Traffic signal optimisation and control using deep learning framework deployed over cloud for connected Traffic Management

MSc Research Project
MSc Cloud Computing

Jatin Nikhil Dilliraj Saraswathy
Student ID: 23127228

School of Computing
National College of Ireland

Supervisor: Mr. Vikas Sahni

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Jatin Nikhil Dilliraj Saraswathy
Student ID:	23127228
Programme:	MSc Cloud Computing
Year:	2024-2025
Module:	MSc Research Project
Supervisor:	Mr. Vikas Sahni
Submission Due Date:	03/01/2025
Project Title:	Traffic signal optimisation and control using deep learning framework deployed over cloud for a connected Traffic Management
Word Count:	5383
Page Count:	20

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	Jatin Nikhil
Date:	29th January 2025

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	☐
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	☐
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	☐

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Traffic signal optimisation and control using deep learning framework deployed over cloud for connected Traffic Management

Jatin Nikhil Dilliraj Saraswathy
23127228

Abstract

ITOS (Intelligent Traffic Optimization System) is a key component of effective communication and transport systems. This research presents an effective ITOS. that incorporates high technology such as deep learning and optimization methods odds to improve the traffic density, congestion, and negativity on the environment amongst the growing urban traffic. Innovation is placed on the need to give a robust weather transitioning and computationally optimized framework that implements different architectures of urban and environmental settings. ITOS enhances traffic. flow by applying real-time data analysis, predictive computer simulation tools, and real-time adaptive signal control. The mAP score of YOLOv11 has improved more. 200.7 to 1.0 for different kinds of vehicles that can be identified on the roads. The The YOLOv11 model performs better and has been used for the vehicle count along with STL-ARIMA for future vehicle prediction. This project is to create smart, when and where management technology in order to create the basis for smarter, safer, and environmentally sustainable cities.

Keywords:YOLO models, ITMS, time series, congestion, traffic signal Green Light Optimization

1 Introduction

Smart cities cannot be understood without addressing the issue of mobility since a uni-Fied and interconnect traffic management is the key for sustainable and efficient urban transport infrastructure. An **ITOS** uses artificial neural networks and signal control. strategy to enhance the flow of traffic and decrease traffic jams and risky areas. The sys-team uses state-of-the-art computer vision algorithms comprising the non-YOLO and the latest YOLOv11 model besides the time series forecasting for the dynamic realignment of signal timings. For this reason, ITOS integrates real-time data analytics, predictive analyzing adaptive signal timing technologies to provide a large-scale solution to the city traffic conditions. It is an explicit solar aspect, as these systems do not only enhance com-mature experience but also make complex green contributions towards reduction of car-bon and energy consumption; thus, making cities improve to cater to modern standards and comforts.

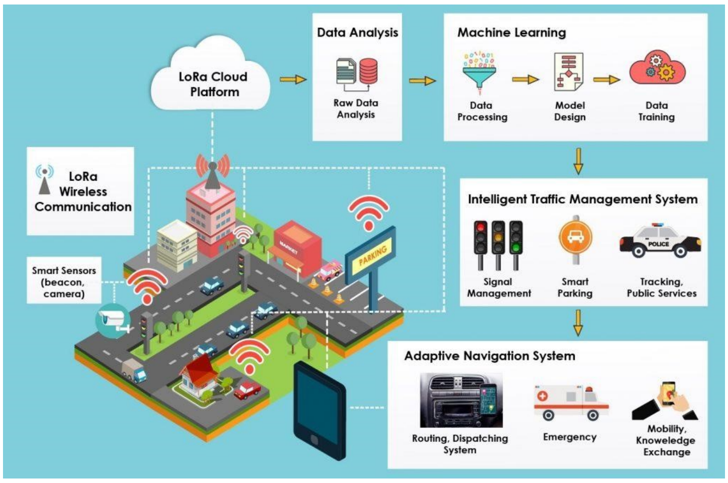

Figure 1: LoRA (long range) based smart traffic management (Source: *What is a Smart Traffic Management System? — symmetryelectronics.com* (n.d.))

1.1 Aim of the project

The main objective of this project is to design and implement a real-time intelligent traffic Management System (ITOS) that optimizes traffic signal timing at urban junctions. Utilize sophisticated machine learning methodologies, cloud architecture, and optimization algorithms to achieve the following objectives:

- **Enhance Traffic Efficiency:** Utilize computer vision to monitor vehicle counts. and predict traffic trends for the dynamic adjustment of green light durations.
- **Mitigate Congestion:** Reduce delays and bottlenecks by imposing penalties or prioritizing traffic in designated directions according to real-time vehicle density.
- **Alleviate Environmental Impact:** Decrease fuel usage and emissions through enhanced transportation efficiency.
- **Guarantee Scalability:** Formulate lightweight, storage-efficient, and computationally optimized algorithms for swift real-time decision-making. -time decision-making.

The system does emulate the decision-making powers of traffic police, offering an intelligent gent, data-driven reaction mechanism that enhances urban mobility and facilitates more effective transportation management.

1.2 Motivation

The urgent demand for sustainable and effective urban transportation solutions renders the advancement of intelligent traffic management systems (ITS) is essential for contemporary cities. By using video detection systems, edge processing, and interconnected traffic signal systems, Intelligent Transportation Systems (ITS) enhance traffic flow while also tackling significant urban issues. This project is motivated by the advantages of intelligent traffic management systems:

- **Predictive Insights for Planning:** Real-time data from traffic sensors and inter connected systems provide critical insights into roadway utilization, allowing urban planners to develop infrastructure that addresses future requirements.
- **Improved Safety:** Through the incorporation of sensors and AI-driven systems, Intelligent Transportation Systems (ITS) may markedly decrease traffic collisions. and pedestrian deaths, hence assuring safer roadways.
- **Cost Efficiency:** By reducing crashes and optimizing mobility, ITS mitigates economic losses and improves resource usage.
- **Accelerated Emergency Responses:** Enhanced traffic flow enables emergency vehicles to maneoeuvre effectively, saving lives in urgent circumstances.
- **Environmental Sustainability:** • ITS is essential in minimizing emissions via route optimization, establishing it as a fundamental element of sustainable urban planning.

