Density-based Automatic Traffic Control using Machine Learning

Pradip Kumar Barik, Kishan Munjpara, Anupam Gevariya, Hil Mangukiya, Misha Kotadiya, and Niti Mulani
Department of ICT, Pandit Deendayal Energy University, Gandhinagar, India
Email: pradip.barik@sot.pdpu.ac.in, kishan.mict20@sot.pdpu.ac.in, anupam.gict20@sot.pdpu.ac.in, hil.mict20@sot.pdpu.ac.in, misha.kict20@sot.pdpu.ac.in, niti.mict20@sot.pdpu.ac.in

***Abstract*—The existing traffic light control systems suffer from inefficiencies due to the utilization of predefined methods based on offline data, resulting in long delays and energy wastage. Insufficient planning of transportation infrastructure and policies contributes to high traffic congestion, making it crucial to estimate traffic density accurately. This research aims to enhance static traffic signal systems by implementing a dynamic system using machine learning techniques to measure traffic density. The primary focus is on four-way junctions, as high potential for traffic congestion. This study aims to improve traffic management efficiency by incorporating real-time traffic information, object detection, and machine learning-based traffic density prediction. The proposed technique demonstrates its efficiency in managing real-time traffic by utilizing cameras for real-time vehicle detection and analyzing previous traffic conditions. The findings highlight the potential of a few approaches to optimize traffic signal control and alleviate congestion, addressing the increasing challenges possessed by urban traffic and promoting more intelligent and adaptive traffic management systems.**

***Index Terms*—Traffic congestion, Traffic density estimation, Real-time traffic information, Object detection, YOLOv8**

I. INTRODUCTION

Urban regions are now faced with an enormous obstacle of traffic congestion, that results in significant economic losses and lowers the standard of living for locals. Traffic control systems have historically depended on set signal timings and pre-established traffic management strategies to reduce congestion and improve traffic flow. However, these systems usually fail to respond to changing traffic circumstances efficiently, resulting in inadequate traffic management and severe congestion [1]. The availability of large amounts of traffic data and recent developments in machine learning algorithms have opened up new possibilities for creating intelligent traffic control systems. Density-based autonomous traffic control is a potential method that uses real-time data on traffic density to make adaptive decisions about the timings of traffic signals [2].

The You Only Look Once (YOLO) method and other deep learning models have revolutionized object recognition in computer vision tasks. In particular, YOLOv8 has shown exceptional performance in real-time object localization. A density-based traffic control system can be created to analyze traffic conditions in real time and optimize traffic signal timings by utilizing the capabilities of YOLO version 8 and machine learning ideas. Fig. 1 shows a common scenario of traffic congestion problem in urban areas.

The YOLOv8 model is used in the proposed density-based traffic control system to find and track moving objects in video streams from security cameras or other sources. Training the YOLO algorithm on a sizable collection of traffic recordings will be required for the study in order to achieve precise vehicle detection. After that, intelligent decision-making techniques for traffic signal optimization will be developed using machine learning algorithms to analyze the patterns of detected vehicle density.

Fig. 1. Traffic congestion in urban areas

By contrasting it with conventional fixed-time traffic control systems, the research will assess the effectiveness of the density-based traffic control system. This study's findings have important implications for the development of intelligent traffic control systems. The suggested density-based traffic control system has the potential to revolutionize traffic management, reduce congestion, and increase overall transportation efficiency by leveraging YOLO version 8 machine learning techniques.

The main contribution of this paper are as follows:

1) We have proposed a technique for the automatic traffic control using popular object detection algorithm YOLOv8.
2) By leveraging the YOLO object detection model, we are able to accurately detect vehicles in each lane of a junction.
3) We have also computed vehicle density based on the total number of vehicles detected in each lane. This metric provides a quantitative measure of traffic density, which can be used to assess congestion levels and allocate appropriate resources.
4) Our proposed scheme dynamically adjusts traffic signal timings based on the density of vehicles in each lane by effectively prioritizing more congested lanes and optimizing traffic flow.

