GITAM (DEEMED TO BE UNIVERSITY)

Image & video processing ,analysis of traffic signal areas using deep learning ,AI Models

EECE DEPARTMENT
GITAM
BENGALURU

AY 2021-25

GITAM UNIVERSITY

Major Project
Project ID: XX

A University should be a place of light, of liberty, and of learning.

Department of Electrical Electronics and Communication Engineering

Project Mentor:
- Sanhita Manna

Project Team:

BU21EECE0100515 P Lakshmi Vignesh
BU21EECE0100104 Moda SriRangaManjula
BU21EECE0100487 Lalam Jithendhra

www.gitamedu.com

Project Group – Details

Photo	Track	Roll No	Name
	EECE	BU21EECE0100515	P Lakshmi Vignesh
	EECE AI/ML	BU21EECE0100104	Moda SriRangaManjula
	EECE	BU21EECE0100487	Lalam Jithendhra

Objective and Goals

Objective

Objectives

- This project involves developing a machine learning model that utilizes object detection techniques to analyze vehicle density across traffic lanes. An algorithm will be designed to display identified high priority vehicles at traffic junctions, enabling reduced emergency rate and accurate predictions of the time required to clear each lane and improving overall traffic management efficiency.

Goals

Main Goals

- To assign the time to each lane (i.e 5<=time<=120) by analyzing the vehicle density of that lane.
- To prioritizing the high priority vehicles identified while analyzing the vehicles and displaying them to clear path for those vehicles in that lane.

Additional goals

- To reduce the emergencies for people by prioritizing the high priority vehicles.
- To reduce the time for which the vehicles are waiting at traffic junctions.

ABSTRACT:

In any part of our life, each of us has encountered high-priority vehicles like ambulances, fire engines, and police vehicles waiting during their time of emergency. We have also experienced waiting for 90 seconds at traffic signal lanes even if the other lanes are empty.

We can save significant time and prioritize vehicles at traffic signals by analysing and executing better algorithms with the help of video processing and AI models.

The main motto of this project is to eliminate the excess time and prioritize the vehicles at the traffic signals. This can be achieved by training deep learning models.

Firstly, the models will analyse the density of the vehicles at each lane of traffic areas. Additionally, the models will identify high priority vehicles such as ambulances and fire engines. Based on the density of vehicles, the models will then assign time to each lane, ensuring the time assigned is within the range of 5 to 120 seconds.

Finally, traffic signal control systems need to be integrated with the AI models to dynamically adjust signal timings based on real-time traffic conditions and the presence of high- priority vehicles

INDRODUCTION: This problem statement revolves around the need to develop an advanced system that can automatically monitor, process, and analyze images and videos from traffic signal areas. By applying image and video processing, deep learning, and AI to analyze the vehicle densities to predict the time required for that lane and to identify the high priority vehicles and to prioritize them at traffic junctions. Which reduces the emergency rate which can save lives by reducing the waiting time and also results in time effective advancement at traffic junctions.

GITAM (DEEMED TO BE UNIVERSITY)

Project Plan

EECE DEPARTMENT GITAM BENGALURU

Software/Hardware

- Though this project mostly comprises of software technologies, it is an integration of hardware too. This project is to be implemented at traffic junctions.