This initiative sought to leverage new technologies, including AI, ML, and IoT, to tackle these difficulties. Aimed to develop an adaptable and scalable traffic optimization system to provide a safer, greener, and more efficient urban transportation network. The revolutionary capacity of ITS extends beyond simple congestion alleviation; it established the groundwork for more intelligent and habitable urban environments.

1.3 Research Objective

The objective of this research is to develop a comprehensive and adaptive traffic optimization framework by deeply analyzing and integrating the latest advancements in computer vision and time series forecasting models. The study does:

- **Examine Computer Vision Algorithms:** Perform a detailed comparative analysis. analysis of traditional non-YOLO models (such as R-CNN, SSDNet, and Mask R-CNN) and the latest YOLO models, including the recently introduced YOLOv11, to eval- rate their performance in detecting and analyzing vehicle density under diverse conditions, including adverse weather scenarios.
- **Integrate Time Series Forecasting Models:** Explore state-of-the-art time series forecasting algorithms to predict traffic flow patterns, giving precedence to these models during challenging conditions (e.g., rain, fog, or low visibility) where computer vision models may struggle.
- **Develop a Unified Solution:** Create an integrated system that dynamically prioritizes time series forecasting over computer vision in tough environmental conditions, ensuring seamless traffic management with minimal disruption.
- **Optimize for Real-World Use:** Evaluate the system's computational efficiency, response time, and storage requirements to ensure real-time adaptability and feasibility ability for deployment in metropolitan areas.

By combining insights from cutting-edge computer vision and predictive analytics, this re search aims to deliver a robust, weather-resilient, and scalable traffic optimization solution. that enhances urban mobility while maintaining safety and efficiency in all conditions.

1.4 Research Questions

Based on the above discussions, the following are the research questions,

1. What are the most effective methods for utilizing real-time traffic data from CCTV? cameras and sensors to optimize traffic signal timings at urban intersections?

2 Related Work

2.1 Current Traffic Detection Methods

In mitigating the challenges of identifying foreground objects in urban traffic scenes. Ghahremannezhad et al. (2023) developed a background subtraction. This method identifies an adaptive local median texture feature making use of weber 's law for changing illuminations. Hence from the portable visual background extractor, the researchers suggested a model based on consensus sampling. This model has a random updating mechanism that addresses dynamic backdrops within the model. It can manage change in light level and has better performance.

The rapid growth of the urban structure is a major problem for traffic management that traditional solutions would have great difficulty in solving. Kumar et al. (2024) described an approach for obtaining accurate traffic information such as flow rate, travel pattern and speed of the vehicles with the help of video cameras set up in the cities. Their approach makes use of YOLOv3 and a SORT monitor for traffic classification and traffic prediction, respectively.

According to Xu et al. (2022) ,a model that adopts both a BiLSTM_Attention network and the Whale Optimization Algorithm (WOA) may help raise the signal prediction accuracy. The traffic flow is predicted by the BiLSTM_Attention network, and WOA is used to find the optimal learning rate, number of epochs, and number of nodes in two hidden layers, respectively. This WOA_BiLSTM_Attention model also outperforms the classical neural network models and novel WOA-optimized neural networks in terms of MAPE, RMSE, MAE, and R^2 suggest increased accuracy.

The problem of identifying traffic is quite challenging; however, deep learning tech techniques are emerging as a possible solution to the network traffic classification. However, such techniques mostly require a large amount of training data and extracting expert features, which is very time-consuming and often requires a lot of effort. Regarding these challenges, Izadi et al. (2022) propose a novel technique, where CNN and ALO are used together in conjunction with SOM for traffic classification.

The authors have designed this traffic forecasting system for advanced transportation systems by H R et al. (2022). It employs a procedure of using data of the preceding year. and the current data to predict accurately the traffic conditions. It helps people to track the current traffic situation, which is also extremely helpful for drivers who need traffic updates on the spot. It aids in finding out the areas of the city that experience high traffic, and the best decisions can be made on the manner to approach the same roads. while driving.

2.2 Advances in traffic Optimization Strategies

Ntakolia and Lyridis (2022)present a novel n-DACOF algorithm—our proposed n-D ant. colony optimization technique with fuzzy logic for air traffic flow management. The aim of the algorithm is the reduction of total costs associated with being airborne. and held on the ground, fluctuations in speed, route changes, and cancellation policies. The algorithm's objective is to minimize the overall expenses incurred due to delays in air and ground holding, variations in speed, changes in routes, and cancellation policies.

Ezekwere et al. (2020) use the Spider Monkey Optimization (SMO) algorithm-based on the Fission–Fusion Social System (FFSS) behavior in spider monkeys to avoid traffic congestion. To illustrate its effectiveness in reducing total distance travelled, an intersection is modeled as a four-leg intersection. The findings herald that SMO trumps ABC. because it is decentralized, stochastic, and self-organizational in nature. This paper shows that SMO is more effective than fully actuated control methods, especially for conditions with heavy traffic flow.

Forecasting and mitigating road traffic congestion is the topic that Bartlett (2023) examines using the application of machine learning. The work offers a detailed model on how to estimate traffic flows that are common on the internet, with a special emphasis on the miscellaneous traffic types common in urban areas. The main contributions include a detailed analysis and comparison of existing machine learning models. The outcomes show that artificial neural networks are best suited for short-term prediction and further, that accuracy could be boosted by segmented classification towards different vehicles types.