The rest of the paper is organized as follows: Section II provides a comprehensive study of existing methods of traffic control algorithms and their limitations. The methodology used for data collection and analysis and the implementation details of the density-based traffic control system using YOLO version 8 machine learning algorithm is discussed in Section III, We discuss the experimental results and their implications in Section IV. Finally, conclusions are given in Section V.

II. LITERATURE SURVAY

Traffic management is essential for controlling the flow of traffic on the roadways and increasing overall transportation effectiveness. A growing number of people are interested in using machine learning techniques to automate traffic control systems as a result of technological improvements. By analyzing and processing enormous volumes of traffic data, machine learning algorithms have the ability to make wise decisions that will improve safety, reduce congestion, and optimize traffic flow. This literature review attempts to give a summary of the current research on machine learning-based autonomous traffic control.

Accurate traffic control depends on the ability to predict traffic flow. Machine learning algorithms have been explored in several research as a potential tool for predicting traffic patterns. In order to estimate traffic flow patterns based on historical data, weather conditions, and other pertinent aspects, techniques including Artificial Neural Networks (ANN), Support Vector Machines (SVM), and Random Forests (RF) have been used. Traffic signal timing can be optimized using these forecasts, and signal designs can be changed on the fly. With the goal of enhancing traffic signal control and lessen traffic congestion, machine learning techniques have been used. On the basis of traffic patterns and historical data, reinforcement learning algorithms such as Q-learning and Deep Q-learning have been used to learn the ideal traffic light timings. These algorithms can respond to changes in traffic conditions and enhance signal layouts over time while reducing delays [1,2].

To create intelligent traffic management systems, machine learning techniques have been used. These systems analyze and forecast traffic patterns using a variety of data sources, including GPS data, video surveillance, and real-time traffic sensors. These systems can accurately decide on the timing of traffic signals, lane control, and incident detection with the use of machine learning algorithms, resulting in more effective traffic management. For efficient traffic control, precise vehicle detection and tracking are essential. Vehicles have been detected and tracked in real-time using machine learning algorithms, notably computer vision methods like Convolutional Neural Networks (CNN). By using this data, signal timings may be improved, traffic bottlenecks can be found, and adaptive traffic control measures can be put into place. For better traffic control capabilities, machine learning has been used into ITS. For traffic forecasting, incident detection, route optimization, and adaptive signal management, ITS uses machine learning techniques. These technologies can dynamically change traffic flow and lessen congestion by analyzing large-scale traffic data. In order to model and improve traffic flow in virtual settings, machine learning approaches have been used. Researchers may test and assess various traffic control solutions using agent-based modelling and simulation tools in conjunction with machine learning algorithms. These simulations aid in comprehending how different variables, such as signal timings, road layouts, and driver behavior, affect traffic flow [3,4,5].

Many traffic analysis projects now use deep learning methods. Recurrent neural networks (RNN) and convolutional neural networks (CNN) have been employed for tasks including vehicle categorization, pedestrian detection, and traffic sign recognition. By delivering precise and timely information about the traffic environment, these models can handle complicated visual input and enhance traffic management systems. To improve traffic management systems, researchers have been investigating the integration of various data sources. This entails combining information from mobile devices, social media, weather forecasts, and traffic sensors. These many data sources are analyzed and combined using machine learning algorithms, giving a more thorough knowledge of traffic patterns and facilitating more efficient traffic management decision-making. Prompt detection and action are essential for traffic regulation. Real-time event detection using machine learning algorithms has been done utilizing data from traffic sensors, social media, and surveillance cameras. To spot unusual traffic patterns, accidents, and roadblocks, methods including anomaly detection, pattern recognition, and clustering algorithms have been used. This information can help traffic control officials act quickly to reduce disturbances and guarantee the safety of road users [5,6].

