Multimedia Tools and Applications
https://doi.org/10.1007/s11042-024-19331-4

Smart traffic control: machine learning for dynamic road traffic management in urban environments

Hameed Khan[1] · **Jitendra Singh Thakur**[1]

Received: 1 February 2024 / Revised: 19 April 2024 / Accepted: 1 May 2024

Abstract
Roadside and outside environmental elements contribute to the road traffic setting's highly dynamic and turbulent nature. The human factor, primarily disregarded in the present research, is an essential element that contributes to the traffic context in addition to infrastructure-related elements like traffic signals, road infrastructure, and other road networks. Timing the green light and tracing the object that makes the incorrect turn using real-time visual information for traffic monitoring are still challenging tasks for the conventional traffic control system. We describe a self-adaptive real-time algorithm based on real-time traffic flow and monitoring. Combining image processing with AI-powered, self-adaptive machine learning for controlling traffic clearance at intersections is a forward-thinking approach with great potential. The suggested system uses the You Only Look Once v3 (YOLOv3) model and single image processing using a neural network to determine traffic clearance at the signal. YOLOv3 method to recognize objects from video frames. Subsequently, the centroid object tracking technique is used to monitor the movement of each vehicle within a proposed framework. We implemented algorithms to identify vehicles traveling in the incorrect direction based on their trajectories. This integrated approach enhances accurate object recognition, real-time vehicle tracking, and the detection of traffic violations, enhancing overall road safety measures. The experimental findings are quite promising, achieving an exclusive comparison between expected and actual vehicle numbers is crucial for any traffic monitoring system. The average object detection accuracy of 88.43% is impressive, and the exceptional 90.45% accuracy in tracking vehicles engaging in wrong turns or reckless driving behaviors is particularly noteworthy—it provides the system's ability to address safety concerns effectively. Integrating a Convolutional Neural Network (CNN) into the algorithm to alleviate traffic congestion at intersections is a smart move. CNNs are known for their effectiveness in image processing tasks, making them well-suited for tasks like object detection and tracking in complex environments like intersections.

Keywords Traffic optimization · Machine-learning driven · YOLOv3 · Multi-object tracking · Open CV

Extended author information available on the last page of the article

Published online: 14 May 2024

1 Introduction

The enormous number of automobiles on the roads in smart cities makes traffic congestion a severe problem. A traditional traffic management system adjusts the green light's duration based on the average traffic volume at the junction. Enhancing and improving vehicle management systems can be achieved through various methods, including infrastructure upgrades, public transportation expansion, traffic enforcement, education initiatives, and smart parking solutions. However, effectively managing road traffic issues often necessitates the implementation of a smart traffic management system (STMS) [1]. These systems utilize advanced technologies such as artificial intelligence, machine learning, and real-time data analytics to optimize traffic flow, reduce congestion, and enhance overall efficiency on the roads. By dynamically adjusting traffic signals, analyzing real-time traffic data, and employing predictive modeling techniques, STMS can revolutionize how traffic is managed in urban environments. Indeed, considering their extensive usage, traditional traffic signal systems sometimes confront numerous well-known issues in many nations, even industrialized ones. Here some potential research gaps are included in the field of smart traffic management systems:

(a) Conventional traffic signal systems with fixed timing could be more efficient at managing traffic signal timing at road junctions. Signal timings that are set rather than dynamically altered depending on real-time traffic circumstances may have several negative implications, including increased wait time, traffic congestion, air pollution, and safety issues.
(b) In junctions, the proportion of time allotted to each route is fixed, irrespective of the volume of traffic or flow rate. Because time is not distributed in the real-time traffic scenario, this results in inefficient traffic flow. Certain times of the day may see more traffic on particular routes than others, necessitating a longer time to clear the congestion. However, this capability is not available with a regular traffic light.
(c) Conventional traffic management systems depend on permanent infrastructure, such as traffic signals, signs, and cameras, to control traffic flow and enforce laws. While these systems may adequately manage regular traffic scenarios, they generally need help handling dynamic occurrences such as cars traveling the wrong routes or participating in reckless driving. In circumstances where a vehicle travels the wrong route or participates in reckless driving, standard traffic control systems may not be able to react in real-time or effectively handle the problem.

Based on the previously described issues of the conventional traffic management system, it is evident that there is a tremendous need to enhance its functioning by giving it some intelligence to allow dynamism and adaptability. Adaptive traffic signal control provides a technique that modifies traffic lights in real time depending on traffic patterns. The technology employs sensors to determine the number of cars on the route and changes traffic lights accordingly. Adaptive traffic signal management may minimize congestion, enhance traffic flow, and cut travel times. In addition to improving traffic flow and adaptable lane management, smart traffic control systems apply intelligent route optimization to handle dynamic occurrences such as vehicles choosing the wrong direction or participating in reckless driving.

The precise research contributions are:

- Providing a self-adaptive traffic control system algorithm (TCSA) approach for controlling traffic lights in real-time scenarios, which relies on real-time traffic flow evaluation and monitoring.
- In a sophisticated traffic management system, we design a framework based on Line_1 and Line_2 approaches with three imaginary lanes. The rules are meticulously designed to ensure safety, efficiency, and a smooth traffic flow.
- The entire dataset has been uploaded to GitHub for further research work in the field of traffic optimization.