Describing congestion and classifying it has been done by Zarindast et al. (2022) using past probe data from Des Moines, Iowa. The researchers were able to partition speed. signals and find temporal congestion instances following the Bayesian change point detection methodology. The study classified the observed congestion breakdown into recurrent congestion (RC) and the non-recurrent congestion (NRC). It outlined a statistical expert. system and technique that worked with the big data to properly capture the two varieties of congestion.

They also presented a MLDNN and a CI based on traffic density factor for direct traffic congestion prediction by Kumar et al. (2024). The authors gathered data in the Delhi city using a video camera placed at a chosen site during business hours of the week. The collected data were divided into five-minute intervals and put into a matrix.

2.3 Recent technologies for traffic and data management

Using Virtual Ad-hoc Networks (VANETs) and the Internet of Vehicles (IoV), Elsagheer Mohamed (2019)developed an ITMS that includes a traffic signal controller. To improve traffic flow, reduce congestion, and provide precedence to emergency vehicles, this technology allows for wireless communication between infrastructure and automobiles. The average waiting time and the number of cars serviced were both improved. with the help of an adaptive algorithm.

Using Internet of Things (IoT) and Wireless Sensor Networks (WSNs), Mohammed et al. (2024)created a Data Traffic Management system for smart agriculture. They used differential encoding and Huffman techniques to build a lossless compression scheme at the sensor node level, which was lightweight. When put to the test with temperature readings, their method outperformed more conventional WSN approaches in terms of lowering data transmission and increasing energy savings.

2.4 Research Gap

Despite some advance in traffic optimization and management systems some research gaps exist; As shown by Ghahremannezhad et al. (2023) and Kumar et al. (2024) many studies demonstrate positive results of computer vision and deep learning approaches. However, their ability to perform in real dynamic environments, for instance more car traffic on the roads or in extreme weather conditions, has received little consideration. This is the reason while working with such models as YOLO computer vision models often face challenges to maintain a high level of recognition when visibility is low, for example, during rain or fog. However, models like BiLSTM_Attention that has integrated Whale Optimization Algorithm Xu et al. (2022)for traffic forecasting are accurate, their integration in real-time systems requires further research regarding fluid decision-making application. Moreover, the effectiveness and adaptability of these algorithms implemented on a large scale in cloud systems, as discussed in the study using SMO and IoT frameworks, should be improved for dynamics in real urban scenarios. Last but not the least, less emphasis has been placed on the integration of two or more technologies in an efficient and flexible traffic management system as in the case of VANETs, IoT and deep learning.

2.5 Research Contribution

This work helped in the design of an enhanced real-time intelligent traffic management system by further investigating an array of computer vision algorithms ranging from conventional Non-YOLO models, YOLOv7, YOLOv8, down to the current YOLOv11. This is synchronized within an experimental assessment of time-series forecasting models resulting in an overall differentiating and optimal solution for urban traffic management in dynamic and unfavourable conditions such as rain or low visibility. While real-time detection and counting of vehicles is achieved through computer vision methods, time series forecasting is more important to important decisions in the proposed system. The proposed concept is to use the advanced Machine Learning techniques, Cloud Computing resources and optimization for efficient traffic signalling at the metropolitan junctions. Is currently able to predict the flow of vehicles and vary the green light durations by sanctioning the number of vehicles in the either direction an essentially replicating the decision-making ability of a traffic policeman. Further, it stressed the computational complexity with respect of time, space and storage to get a solution that is optimized in terms of performance in addition to the storage required by the algorithms. They enhance environmental sustainability through addressing the scourge of traffic congestions and associated delays through channelling pose traffics them through timely responses and thus cutting any form of dead time.

3 Methodology

3.1 Dataset Description

This project aimed to enhance traffic optimization through object identification using the well-known KITTI dataset, which is a popular benchmark for research in autonomous driving. The KITTI collection contains high-resolution images of car cameras shooting in various environments, like a city, countryside, or highway. The KITTI_YOLO_LABELS dataset contains eight object classes in YOLO format: Car, Pedestrian, Van, Cyclist, Truck, Misc., Tram, and Person Sitting. A total of 749 images was divided into different training and testing sets, with 6,732 images in the training. The work was partitioned such that we could train and test YOLO models on different versions such as YOLOv3, YOLOv4, YOLOv8, YOLOv9, and YOLOv10, alongside other object detection models types like SSDNet, RetinaNet, and Mask R-CNN in object detection for traffic. All the images bore YOLO-specific class IDs as well as bounding box coordinates for each picture. This format thus guarantees the precise training and evaluation of our model.

Figure 2: A sample of KITTIE dataset with point cloud

Figure 3: A sample of KITTIE dataset with point cloud

[1]

3.2 YOLOv11

YOLOv11 has made computer vision lighter and more effective by optimizing the back bone and neck of the architecture while gaining a 0.5 percent increase in mAP and 22% fewer parameters than YOLOv8. It covers many tasks starting from detection and classification, it demonstrates versatility in working across edge devices and GPUs and can help in various industries for training, testing, and deploying models easily.

3.3 LSTM

LSTM networks are a great tool to model sequential data since they are able to deal with long-term dependencies because the internal memory is controlled.

[1]https://www.kaggle.com/datasets/katarzynakowieska/kitti-vehicle-detection-dataset

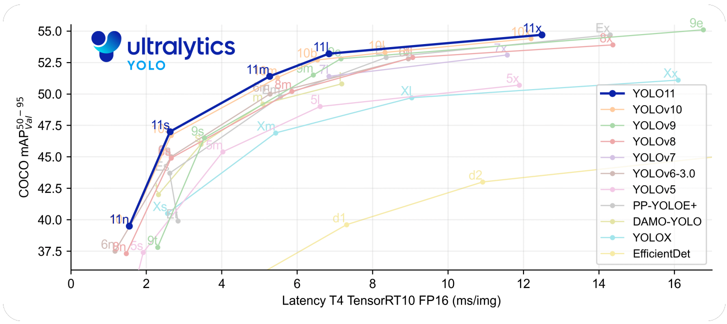

Figure 4: Performance Evaluation of Different YOLO model (Source:Ultralytics (n.d.)