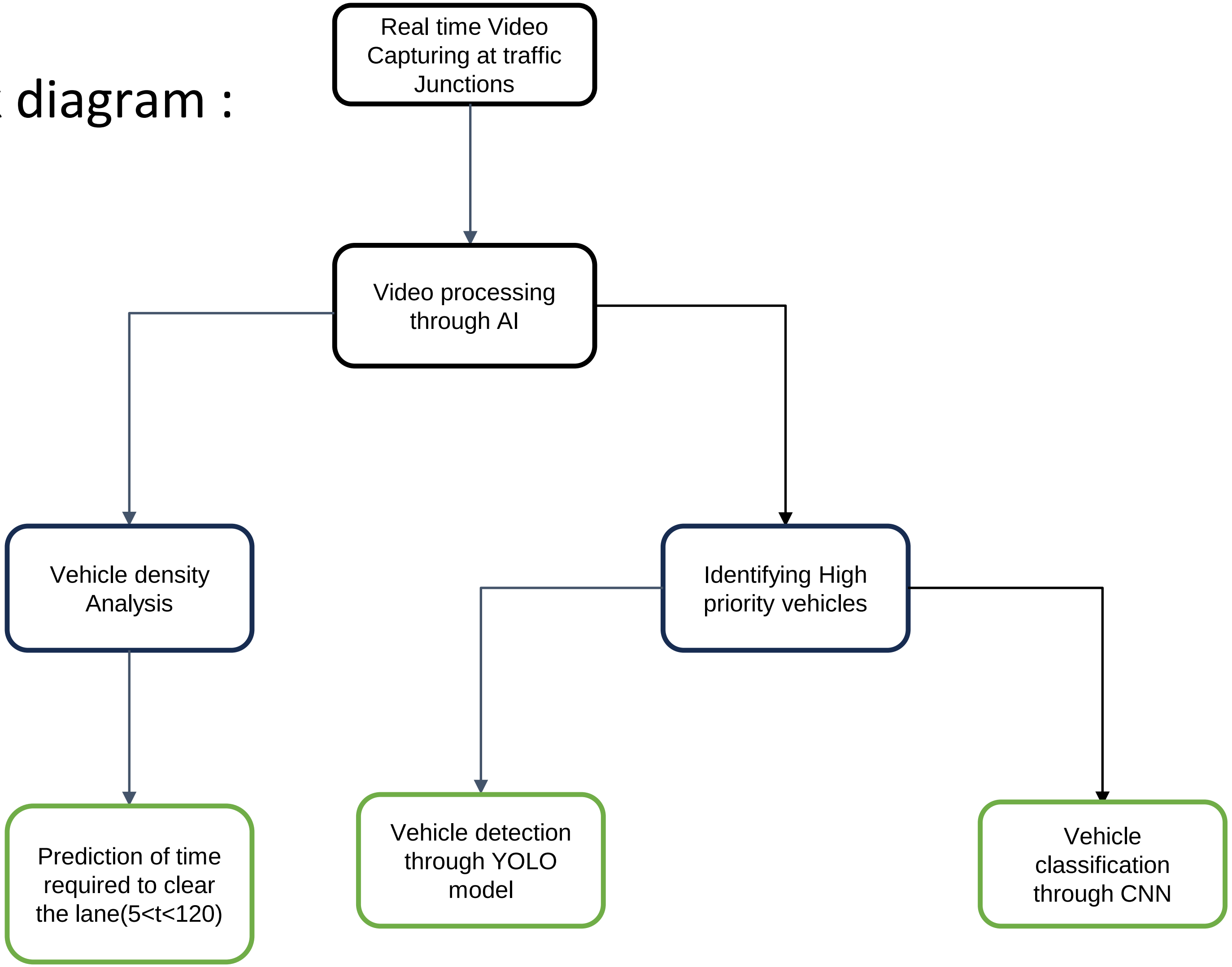
Software Requirements

- Object detection AI
- Object Classification AI
- Video processing
- Image preprocessing
- Vehicle density Analysis