Traditional traffic signal management systems frequently follow set time schedules, which may not be the best option when traffic circumstances change. Machine learning techniques allow for adaptive traffic signal control, in which the timing of the signals is changed in real-time in response to the flow of traffic. Deep Q-networks (DQN) and Proximal Policy Optimization (PPO), two reinforcement learning algorithms, have been used to enhance the efficiency of traffic flow and optimize signal timings. These adaptive control systems seek to enhance overall traffic performance by reducing congestion and trip times. Although many machine learning algorithms have shown favorable results in research settings, scaling and implementing them in actual traffic management systems presents difficulties. When implementing machine learning algorithms in massive traffic networks, scalability and computing efficiency are crucial aspects to take into account. Additionally, for dependable and safe traffic control systems, establishing resilience against uncertainties, such as changing traffic patterns, unanticipated circumstances, and adversarial assaults, is crucial [7].

Machine learning algorithms provide useful insights and decision-making skills for a variety of applications, including traffic flow prediction, signal control optimization, intelligent transportation systems, and traffic simulation. To ensure the viability and efficiency of machine learning-based traffic management solutions, more research is needed to solve issues with scalability, real-time implementation, privacy, and security.

III. PROPOSED METHODOLOGY

We have created a system to improve the current methods used in this field after carefully examining many active projects and previously proposed frameworks for the efficient operation of traffic signals. Effective Real-Time Traffic Management Using Machine Learning Algorithm is our suggested methodology.

A. Machine Learning model (YOLO) for object detection

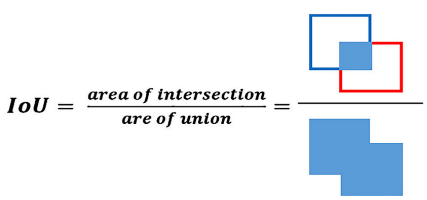
Each grid cell in the model predicts B bounding boxes along with their corresponding confidence ratings. These ratings indicate the model's level of certainty regarding the presence of an object within the box and the accuracy of its prediction. The confidence is calculated by multiplying the probability of an object being present (Pr(Object)) with the intersection over union (IOU) value between the predicted box and the actual data (truthPred). When there is no object present in the cell, the confidence rating is expected to be 0. Conversely, if an object is present, the confidence score should align with the IOU. Furthermore, the model also estimates C conditional class probabilities $\Pr(C_i|\text{Object})$ for each grid cell.

Confidence Score:

$$\begin{aligned}&\Pr(Class_i|\text{Object}) \times \Pr(\text{Object}) \times \text{IOU}^{\text{truthPred}}\\&= \Pr(\text{Object}) \times \text{IOU}^{\text{truthPred}}\end{aligned} \tag{1}$$

where,

$\Pr(\text{Class}_i|\text{Object})$ = Probability of ith Class for given object



A distinguishing feature of YOLO, which was developed in 2015 by Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi, is its ability to operate in real-time without sacrificing accuracy[11]. Frame object identification is handled as a regression issue and fed into a fully connected neural network by YOLO. In order to create predictions for bounding boxes, box confidence, and class probability for each grid cell, the system divides the input picture into a SxS grid. These predictions are represented by the tensor SxSx(Bx5 + C) [2]. YOLOv8 accurately detects vehicles in real-time while giving more speed of execution [12]. The variation of Mean Average Precision (MAP) for different frames per seconds and the variation of MAP based on different latency are shown in Fig. 2.

In the YOLO (You Only Look Once) algorithm, a linear activation function is utilized in the final layer, while the leaky rectified linear activation is employed in all other layers. The use of a linear activation function in the final layer is significant as it allows the model to generate output values