3.4 ARIMA

ARIMA stands for Autoregressive Integrated Moving Average, and it's a technique for time series analysis and for forecasting possible future values of a time series.

3.5 MAPE

MAPE, also referred to as MAPE, stands for mean absolute percentage error and is another statistical measure utilized to assess the accuracy of a specific model. This accuracy is normally rounded off to a percentage, which is arrived at using the given formula.

MAPE=100 1/n _(k=1)n̂—(A_t-F_t)/A_t —

The Root Mean Square Error (RMSE) is the square root of the average squared difference between the predicted values and the actual values. Residuals quantify the deviation of data points from the regression line, whereas RMSE quantifies the dispersion of these residuals. Put simply, it provides information on the level of data clustering. around the line of best fit.

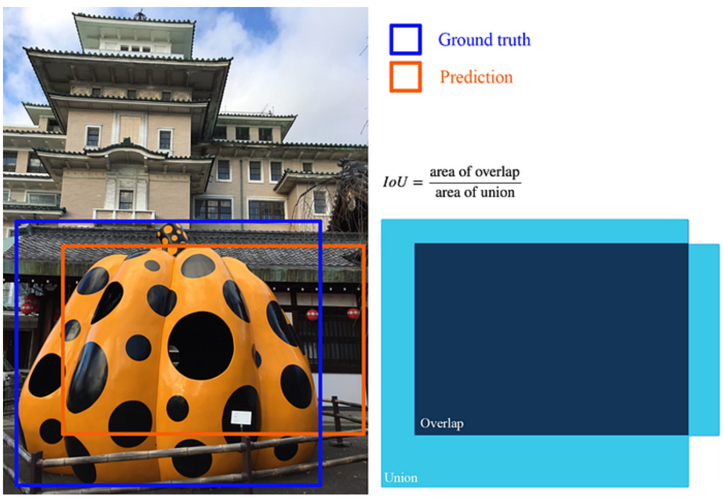

Figure 5: IOU and mAP (Source: Hui (n.d.))

IoU reflects to what extent there is an overlap of two boundaries. We use that to calculate how much of our predicted boundary aligns with the true boundary or the ground truth as some prefer calling it. In some of the dataset, we ourselves set an IoU threshold (for instance 0.5) when deciding whether the prediction is a true positive or false positive.

4 Design Specification

4.1 System Novelty

Following are the unique things that has been done in this project,

- In terms of the computer vision-based solution, we identified YOLOv11, which is the latest addition to the YOLO models is used to compare a deep dive comparison of all the different types of models, which is not done yet.
- We incorporated time series algorithms to help in the future time frame prediction. so that during difficult situations the scouting is agnostic. For example, during immense fog, the time series can help in getting the number of vehicles detected.
- The response system can be integrated with any of the cloud frameworks to give the real-time green signal based on that particular city.

4.2 Algorithm Flow

Step 1: The first step is object detection in real time; the second step is the preparation. of the set for further management of green light timing. This includes analysis of density. road and number of vehicles present during the operation of the green signal. The CTMS operates on a three-agent framework:

- Object Detection
- Vehicle Density Estimation
- Traffic prediction based on Weighted assignments

Several traffic cameras help in capturing the frame of a particular intersection in order to monitor the movement of vehicles.

Real-Time Object Detection: A Custom Retrained detection algorithm processes the captured video frames in the cloud to identify and classify different types of objects, such as:

- Auto-richshaws
- Motorbikes and bicycles
- Buses and Trucks
- Pedestrians

Step 2: *Dataset Formulation:* In object tracking SORT (Simple Online and Realtime Tracking) algorithm is applied. Steps include:

- Segment Implement Multiple Object Tracking (MOT) with SORT.
- Use an imaginary line, for instance the zebra crossing line to count the number of cars that cross the line.

When using aggregate vehicle counts, the data should be made seasonal by counting the number of vehicles in 10-minute slots in order to avoid overfitting of predictions. Organize a dataset based on these counts for better understanding of traffic flow and future traffic patterns.

Unique ID Management: Provide and erase distinctive identification references for vehicles to facilitate tracking and to reduce sorter detection.

Step 3: *Traffic Forecasting and Congestion Detection:* Monitor all the traffic directions in real time at a go. Identify congestion and calculate density for the roads to help enhance traffic flow. As a result, apply time series algorithms in terms of vehicle patterns. and timing signals optimization.

Result Aggregation: It must also be noted that weighting of results from traffic fore casting and density detection be done. Supplement a prescriptive approach to control signal timings in order to operate optimally and continually.

Step 4: *Algorithm Tuning:* The efficiency of the algorithms should be repeatedly assessed. Adjust them if performance levels are not attained to provide a greater level of precision and dependability.

Optimization: The ideas from the above steps should be incorporated in order to minimize congestion and enhance the overall flow in all the directions adequately.

Step 5: *: Time Series Forecasting:* Once the computer vision detection is done, the same is stored as the timestamped vehicle counts number, in real time both the object detection detects the number of vehicles, and the time series detects the amount. of vehicles that can be present in the next moment.

This has an effectiveness of dealing with the traffic by using real-time data advanced detection systems and ability to adapt to change in smart cities.

5 Implementation

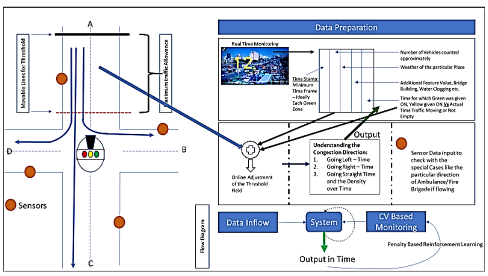

Figure 6: Implementation Flow

The architecture design of our development with Intelligent Traffic Optimization Systems (ITOS) is shown in the figure below. Fundamentally, this system aims at acquiring real time data for the purpose of correct object counting with the ROI as the most important aspect. The ROI represents the specific area in the traffic footage where vehicle detection and counting occur, as shown in Figure 9: Region of Interest (ROI).