Machine learning for dynamic road traffic management in urban areas aims to reduce traffic congestion, improve road safety, optimize resource allocation, integrate public transportation systems, and provide real-time traffic information and individual direction. Smart traffic management systems may solve these difficulties using sophisticated machine learning methods to build more efficient, sustainable, and livable urban transportation networks.

The proposed work is self-adapting TCSA approach for traffic optimization. The solution indeed outlines a promising approach to addressing the challenges in traffic management using machine learning, specifically the YOLO object detection algorithm. By leveraging real-time images from traffic signals and processing individual images rather than entire video streams, the solution aims to be more efficient in terms of computational resources and time [2]. Enhancing traffic optimization algorithms with predefined traffic rules is a crucial step towards making traffic management systems more intelligent. By incorporating speed limits, lane regulations, traffic signal timing, and right-of-way rules, the system can better allocate resources and prioritize traffic flow. Utilizing computer vision techniques, possibly in combination with machine learning algorithms, to track objects in real-time video feeds from traffic cameras is a powerful approach. Object tracking algorithms can help monitor the movement of vehicles, pedestrians, and cyclists, providing valuable data for traffic management decisions. Analyzing the behavior of tracked objects to determine whether they are following or disobeying traffic rules is a significant advancement. By detecting actions such as running red lights, speeding, illegal lane changes, and pedestrian jaywalking, the system can take proactive measures to improve safety and efficiency on the roads. Combining image processing and machine learning technologies for controlled traffic clearing at intersections represents an innovative approach to traffic management. This integration allows for real-time analysis of traffic conditions and efficient decision-making processes. The suggested method establishes traffic clearing at road intersections using the object detection model (YOLOv3) and artificial intelligence neural networks for single and multiple image processing. These framework encompassed metrics such as rapid and mean average precision (mAP), processing time, high accuracy, intersection over union (IOU) values, and speed. By optimizing these metrics, we aimed to achieve superior performance in object identification tasks, ensuring efficient and accurate detection of objects in real-time scenarios. We examined the "on-time period green light" based on YOLO's advice to determine the best traffic-clearing strategy. The "on" duration of the green light is calculated by considering many variables, such as the number of two-wheelers (TW) and four-wheelers (FW) on the road, its breadth, and the moment when vehicles pass a junction. In the context of intelligent traffic management, computer vision systems can analyze real-time video feeds from traffic cameras to monitor traffic flow, detect congestion, and identify traffic violations such as running red lights or illegal lane changes. These systems can provide valuable data for optimizing traffic flow and enhancing road safety by tracking moving objects such as vehicles, pedestrians, and cyclists. Similarly, in security surveillance applications, computer vision technologies are essential for monitoring and identifying moving objects in real-time. By detecting suspicious activities or unauthorized intrusions, these systems can alert security personnel and take appropriate action to mitigate potential threats. This study is organized as follows: In Section 2, which summarizes the relevant work in this area, we start by looking at an overview of the current literature. In Section 3, we explore the complexities of our traffic simulation and proposed framework, which are critical parts of our framework for intelligent traffic management. The system architecture, methodological foundation, and algorithmic nuances of our proposed system are all explained in Section 4. Next, we provide a thorough simulation and analysis of the performance metrics obtained from our innovative strategy in Section 5. This section explains how well and efficiently our system manages various traffic situations. Section 6 summarizes our study's conclusions and future developments in intelligent traffic control systems as we wrap up our discussion.

2 Related work

In numerous regions, traffic congestion poses a significant concern. With countless objects occupying roads on a daily basis, managing traffic becomes increasingly difficult. Traffic management systems researchers have conducted extensive studies to enhance flexibility, intelligence, and self-adaptability. They have installed cameras at highways and intersections to actively monitor and enforce autonomous penalties for traffic regulation violators. While most traffic intersections use a fixed-time green light cycle for traffic control, this approach has drawbacks, such as ineffective traffic congestion reduction. The standard time-limited signal system functions well when traffic flow is nearly identical in all four directions. However, advanced strategies are required to address more complex traffic scenarios. During the daytime, traffic tends to be heavier in one direction compared to others. However, the existing traffic management method lacks intelligence, leading to unnecessary waiting times even when there are no vehicles on the opposite side. This inevitable waiting period can cause frustration among drivers, potentially resulting in rule infractions and accidents. Additionally, it contributes to increased fuel usage and environmental pollution. To address these issues, several research studies have focused on tackling traffic congestion and managing traffic flow more effectively. One advanced approach involves making traffic signals responsive to current traffic patterns. By utilizing real-time data on traffic volume and demand, STMS can adjust signal timing dynamically. This adaptive approach aims to minimize delays and optimize traffic flow, ultimately improving overall traffic efficiency and reducing the negative impacts of congestion on both drivers and the environment [3].