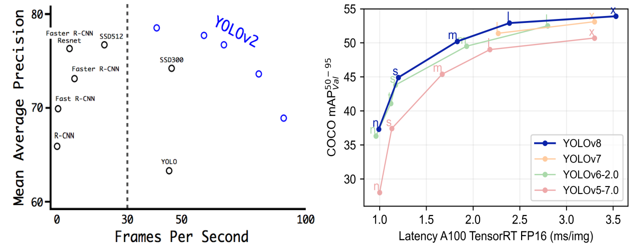


Fig. 2. *Variation of Mean Average Precision for different frames per seconds (left), Variation of MAP based on different latency (right)*

without any bounds. This characteristic is particularly crucial for object detection tasks where predictions need to span a wide range of possibilities. On the other hand, employing the leaky rectified linear activation in the other layers introduces non-linearity to the model. This non-linearity enables the model to learn intricate patterns and features from the input data, thereby improving its overall performance and its ability to accurately detect various types of objects in images. The cost function for YOLO, known as $\mathcal{L}_{\text{YOLO}}$, is defined as follows:

$$\phi(x) = \begin{cases} x & \text{if } x > 0 \\ 0.1x & \text{otherwise} \end{cases}$$

The multi-part loss function in the YOLO object detection algorithm plays a crucial role in training the model by combining different loss components to optimize object localization, class prediction, and confidence estimation. This enables more accurate and robust object detection, improving the overall performance of the algorithm.

$$\begin{aligned}\mathcal{L}_{\text{YOLO}} =& \lambda_{\text{coord}} \sum_{i=1}^{S^2} \sum_{j=1}^{B} 1_{ij}^{\text{obj}} \left[(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2\right] \\ &+ \lambda_{\text{coord}} \sum_{i=1}^{S^2} \sum_{j=1}^{B} 1_{ij}^{\text{obj}} \\ &\left[(\sqrt{w_i} - \sqrt{\hat{w}_i})^2 + (\sqrt{h_i} - \sqrt{\hat{h}_i})^2\right] \\ &+ \sum_{i=1}^{S^2} \sum_{j=1}^{B} 1_{ij}^{\text{obj}} \left(C_i - \hat{C}_i\right)^2 \\ &+ \lambda_{\text{noobj}} \sum_{i=1}^{S^2} \sum_{j=1}^{B} 1_{ij}^{\text{noobj}} \left(C_i - \hat{C}_i\right)^2 \\ &+ \sum_{i=1}^{S^2} 1_i^{\text{obj}} \sum_{c \in \text{classes}} (p_i(c) - \hat{p}_i(c))^2,\end{aligned} \tag{2}$$

Where:

- λ_{coord} and λ_{noobj} are the coordination and no-object confidence loss weightings, respectively.
- S represents the grid size.
- B represents the number of bounding boxes per grid cell.

- x_i, y_i, w_i, h_i, and C_i are the predicted values for the i-th grid cell and j-th bounding box.
- $\hat{x}_i$, $\hat{y}_i$, $\hat{w}_i$, $\hat{h}_i$, and $\hat{C}_i$ are the ground truth values for the i-th grid cell and j-th bounding box.
- 1_{ij}^{obj} and 1_{ij}^{noobj} are indicator functions indicating the presence of an object or absence of an object in the i-th grid cell and j-th bounding box.
- $p_i(c)$ and $\hat{p}_i(c)$ are the predicted and ground truth probabilities for class c in the i-th grid cell.

B. Effective Real-Time Traffic Management using ML

To ensure effective management of real-time traffic, it is essential to have access to current and accurate traffic information. One method to accomplish this is by utilizing cameras strategically positioned at intersections to obtain live video feeds. In some cases, additional cameras may be installed specifically for this purpose. Our system utilizes four lanes to record video streams and extract frames during red traffic signals, allowing for vehicle detection. These extracted frames are then processed through object detection using YOLOv8. The flow chart presented in Fig. 3 provides a visual representation of the real-time traffic management process.

The methodology for an automatic traffic management system involves the following steps. Firstly, a traffic signal camera is initialized and set up at the desired location to capture real-time traffic video footage. The camera is positioned strategically to ensure a clear view of the lanes for optimal data collection. Video frames are then captured from the camera at regular intervals, and preprocessing techniques such as resizing, normalization, and image enhancement are applied to improve frame quality. The preprocessed frames are then subjected to the YOLO algorithm for object detection. By dividing the frames into grid cells, the algorithm predicts bounding boxes and class probabilities for each cell, allowing for accurate vehicle detection. From the detected objects, vehicle bounding boxes are extracted, which enclose the vehicles' positions and provide essential information for subsequent analysis.