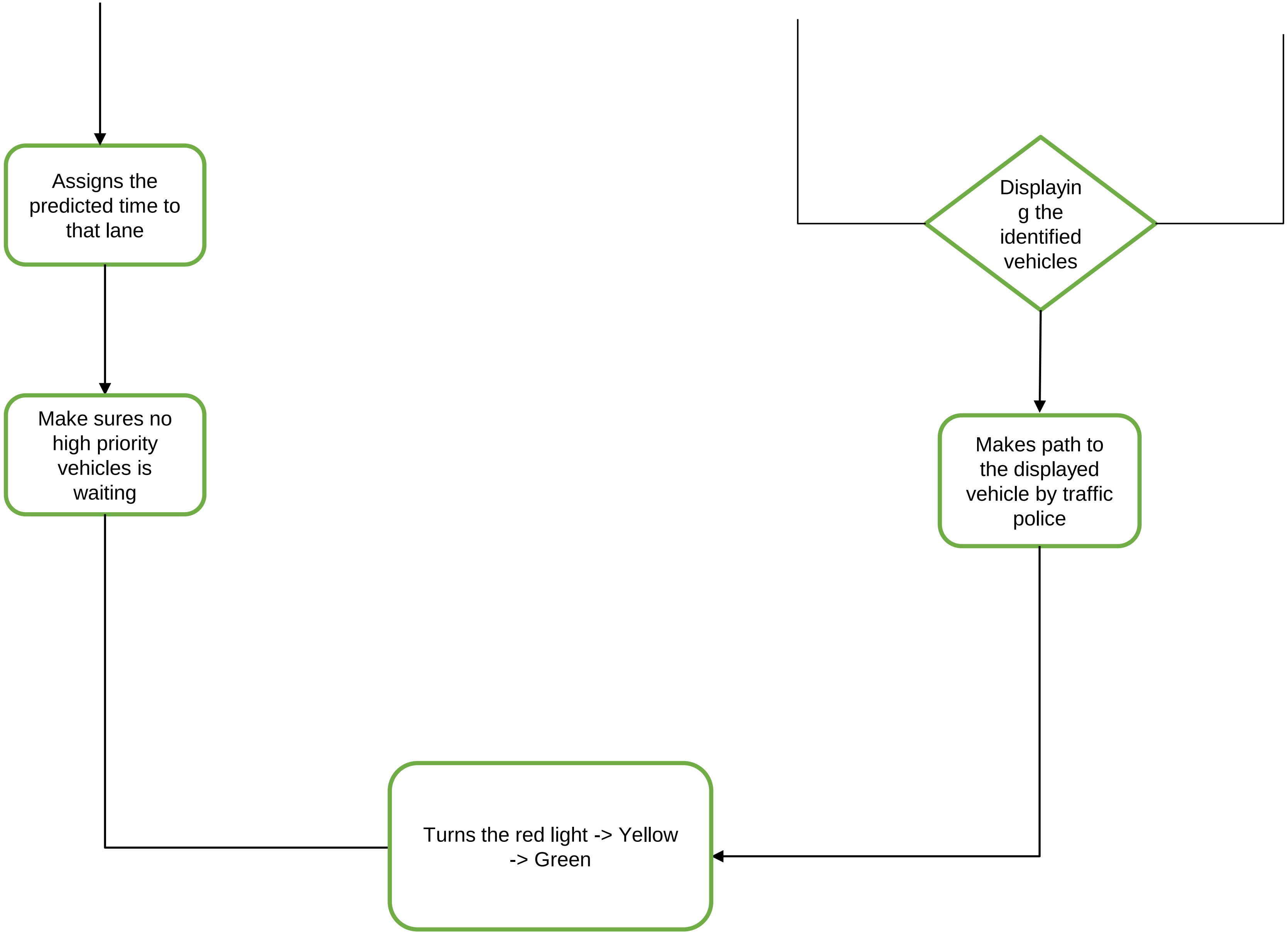
Hardware Requirements

- CCTV cameras
- LED displays
- Security cameras

Methodology - block diagram :

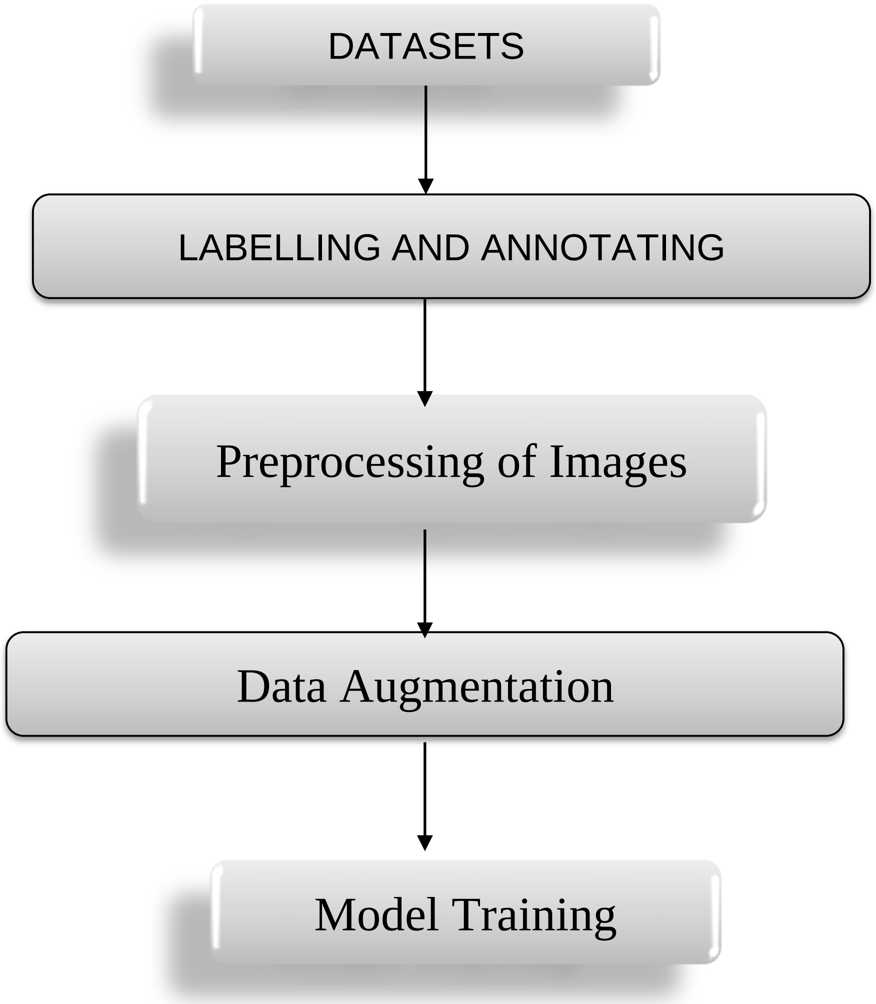


•


Assigns the predicted time to that lane
Make sures no high priority vehicles is waiting
Displayin g the identified vehicles
Makes path to the displayed vehicle by traffic police
Turns the red light -> Yellow -> Green

Architecture

Structural Diagram
Block Diagram/Pin Diagram



Use Cases & Testing

Use Cases:

Use Case 1 – Multiple Vehicle Detection with Prioritization:

•Multiple vehicles (e.g., cars, fire engines, ambulances) are detected at an intersection.

•**Action**: The system classifies the vehicles by priority, giving emergency vehicles (like fire engines or ambulances) the highest priority.

•**Expected Outcome**: The lane with the high-priority vehicle is cleared first, and signal timings are adjusted based on vehicle density.

Use Case 2 – Vehicle Distance and Time Assignment:

•**Scenario**: The system detects vehicles at varying distances from the traffic signal.

•**Action**: Based on the distance of vehicles and lane density, the system assigns traffic signal times dynamically (5-120 seconds).

•**Expected Outcome**: Vehicles closer to the signal get higher priority, optimizing traffic flow and minimizing waiting time.

TESTING

Test Case 1 – Greyscale and Augmentation Impact on Detection:

- **Objective**: Test how augmentation techniques (greyscale conversion, blurring, noise addition) impact the detection accuracy.
- **Process**: Compare the performance of models trained with greyscale and augmented images to those trained on original images.
- **Expected Outcome**: The augmented models (with greyscale, blur, noise) provide more robust detection in real-world conditions, especially under low-quality or noisy data.