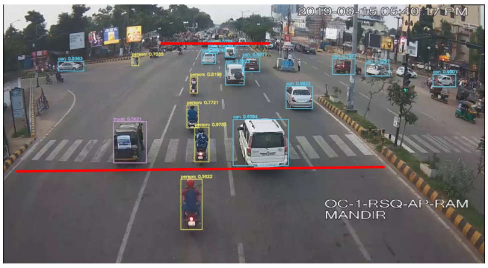

Figure 7: Region of Interest

5.1 Cloud Framework

To assess the numbers of vehicles that can be assigned to the final green light module shown in the above pseudocode, we have launched an EC2 Ubuntu server on AWS to process the vehicle video inputs. The initial steps of the POC included establishing an SSH terminal connection to the EC2 instance and building the virtual environment necessary for the project in terms of Python dependencies.

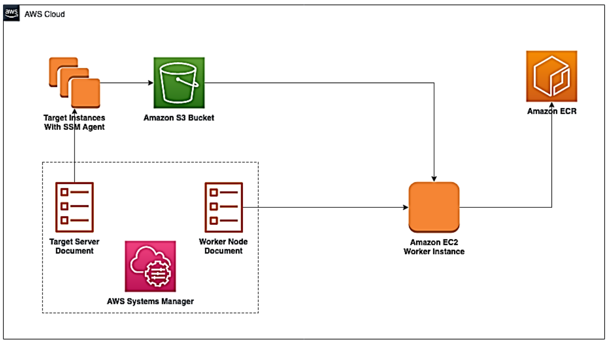

Figure 8: Cloud Connection

When the virtual environment for the project was set up, we proceeded to install the needed Python packages for processing the video, detecting vehicles, and analyzing the traffic flow. Also distinguished the interoperability with some server-side operations in Node.js, in addition to Python, with the use of PM2 as vital tools to keep a Streamlit application running constantly in the background. PM2 was then used to start the Streamlit server to keep the vehicle counting process going. Currently, the live vehicle Videos are processed by the EC2 instance to determine the count of vehicles in the lot. This information is applied to manage traffic light involving the decision of the number of cars that would be given the green light to proceed. This setup is well implemented.

in AWS, where a large volume of vehicle data will be processed, thus making the traffic management system more reliable and scalable.

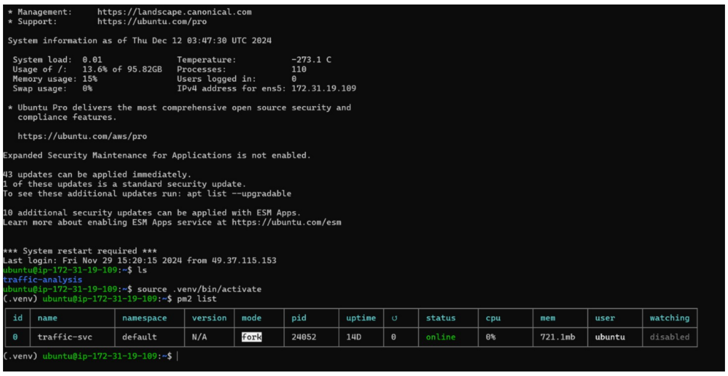

Figure 9: Deployment Status

6 Evaluation

6.1 Object Detection

6.1.1 Using Old Models

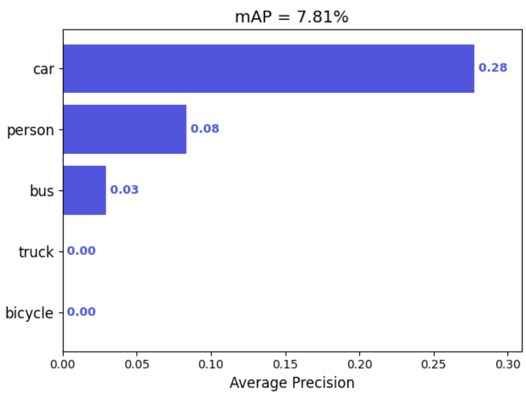

Figure 10: (a) SSDNet

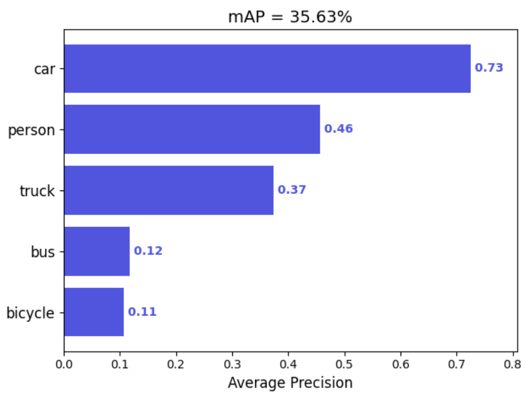

Figure 11: (b) Mask RCNN

Our observation reveals that, while using an image, the efficiencies of SSD net to identify bicycles are very poor, especially if the image shows the front or back view of the bicycles. Nevertheless, its performance is relatively better if the displayed image is the side view of the bicycles. Because objects are the primary items detected on roadways, some of the images of the provided dataset are marked by the presence of a large number of “car.” Therefore, the average precision (AP) of the ”car” class is the highest. In addition The rate of true positives is at its highest in cases defining the top category known as ‘car.’. Mask RCNN’s performance was evaluated using the KITTI dataset, and different results for different categories of objects were observed. The detection accuracy to bicycles was low; the result is similar to SSD Net. The model successfully delivered high