To assess the traffic volume and density, the number of vehicles detected in each lane is counted, and the boundaries of each lane are determined. These counts are used to calculate the vehicle density for each lane by dividing the number of vehicles by the lane width. Based on the calculated densities, the traffic signal timing is adjusted to allocate more time to lanes with higher densities. Appropriate timing strategies are implemented to optimize traffic flow and minimize congestion, taking into account factors such as traffic volume, historical data, and road conditions.

The signal status and timing for each lane are updated according to the density-based adjustments. The processed frame, with the vehicle bounding boxes and updated signal timing, is then displayed to provide a visual representation of the traffic conditions and signal adjustments. The traffic signal is continuously monitored for changes in its status, and if a change is detected, the signal status is updated, and the traffic management strategy is adjusted based on the current vehicle

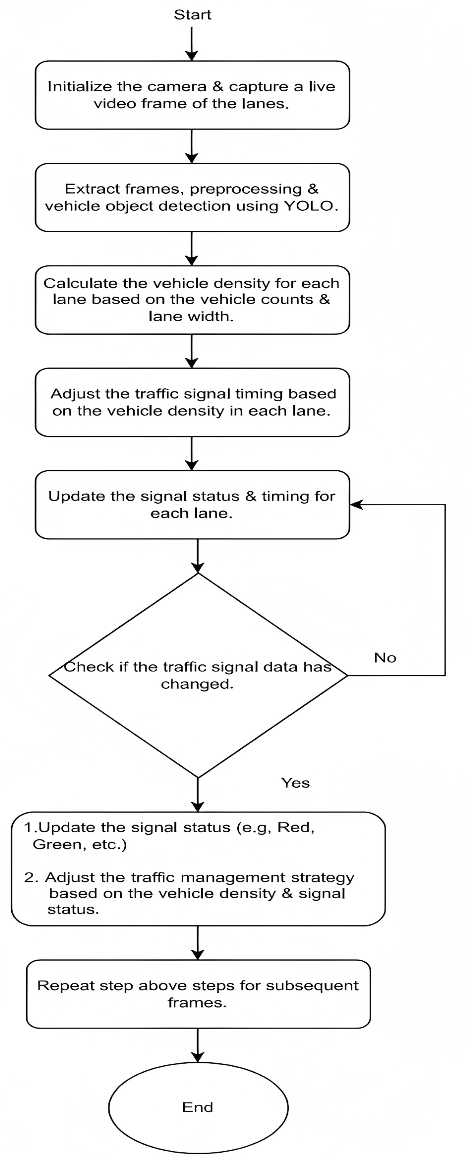


Fig. 3. Flow chart of density-based traffic control scheme

density and signal status [3]. This dynamic approach ensures prompt response to changing conditions, resulting in optimized traffic flow.
ON time of the green signal of each lane may be obtained as:

$$T_{ON} = \frac{\rho_L}{\rho_L^{max}} \times T_{ON}^{max} \text{ Seconds} \tag{3}$$

where, T_{ON}^{max} is calculated as

$$T_{ON}^{max} = \frac{\rho_L^{max}}{\rho^{max}} \times 180 \text{ Seconds}$$

Here, ρ_L is the density (no. of vehicles) in a lane. ρ_L^{max} is the maximum density among all connected lanes in a junction. ρ^{max} is the maximum density of the lanes in the city. We have considered 180 seconds as the maximum ON time of a signal (that can be adjusted manually).