Test Case 2 – YOLOv11n Object Detection Model:

- **Objective**: Test the accuracy of YOLOv11n for detecting vehicles such as ambulances, fire engines, and regular cars in real-time.
- **Expected Outcome**: The model correctly detects vehicles with at least 90% accuracy under different conditions.

Test Case 3 – Epoch Testing for Model Performance:

- **Objective**: Test model performance across different epoch settings (10, 20, 50).
- **Process**: Train the model for 10, 20, and 50 epochs. Evaluate its performance in terms of accuracy, detection speed, and error rates.
- **Expected Outcome**: The model performs best at 50 epochs, balancing accuracy and training efficiency.

Test Case 4 – Traffic Signal Timing Based on Vehicle Density:

- **Objective**: Test the system’s ability to assign appropriate traffic signal timings based on the vehicle density.
- **Process**: Simulate traffic with varying vehicle densities and observe if the time assigned is within the 5-120 seconds range.

Implementation and Results – Iteration 1

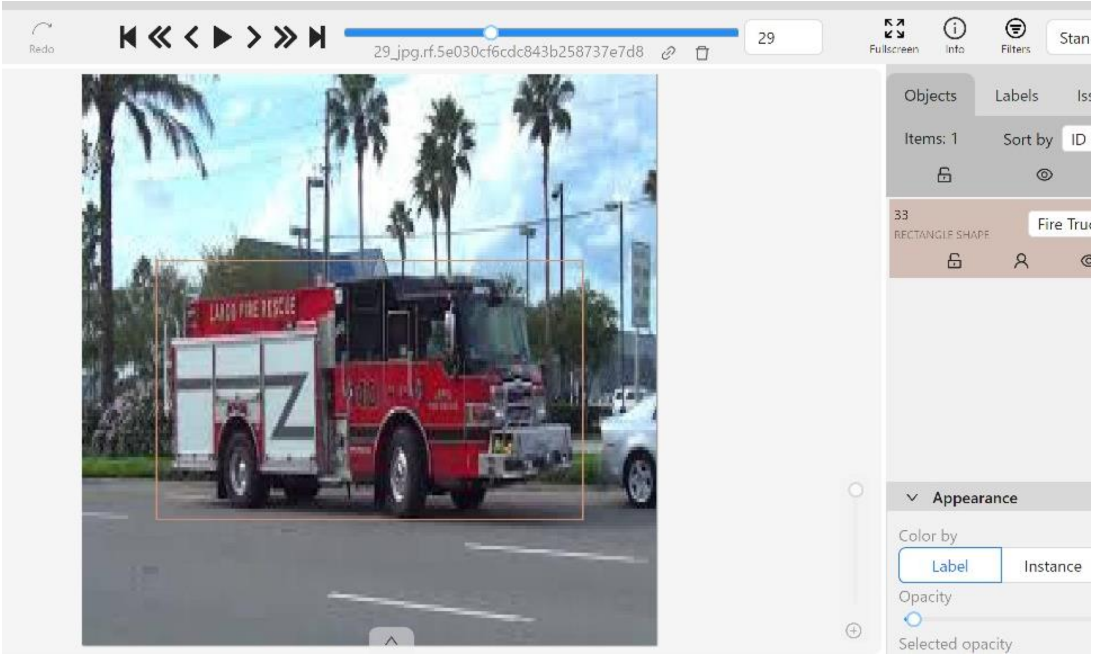
Implementation:

1.Dataset Collection:

- Collected datasets from Kaggle, Roboflow, and open-source image datasets containing vehicle images for training the model.

2.Image Labeling and Annotation:

- Used the **CVAT.ai** platform to label and annotate the vehicle images for the training process. This included drawing bounding boxes around vehicles like ambulances, fire engines, and other regular cars.



3 .**Preprocessing of Images:**

- Resized all images to a fixed resolution of 640 x 640 pixels to maintain uniformity.
- Applied auto-orientation to correct any misaligned images.
- Converted some images to greyscale and performed basic data cleaning as part of preprocessing.

4 . **Data Augmentation:**

- Performed data augmentation to improve the model's ability to detect vehicles in different conditions:
  - Converted 15% of the images to greyscale.
  - Applied blur to some images.
  - Added noise to other images to improve detection under various conditions.

