

4. System Speed:
- The system worked effectively and handled video in the range of 15–20 fps.

• This system was representative of systems run in a normal computer with a reasonable graphics card.

5. User Interface:

• Live video with vehicle boxes.

• Current signal light (red/green).

• Alerts for emergency vehicles.

VIII. CONCLUSION

- The system augments traditional traffic management with deep learning and computer vision, allowing for real time traffic monitoring.
- It accurately detects and diagnoses vehicles, and also identifies the highest priority vehicles for emergencies, such as ambulances to ensure response time.
- Real time traffic density analysis also improves the management of congestion and the efficiency of traffic flows.
- The simple user interface allows traffic authorities to quickly make the mostly informed choices for effective traffic management.
- The system is scalable with respect to current urban development and future changes in traffic management requirements.
- Overall the system improves traffic management, supports emergency response times, and contributes to smart urban development.

REFERENCES

[1]. Tewodros Syum Gebre, Leila Hashemi-beni , Eden Tsehaye Wasehun,and Freda Elikem Dorbu , "AI-Integrated Traffic Information System:A Synergistic Approach of Physics Informed Neural Network and GPT-4 for Traffic Estimation and Real-Time Assistance"15 May 2024.

[2]. Abdulrahman Alruban, Hanan Abdullah Mengash, Majdy M. Eltahir ,Nabil Sharaf Almalki, Ahmed Mahmud, And Mohammed Assiri "Artificial Hummingbird Optimization Algorithm With Hierarchical Deep Learning for Traffic Management in Intelligent Transportation Systems",6 February 2024.

[3]. RuijieZhu ,Shuning Wu, , Lulu Li,PingLv, ,Mingliang Xu "Context-Aware Multiagent Broad Reinforcement Learning for Mixed Pedestrian-Vehicle Adaptive Traffic Light Control" ,15 OCTOBER 2022.

[4]. Neetesh Kumar, Sarthak Mittal, Vaibhav Garg, Neeraj Kumar,"Deep Reinforcement Learning-Based Traffic Light Scheduling Framework for SDN-Enabled Smart Transportation System", MARCH 2022.

[5]. Luiz Fernando Pinto de Oliveira ,Leandro Tiago Manera ,Paulo Denis Garcez da Luz ,“ Development of a Smart Traffic Light Control System With Real-Time Monitoring“ , MARCH 1, 2021.