Taking into account the current flow of traffic, the administration of signal lights can reduce the undesirable vehicle waiting duration. This will aid in the conservation of both time and fuel for individuals. Occasionally, the coordination of signal lights can be a complex undertaking due to unforeseen incidents or road maintenance. The effectiveness of signal lights at a specific intersection also impacts the traffic situation at the subsequent junction, making it imperative to ensure accuracy in this process. To tackle this issue, rather than relying solely on dynamic timing schedules, there may be a desire to handle the circumstances personally. As a result, the system is also devised to request human intervention when necessary.

IV. Simulation Results

This section presents the simulation results of our proposed method. We have considered data set containing different number and types of vehicle in the YOLOv8 model [13]. 2704 images have been considered for training and 300 images are used for validation. Fig. 4 shows the validation result on four images from the testing data set. The vehicle prediction for four different density lanes has been indicated using rectangular boxes on the vehicles.

Fig. 4. *Different density of predicted vehicles from street cameras. Density-1 (top-left), Density-4 (top-right), Density-5 (bottom-left), Density-7 (bottom-right)*

The count of different vehicles and prediction confidentiality from the testing data set are shown in Table I. It is seen that the confidentiality score decreases with increasing the actual number of vehicles in the image. We observe that the model predicts the near vehicles with more accuracy, but it fails to detect the vehicles when the distance increases above a certain threshold (as shown in Fig. 4).

The evaluation of traffic density takes into consideration various congestion scenarios. Fig. 5 (left figure) presents the results obtained by comparing the actual number of vehicles within a given range to the predicted number of vehicles. The plot showcases the density variance, ranging from 0.56 to 0.83, across different vehicle ranges. This variation in density can be visually observed in Fig. 5 (right figure).

TABLE I
Predicted density of vehicles in different scenarios and confidentiality score

Scenario	Actual Range	Predicted Vehicles	Total Vehicles	Confidentiality
Density1	5-10	7-cars, 1-suv	08	0.8267
Density2	11-15	1-bus, 6-cars, 2-minivans, 3-pickups, 1-suv, 1-van	14	0.5605
Density3	16-20	1-bus,16-cars, 1-suv	18	0.7103
Density4	21-25	16-cars, 1-minivan, 2-pickups, 1-suv	20	0.7324
Density5	26-30	17-cars, 2-pickups, 5-suvs, 1-van	25	0.7726
Density6	31-35	1-bus,21-cars, 5-minivans, 4-suvs, 1-van	32	0.6168
Density7	36-40	27-cars, 1-minibus, 3-minivans, 1-pickup, 5-suvs	37	0.6670
Density8	41-45	3-bus,18-cars, 4-minivan, 1-pickup, 9-suvs,1-van	36	0.6933
Density9	46-50	38-cars, 1-minivan, 3-pickups, 8-suvs, 1-three wheelers -CNG-	51	0.6074

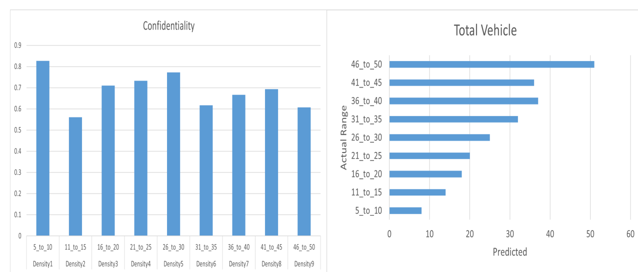


Fig. 5. *Confidentiality score for different density of vehicle in captured image (left), Predicted vehicles for different range of actual vehicle in the image (right)*

The Precision-Recall (PR) curve is a commonly used evaluation metric in machine learning, including object detection models like YOLOv8. The PR curve provides insights into the trade-off between precision (the ability of the model to correctly identify positive samples) and recall (the ability of the model to retrieve all positive samples). In the context of density-based automatic traffic control, the PR curve can be utilized to evaluate the accuracy and reliability of the YOLOv8 model in detecting and classifying vehicles. This evaluation helps in understanding the model's performance in terms of correctly identifying vehicles (precision) and capturing all relevant vehicles (recall) within a given range. The precision (P) and recall (R) values are calculated using the following formulas:

Precision (P) = True Positives / (True Positives + False Positives)

Recall (R) = True Positives / (True Positives + False Negatives)

The precision-recall curve for the YOLOv8-based prediction model is shown for different types of vehicles in Fig. 6.