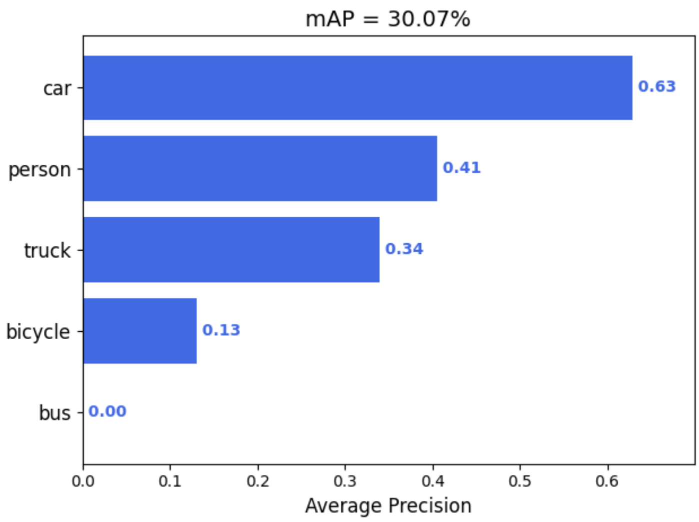

Figure 12: (c) RetinaNet

recognition. efficiency in the car detections, establishing the highest average precision (AP) for this specific class. The detection performance for buses was not that great, but it was better for trucks and trains than some of the other models. The capability of the model of providing proper tackling of pixels that belonged to the item enabled the precise forecasts of the bounding rectangle. However, a major drawback of Mask R-CNN is its long detection time. which is approximately 24s for all potential items in an image.

Specifically for RetinaNet, the "car" class attains the highest AP among the six classes. This paper argues that the detection of buses and trains is relatively inadequate. The Recognition of trucks is much better than in the case of buses and trains. It has been observed from the results that RetinaNet has better detection accuracy, specifically for bicycles as compared with SSDNet. Out of the three models, Mask R-CNN had the highest level of detection accuracy as far as persons are concerned, with almost perfect coordinates to the bounding boxes.

6.1.2 Other YOLO models

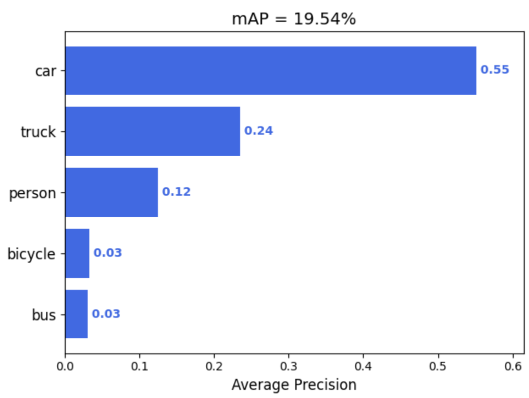

Figure 13: (a)YOLOv3

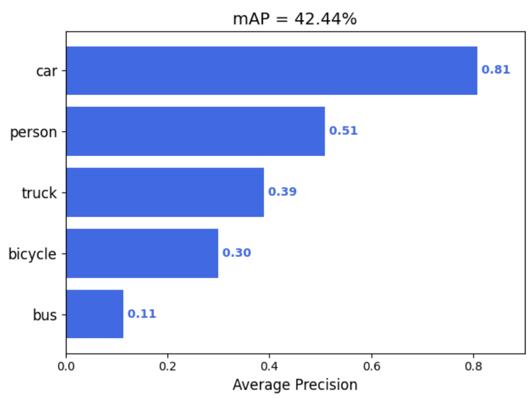

Figure 14: (b)YOLOv4

The YOLOv3 system was most effective in identifying cars, buses, trucks, and trains. person, but was effective only to some extent to detect bicycles, and the reason is due to their frontal and rear views. It was also seen that the accuracy of sample detection and the number of samples detected within an image was significantly high. The average The time to detect the images was approximately 4.49 seconds per image for the first 1000. images of the KITTI dataset. The YOLOv4 model had better performance in detecting the heavy-duty vehicles than the yolov3, especially on the identification of trucks. As for the drawbacks of YOLOv3, namely, it cannot recognize objects located at a distance from the camera and objects that are located in a small part of the image, YOLOv4 eliminated them. For the YOLOv4 model, when we ran 50 images of the first KITTI dataset, it had a mean detection time of approximately 6.6 seconds. This detection time is fairly short. as compared to the other models that were discussed, if we consider acceptable detection accuracy.