607 Total Images

View All Images →

Dataset Split

TRAIN SET 87%	VALID SET 8%	TEST SET 4%
531 Images	51 Images	25 Images

Preprocessing

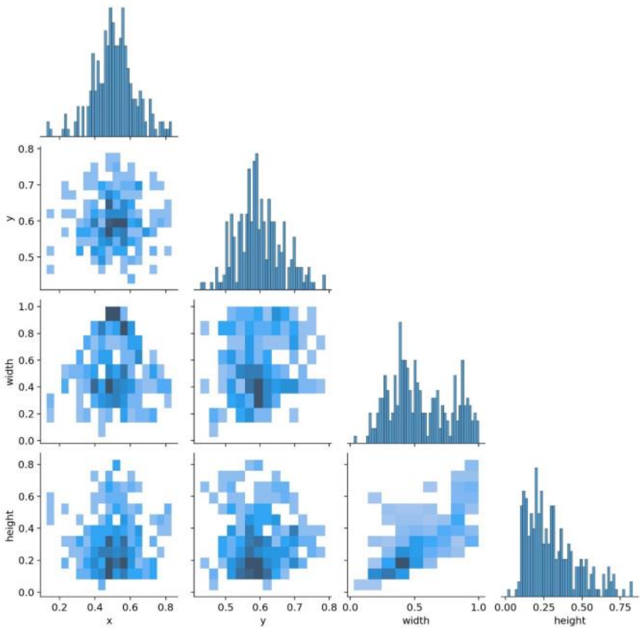
Auto-Orient: Applied
Resize: Stretch to 640x640

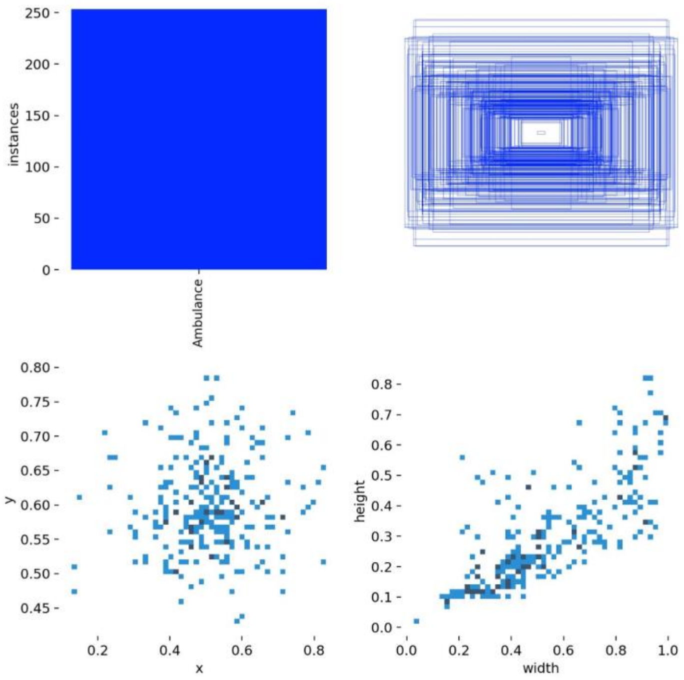
Augmentations

Outputs per training example: 3
Blur: Up to 3.4px
Cutout: 9 boxes with 24% size each

5 . Model Training:

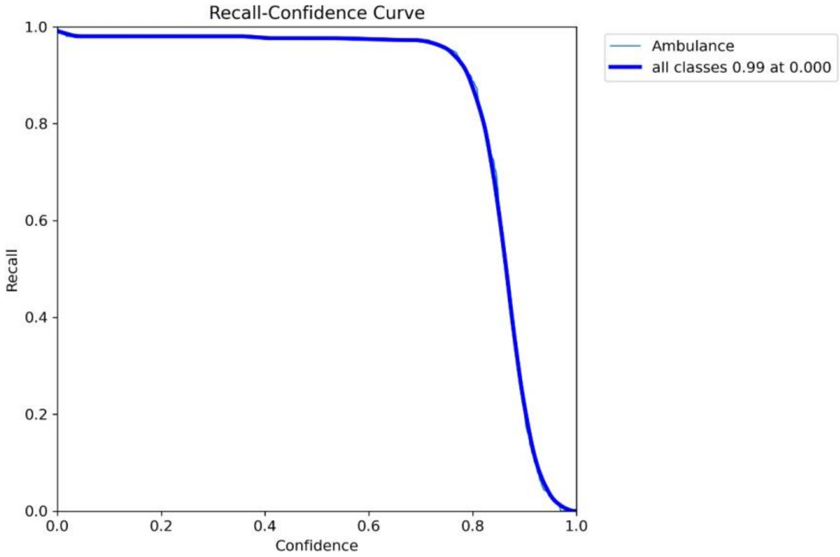
- Used YOLOv11 Object Detection Model to train the AI system for detecting vehicles.
- Initially tried YOLOv11m but faced compatibility issues, switched to YOLOv11n for better performance.
- Trained the model with 150 epochs, but later adjusted to 50 epochs, which provided the best results.
- Used various training iterations (10, 20, 50 epochs) to fine-tune model performance.