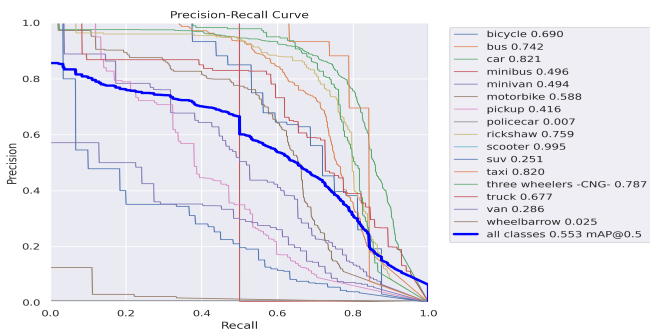


Fig. 6. Precision-Recall curve for different vehicle prediction using YOLOv8

Fig. 7 shows the traffic density of four lanes in a junction. The different types of vehicles and their count is also displayed after running YOLOv8 algorithm on the dataset. Here, Lane-4 is having the maximum density as 37, so $T_{ON} = 131$ seconds as per calculation using (3). Based on the traffic density, Lane-

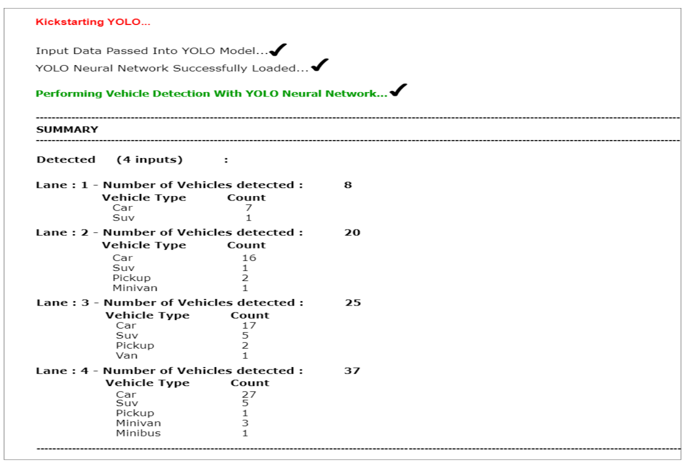


Fig. 7. Density prediction of four lanes using YOLOv8

4 is open (green signal) for 131 seconds. After that, the lane will be closed (red signal) before going to the next step of density prediction as shown in Fig. 8.

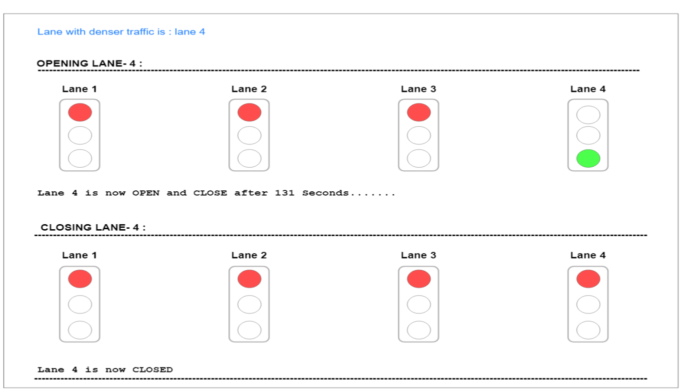


Fig. 8. Density-based traffic control of four lanes

V. Conclusion

This paper provides an efficient and cost effective solution for traffic congestion control using machine learning algorithms, emphasising the advantages of real-time traffic monitoring and accurate object detection with YOLOv8. Significant benefits, such as improved traffic flow, decreased congestion, and increased transportation efficiency, are provided by the incorporation of machine learning techniques in traffic management systems. The results highlight the significance of utilising cutting-edge technologies to improve traffic control tactics and pave the path for more intelligent and effective transportation systems.