6.1.3 New YOLO models

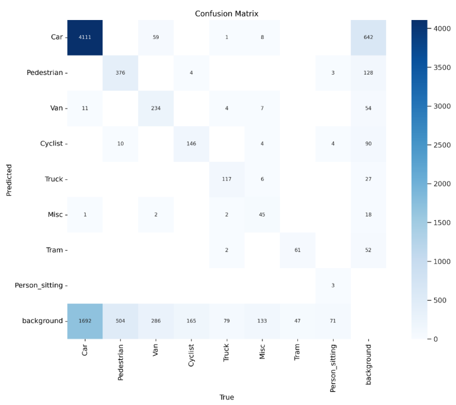

Figure 15: (a)YOLOv10

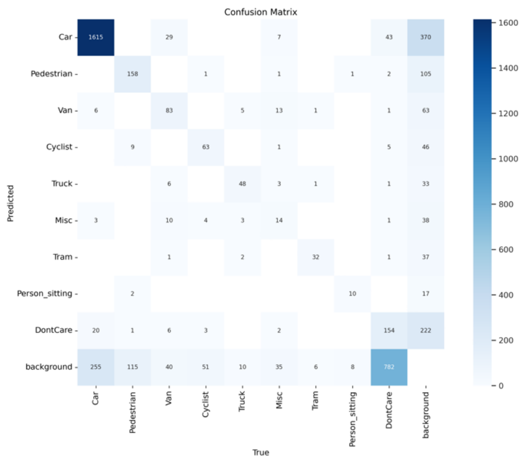

Figure 16: (b)YOLOv11

The model draws bounding boxes around multiple cars, accompanied by confidence scores that indicate the model's level of assurance in its detections. The confidence scores exhibit variation among the detections. Some detections yield scores that approach 1.0, indicating a high level of confidence that the detected object is a car. lesser scores in other cases indicate a lesser level of certainty in the model's classifications. While the majority of bounding boxes are expected to contain automobiles, there is a possibility that a few may include objects that have been incorrectly recognized as cars due to the constraints of the model. For example, items that have comparable forms to vehicles, such as SUVs or vans that are parked at an incline may be categorized incorrectly. Although YOLOv10 may identify certain pedestrians; the confusion matrix indicates that there is a possibility of a considerable amount of undetected instances or incorrect categorizations for this category. Similar to pedestrians, the model may encounter difficulties in reliably detecting van, truck, and cyclist objects in the dataset. The minimal values on the diagonals for the classes Misc, Tram, and Person sitting indicate that YOLOv10 has a low success rate in accurately detecting these things. According to the confusion matrix, the Yolov11 model exhibits a strong performance. in identifying certain classes like Car in the KITTI dataset, due to the presence of a significantly large number of true positives.

We can see that in the car section the true positive is 1615, which is quite a high score. YOLOv11 effectively detects cars in the scene. It draws bounding boxes around multiple cars with high confidence scores. The Scores range from 0.7 to 1.0, indicating a strong level of assurance in its predictions. Even in scenarios where multiple cars are present very close to each other and somewhat overlapping, each car is separately identified by its own bounding box with a high degree of certainty. Like YOLOv3, YOLOv11 also struggles with the problem of recognizing items. that are far from the camera or occupy a very small area in the image.

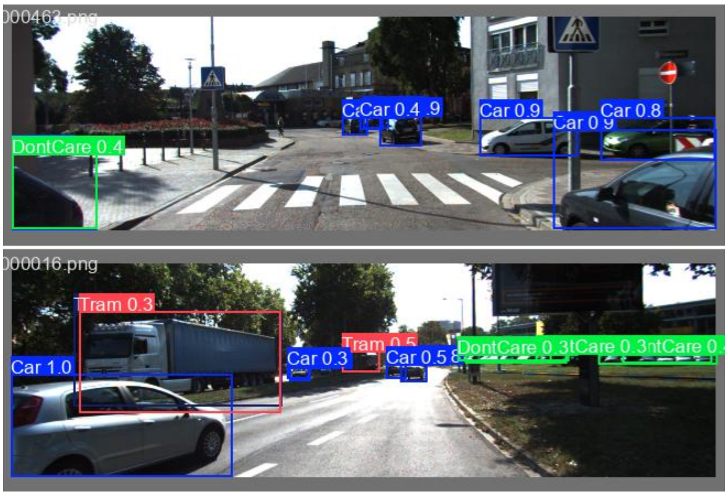

Figure 17: Real time predictions using YOLOv11

YOLOv11 performs better on KITTI data because of main two reason :

- • It uses advanced loss functions like CIoU, which improve the alignment between predicted and ground-truth boxes by considering the overlap and distance between centers, and scale. Its classification loss handles imbalanced data better, and its confidence loss enhances object detection reliability.
- • It's updating the advanced backbone with CSONet, which performs better features. traction on the KITTI dataset.

The main loss function can expressed as

$$L = L_{box} + L_{cls} + L_{conf}$$

where

- **L_box :** Localization loss (e.g., DIoU or CIoU) for better alignment between predicted and ground-truth boxes.
- **L_cls :** Classification loss (e.g., binary cross-entropy or focal loss) for better hand link of imbalanced data.

$$L_{DIoU} = 1 - IoU + \frac{\rho^2(b, b_{gt})}{c^2}$$

- **L_conf :**Confidence loss to improve object presence prediction.

In yolov11 model CIoU loss is improved as :

Further, it combines dynamic anchors for fast and proper matches between predictions. and ground truth, and utilizes augmentation techniques such as mosaic and mix-up. These enhancements result in reduced training and validation losses and increased precision, recall, and mAP that qualifies it to perform better on the KITTI complex situations.

...

6.2 Future Vehicles Forecasting

The real time data was applied to the system

Algorithm	RMSE	MSE	MAPE
ARIMA	52.34	7.78	19.45
STL ARIMA	26.78	5.12	8.98
LSTM	10.56	3.39	5.45

The real-time model performed well with LSTM, but the execution time was higher. while the STL ARIMA is the most efficient. In real time, the green signal timing can be bucketed and deployed into the system.

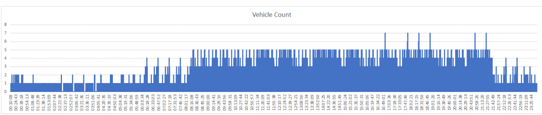

Figure 18: vehicle count

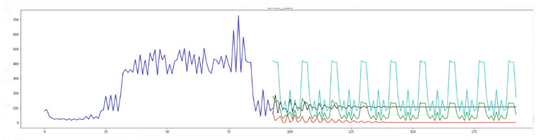

Figure 19: prediction of stl arima

...

6.3 Cloud Framework

The real-time solution executes the number of vehicles detected using the YOLOv11 which hasn't been deployed into any of the solutions as per the research paper since It's the latest addition to the YOLO models.

Green Time=15 Seconds

...

...