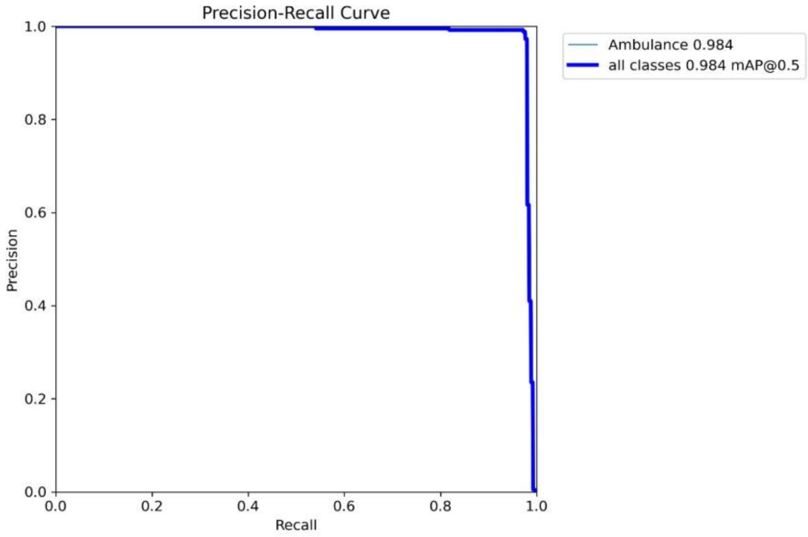


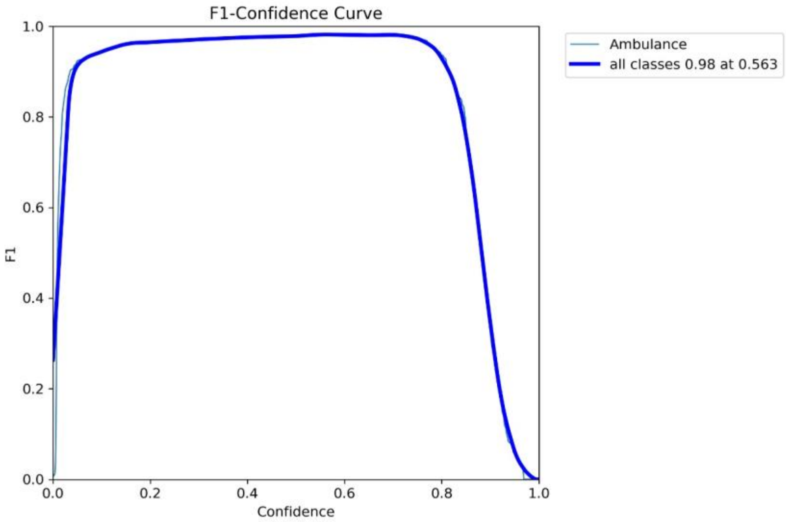


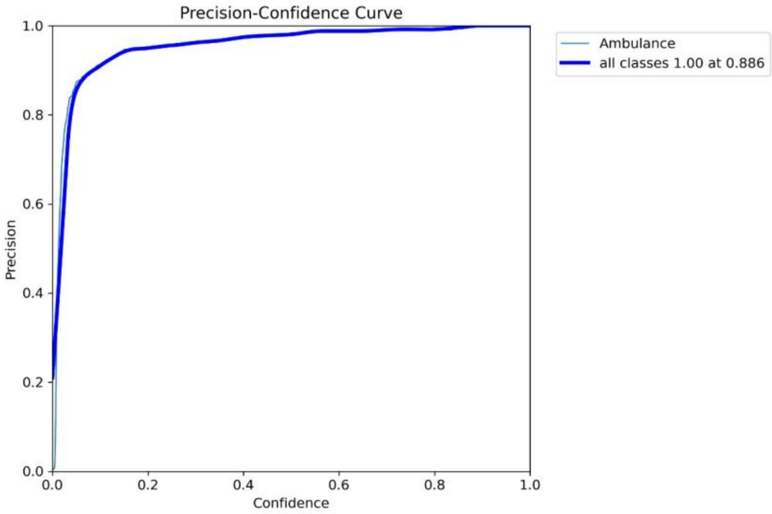
GITAM (DEEMED TO BE UNIVERSITY)