The potential of density-based traffic regulation utilising machine learning must be acknowledged, though, as it will require additional research and development for read-time hardware implementation.

References

[1] M. R. Usikalu et. al., "Design and Construction of Density Based Traffic Control System", *International Conference on Energy and Sustainable Environment*, 2029. doi:10.1088/1755-1315/331/1/012047

[2] Jondhale S.S., Shingote Vijaya R., Badhe Tanuja M., Inamdar Safiya A., Temgire Sonali B., "Machine Learning Based Automatic Traffic Control System.", *International Journal of Emerging Technologies and Innovative Research*, ISSN:2349-5162, Vol.7, Issue 1, page no.297-300, January 2020.

[3] Sayed A. Sayed, Yasser Abdel Hamid and Hesham Ahmed Hefny, "Artificial intelligence-based traffic flow prediction: a comprehensive review.", in *Journal of Electrical Systems and Information Technology*, Vol. 10, no. 13, March 2023. doi: 10.1186/s43067-023-00081-6

[4] Josin Hippolitus a, Namita Joanna Victor, Ronak Bogavalli, Sanjay S And Aththen Premkumar, "Traffic Signal Control Using Machine Learning," *International Journal of Mechanical and Production Engineering Research and Development (IJMPERD)* Vol. 10, Issue 3, Jun 2020, 10849–10872

[5] N. O. Alsrehin, A. F. Klaib and A. Magableh, "Intelligent Transportation and Control Systems Using Data Mining and Machine Learning Techniques: A Comprehensive Study," in *IEEE Access,* vol. 7, pp. 49830-49857, 2019, doi: 10.1109/ACCESS.2019.2909114.

[6] A. Gomaa, T. Minematsu, M.M. Abdelwahab et al., "Faster CNN-based vehicle detection and counting strategy for fixed camera scenes," in *Multimedia Tools and Applications*, vol. 81, pp. 25443–25471, 2022. https://doi.org/10.1007/s11042-022-12370-9

[7] T. Sakaki, Y. Matsuo, T. Yanagihara, N. P. Chandrasiri and K. Nawa, "Real-time event extraction for driving information from social sensors," *IEEE International Conference on Cyber Technology in Automation, Control, and Intelligent Systems (CYBER)*, Bangkok, Thailand, 2012, pp. 221-226, doi: 10.1109/CYBER.2012.6392557.

[8] M. Nama, A. Nath, N. Bechra, J. Bhatia, S. Tanwar, M. Chaturvedi, B. Sadoun, "Machine learning-based traffic scheduling techniques for intelligent transportation system: Opportunities and challenges," in *International Journal of Communication Systems, Wiley Online Library* , vol. 34, issue 9, 2021. https://doi.org/10.1002/dac.4814

[9] Kim D, Jeong O. "Cooperative Traffic Signal Control with Traffic Flow Prediction in Multi-Intersection," *Sensors (Basel)*, vol. 20, no. 1, December 2019 doi: 10.3390/s20010137.

[10] YOLO - object detection, Raphael Holzer Revision, 2019. [Online]. Available: https://opencv-tutorial.readthedocs.io/en/latest/yolo/yolo.html

[11] S. Prasad and S. Sinha, "Real-time object detection and tracking in an unknown environment," *World Congress on Information and Communication Technologies*, Mumbai, India, 2011, pp. 1056-1061, doi: 10.1109/WICT.2011.6141394.

[12] M. Antonakakis et al., "Real-Time Object Detection using an Ultra-High-Resolution Camera on Embedded Systems," *IEEE International Conference on Imaging Systems and Techniques (IST)*, Kaohsiung, Taiwan, 2022, pp. 1-6, doi: 10.1109/IST55454.2022.9827742.

[13] Road Vehicle Images Dataset, Kaggle, 2021. [Online]. Available: https://www.kaggle.com/datasets/ashfakyeafi/road-vehicle-images-dataset?datasetId=1644010