6.4 Discussion

There are several object detection models implemented in this work and how these models performs for several object detections, like vehicles, bicycles, and pedestrians. The

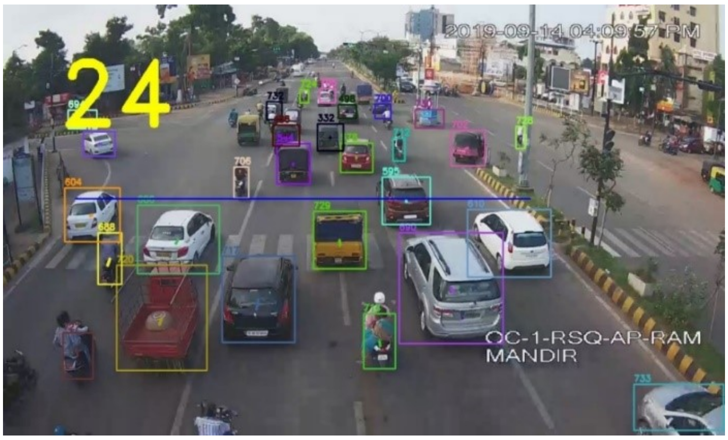

Figure 20: Vehicle Counting

comparison was made with older models, SSDNet, Mask R-CNN, RetinaNet, and newer YOLO versions of algorithms, including YOLOv3, YOLOv4, YOLOv8, YOLOv11, and YOLOv10. The older models, such as SSDNet and Mask R-CNN, had good detection in the car object but poor detection of small or distant objects such as bicycles and buses. On the other hand, more recent YOLO versions, particularly YOLOv8 and YOLOv11, performed comparatively better, in terms of accuracy as well as detection times, thus making the former especially suitable for complex scenarios owing to its more advanced loss functions, as well as its better feature extractor.

This paper demonstrates how researchers face the problem of choosing the best model. for the application with less training time but a high accuracy rate while using the application that grasps the real-time data like the vehicle detection and forecasting. This analysis consequent can be used in traffic forecasting and in smart city planning technologies.

7 Conclusion and Future Work

It is important to know that each model has particular strengths and weaknesses. As for SSDNet, it has high velocity but relatively lower precision and has issues regarding certain objects' identification. YOLOv3 shows fairly good response and accuracy, especially in recognition of usual entities on road scenes, but it fails to recognize bicycles. The Mask RCNN yields high localization accuracy, especially for detecting individuals and vehicles, but that which comes with the slow speed of the algorithm makes it unsuitable for real-time applications. However, YOLOv4 improves the disadvantage of YOLOv3. and has higher speed than Mask RCNN, but is also limited by speed. The YOLOv8 and YOLOv10 models show the best promise out of all models, and while YOLOv11 has a slight edge when it comes to precision and recall with object and pedestrian detection. However, both models still fail to detect some categories perfectly well as we are about to show in a subsequent section.

Thus, in applied problems, for instance, in traffic organization, the choice of the model must combine the required degree of detection and the time taken to analyze it. Thus, YOLOv4, YOLOv8, and YOLOv10 are the most prospective choices, while YOLOv11 has shown a slightly better result in terms of total performance. However, the final choice will depend on some specific requirements for the application, such as, for instance, the requirement for the immediate search and the need for correct recognition of some certain types of the objects.

References

Bartlett, Z. E. (2023). The prediction and mitigation of road traffic congestion based on machine learning.

Elsagheer Mohamed, S. (2019). Automatic traffic violation recording and reporting system to limit traffic accidents : Based on vehicular ad-hoc networks (vanet), pp. 254–259.

Ezekwere, S. E., Anireh, V. and Daniel, M. (2020). Application of the spider monkey optimization algorithm in a class of traffic delay problem, *SSRG Int. J. Comput. Sci. & Eng.(SSRG-IJCSE)* **7**(2): 48–55.

Ghahremannezhad, H., Shi, H. and Liu, C. (2023). Object detection in traffic videos: A survey, *Trans. Intell. Transport. Sys.* **24**(7): 6780–6799.
URL: *https://doi.org/10.1109/TITS.2023.3258683*

H R, D., Madhav, A. and Tyagi, A. (2022). *Traffic Prediction Using Machine Learning*, pp. 969–983.

Hui, J. (n.d.). mAP (mean Average Precision) for Object Detection — jonathan-hui.medium.com, `https://jonathan-hui.medium.com/map-mean-average-precision-for-object-detection-45c121a31173`. [Accessed 23-12-2024].

Izadi, S., Ahmadi, M. and Nikbazm, R. (2022). Network traffic classification using convolutional neural network and ant-lion optimization, *Computers and Electrical Engineering* **101**: 108024.
URL: *https://www.sciencedirect.com/science/article/pii/S0045790622002889*

Kumar, K., Kumar, M. and Das, P. (2024). Traffic congestion forecasting using multilayered deep neural network, *Transportation Letters* **16**(6): 516–526.

Mohammed, Z. A., Gheni, H. Q., Hussein, Z. J. and Al-Qurabat, A. K. M. (2024). Advancing cloud image security via aes algorithm enhancement techniques, *Engineering, Technology & Applied Science Research* **14**(1): 12694–12701.

Ntakolia, C. and Lyridis, D. (2022). A n d ant colony optimization with fuzzy logic for air traffic flow management, *Operational Research* **22**: 1–19.

Ultralytics (n.d.). YOLO11 NEW — docs.ultralytics.com, `https://docs.ultralytics.com/models/yolo11/`. [Accessed 23-12-2024].

What is a Smart Traffic Management System? — symmetryelectronics.com (n.d.). `https://www.symmetryelectronics.com/blog/what-is-a-smart-traffic-management-system/`. [Accessed 23-12-2024].

Xu, X., Liu, C., Zhao, Y. and Lv, X. (2022). Short-term traffic flow prediction based on whale optimization algorithm optimized bilstm_attention, *Concurrency and Computation: Practice and Experience* **34**(10): e6782.
URL: *https://onlinelibrary.wiley.com/doi/abs/10.1002/cpe.6782*

Zarindast, A., Poddar, S. and Sharma, A. (2022). A data-driven method for congestion identification and classification, *Journal of Transportation Engineering, Part A: Systems* **148**.