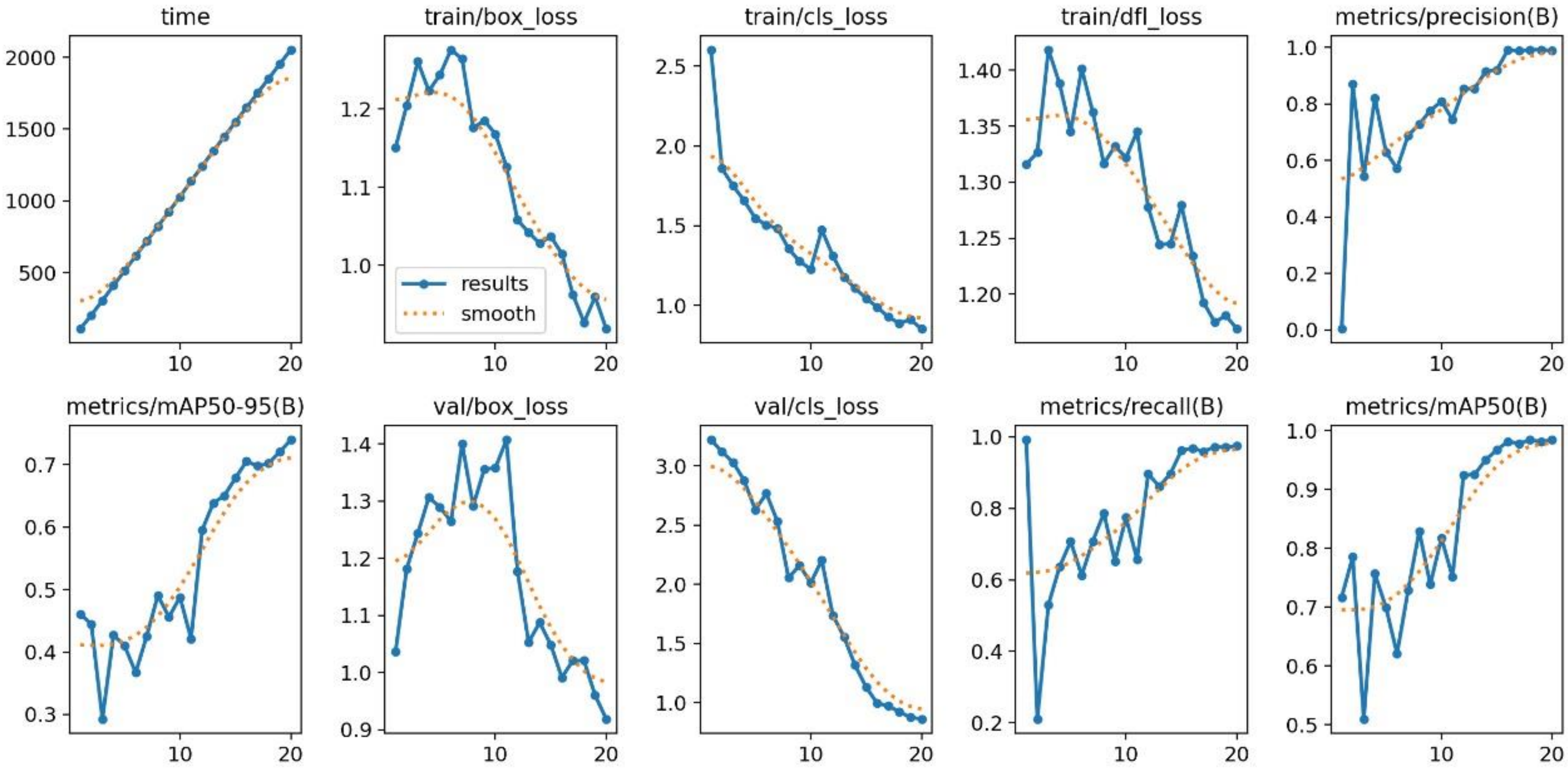
EECE DEPARTMENT GITAM BENGALURU










time
train/box_loss
train/cls_loss
train/dfl_loss
metrics/precision(B)
results
smooth
metrics/mAP50-95(B)
val/box_loss
val/cls_loss
metrics/recall(B)
metrics/mAP50(B)

Implementation and Results – Iteration 2

Iteration : Results + Validation against the use cases and test cases

Iteration 1 – YOLOv11n with 10 Epochs

- **Objective**: Test initial object detection accuracy and performance.
- **Result**: The model did not perform as expected. It failed to detect vehicles accurately, especially in real-time conditions.

Iteration 2 – YOLOv11n with 20 Epochs:

- **Objective**: Improve detection accuracy by increasing the number of epochs.
- **Result**: The model showed slight improvement but still had issues with accuracy, especially with detecting smaller vehicles and under noisy or blurry conditions.

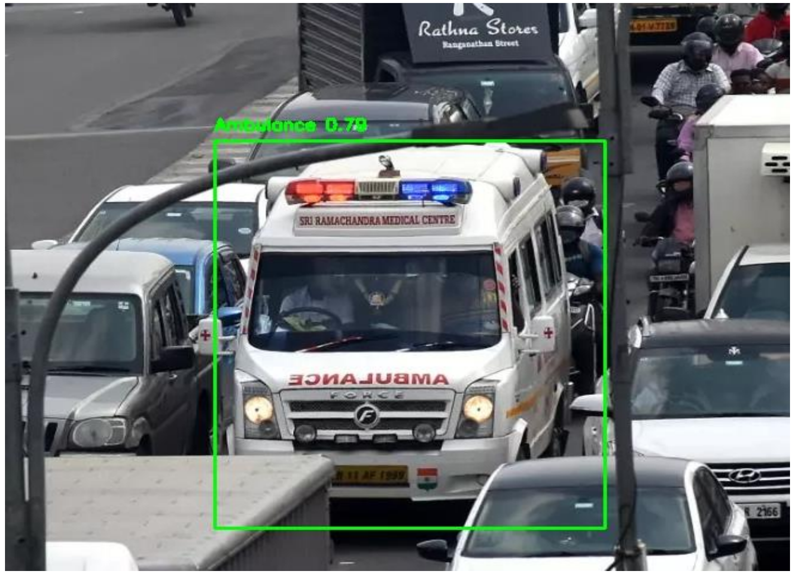
Iteration 3 – YOLOv11n with 50 Epochs:

- **Objective**: Optimize model performance with 50 epochs and further data augmentation.
- **Result**: The model achieved significant improvements, with over **90% accuracy** in detecting vehicles such as ambulances, fire engines, and regular cars. It performed well in real-time, even in varied lighting and environmental conditions.
- **Validation**:
    - **Use Case Validation**: The model successfully detected high-priority vehicles and adjusted traffic signals dynamically, as per the defined use cases.
    - **Test Case Validation**: The system passed all critical test cases, including vehicle detection under noise, blur, and various environmental conditions. The performance in terms of accuracy and real-time object detection was validated to be within the desired range.

Implementation and Results – Iteration 3 (Optional)

Results:

- The final YOLOv11n model, after 50 epochs, was able to accurately detect ambulances, fire engines, and regular vehicles with high precision.
- The model was able to achieve over 90% detection accuracy, making it highly reliable for real-time traffic management.



Contribution

Team Progress and Movement

- Dataset Collection
- Data labelling and Annotation
- Model Training
- Testing

Individual Contribution

Key contributions: Putlooru Lakshmi Vignesh

- AI Model Training
- Model Testing

Key contributions: Lalam Jithendra

- Dataset Collection
- Model Testing

Key contributions: Moda Sri Ranga Manjula

- Data labelling and Annotation
- Model Testing

Analysis - SWOT

Strengths

- Real-time Vehicle Density Analysis
- Prioritization of High-Priority Vehicles
- Improved Traffic Management
- Integration of Software and Hardware

Weaknesses

- Dependence on High-Quality Hardware
- Complexity of Implementation
- Data Privacy and Security Concerns
- Maintenance Requirements

Opportunities

- Scalability to Other Cities
- Potential for Integration with Smart City Initiatives
- Expansion to Additional Use Cases
- Collaboration with Government and Private Sectors

Threats

- High Initial Costs
- Regulatory and Compliance Challenges
- Resistance to Change

Analysis – 4W1H

Why: The main goal is to reduce congestion and waiting times at traffic junctions by dynamically managing the traffic light timings based on real-time vehicle density analysis. Ensuring that high-priority vehicles like ambulances and fire trucks have a clear path, thereby reducing the response time during emergencies.

What: : The project involves developing a machine learning-based system for detecting and classifying vehicles, analyzing vehicle density, and managing traffic signals accordingly. Utilizing video and image preprocessing techniques to analyze traffic conditions in real time.

Where: The system is designed to be implemented at traffic junctions, particularly in busy urban areas, to optimize traffic flow and enhance safety.

When: The system is designed to be implemented at traffic junctions, particularly in busy urban areas, to optimize traffic flow and enhance safety.

How: The system will use CCTV cameras and other security cameras installed at traffic junctions to capture real-time video footage. The system will use CCTV cameras and other security cameras installed at traffic junctions to capture real-time video footage. Deploying machine learning models to analyze the captured footage and make decisions regarding traffic light timings and high-priority vehicle management.

Refined Objective: To Develop an AI-Powered Traffic Management System that Enhances Traffic Flow and Safety by Dynamically Adjusting Traffic Signals Based on Real-Time Vehicle Density and Prioritizing High-Priority Vehicles at Traffic Junctions. This system aims to reduce minimize emergency response times, and improve overall traffic efficiency
.

Conclusion & Future Work

Summary and Conclusion :

Summary:

In this project, we developed a smart traffic management system capable of real-time vehicle detection using advanced YOLOv11 models.Data augmentation techniques such as greyscale conversion, blurring, and noise addition were applied to improve the model's robustness.

Through several iterations, we improved the model's accuracy, achieving over 90% detection accuracy after training for 50 epochs.The system successfully detects ambulances, fire engines, and other vehicles, adjusting traffic signals dynamically in real-time based on vehicle priority and proximity.

Conclusion:

The implemented YOLOv11n model, after various iterations and augmentations, provides reliable and accurate vehicle detection, ensuring smooth traffic flow and prioritization of emergency vehicles. This system is adaptable to real-time conditions, making it a strong candidate for deployment in smart traffic management systems

Future Work:

In future work, we aim to increase the accuracy of detecting fire engines and further enhance the system's performance. Additionally, we will work on finding the distances of high-priority vehicles from traffic signals. Based on this, the system will dynamically assign traffic signal time to prioritize emergency vehicles efficiently. This will lead to better traffic optimization and quicker response times for emergency services.

References

REFERENCES

- [1] Nellore K., Melini S.B. Automatic Traffic Monitoring System Using Lane Centre Edges. IOSR J. Eng. 2012; 2:1–8. [iosrjen]
- [2] Sangamesh S B, Sanjay D H, Meghana S, M N Thippeswamy "Advanced Traffic Signal Control System for Emergency Vehicles" International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878, Volume-8 Issue-3, September 2019 [IRJET]
- [3] R. Bharadwaj; J. Deepak; M. Baranitharan; V. V. Vaidehi. "Efficient dynamic traffic control system using wireless sensor networks" Doi: 10.1109/ICRTIT.2013.6844280 [IEEE]
- [4] Krishnendu Choudhury; Dalia Nandi "Detection and Prioritization of Emergency Vehicles in Intelligent Traffic Management System" 2021 IEEE Bombay Section Signature Conference (IBSSC) DOI: 10.1109/IBSSC53889.2021.9673211 [IEEE]
- [5] Lingani, Guy M., Danda B. Rawat, and Moses Garuba. "Smart traffic management system using deep learning for smart city applications." 2019 IEEE 9th annual computing and communication workshop and conference (CCWC). [IEEE] , 2019.

THANK YOU