A sophisticated traffic management system employs a blend of automated mechanical and electronic devices to regulate traffic flow effectively. This includes sensors for monitoring traffic volume and vehicle movements, automated signal controllers adjusting timing based on real-time data, and potentially mechanical barriers or gates for access control [4]. By leveraging advanced technology like AI and machine learning, the system intelligently adapts to changing traffic conditions, optimizing flow in real-time. Benefits encompass reduced congestion, improved safety, and enhanced transportation network efficiency. However, challenges such as cost, technical complexity, and privacy concerns related to data processing and collection need addressing [5]. Analyzing traffic management systems involves a structured approach that considers various factors such as technology, functionality, and objectives. This comprehensive analysis covers a broad spectrum of topics, ranging from conventional methods to cutting-edge approaches. It delves into the strengths and limitations of each system while also exploring emerging trends in the field. [6]. The main body of the paper likely details the methodology for developing and implementing a fuzzy inference system for traffic light control, including defining linguistic variables and establishing fuzzy rules. Simulation results and performance metrics likely demonstrate the system's superiority over traditional methods [7].

The article on smart cities and smart roads likely emphasizes integrating advanced technology into road infrastructure to address challenges like congestion and pollution while enhancing safety and mobility. It underscores benefits like intelligent transportation systems and human-centric features, discussing challenges, opportunities, and future directions [8]. Research on urban challenges discusses the importance of efficient traffic management systems for addressing congestion, enhancing safety, and promoting sustainable mobility. It contributes valuable insights for policymakers, planners, and transportation professionals [9, 10]. Technology integrations like YOLO for object detection and IoT-based traffic control systems showcase efficient monitoring, congestion detection, and real-time management, offering practical solutions to traffic issues [11]. The paper on the YOLOv3 algorithm highlights its high precision and near-real-time performance in detecting traffic participants across varied driving conditions, showcasing adaptability and effectiveness [12]. Table 1 compares the many sorts of work scholars do on various platforms and their applications. All traffic management systems accept video data as input and analyze it to extract information about the traffic situation.

The article proposes leveraging machine learning-driven technology, specifically the YOLOv3 object detection technique, to address the challenges associated with processing traffic stream videos for traffic management. YOLOv3 is known for its ability to detect objects in real-time images efficiently. The article suggests using YOLOv3 to determine the duration of green lights based on the number of vehicles detected, potentially optimizing traffic flow in real-time. This approach could potentially offer a quicker, more accurate, and more cost-effective solution compared to traditional methods of processing video streams for traffic management [25, 26]. Table 2: Studies that examine the conceptual aspects of multiple YOLO-based machine learning-driven models for object recognition and compare

Table 1 Summary of current studies and applications

Study	Algorithm and application	Simulation Approach
Madalin, [13]	Traffic Simulation of Intersection "Electro"	Simulation Experiment
W Genders, [14]	Traffic Micro-simulator SUMO	Empirical Study
T Thunig [15]	Optimizes Fixed-Time Traffic Signal	Analytical Simulation
Zibo Ma, [16]	Optimization of Traffic Signal Timing	Traffic Simulation SUMO
B Stefan. [17]	Reinforcement Learning for Traffic Control	Agent-Based Simulation
K Małecki, [18]	Algorithm to Traffic Lights Timer	Machine-Based Simulations
Bandaragoda, [19]	Commuter Behavior Profiling Framework	Traffic Simulator SUMO
J. García-Nieto, [20]	Traffic Light Scheduling	SUMO Simulation
Jia H, [21]	Particle Swarm Optimization Algorithm	Mathematical Simulation
W Genders, [22]	Adaptive Traffic Signal Control	Traffic Simulator SUMO
X. Liang, [23]	Self-Adaptive Control System	Empirical Study
Sangeetha, S. [24],	Support Vector machine(SVM) Classification Algorithm	Spatial and Temporal Remote Sensing's

Table 2 Summary of current studies and applications of various YOLO-based machine learning models

Study	Yolo Models	Application	Basic Characteristics
[27]	YOLO: V1	Classifying trash	The suggested model can produce a high accuracy rate of 69.70% with a quicker speed, a smaller model size, and is more resilient
[28]		Intelligent binocular system	Using Yolo-Lite for target detection
[29]	YOLO: V2	The detection of multiple vehicles	A revamped YOLO has been implemented on traffic flow and scenes alter depending on the vehicle features
[30]		Traffic signal light color detection	Detect traffic lights using images and videos
[31]	YOLO: V3	Pole counting and detection from video	To identify poles, YOLO (V3) is used
[32]		Applications of embedded system	For multi-scale object identification, a novel model mini-Yolov3 is presented
[33]	YOLO: V4	Computer-vision	"Aerial thermal imaging application for multiscale objects detection"
[34]		Coastal surveillance system	A light-weight ship detection utilizing Multichannel fusion
[35]		Flower detection from image	CSPDarkNet53 serves as the YOLOv4 backbone network for detection
[36]	YOLO: V5	Finding of heavy-duty vehicles	Detection of large trucks in locations to safeguard against weather conditions
[37]		Detection and categorization of Rice leaf disease	Using YOLO V5, they diagnose illnesses of rice leaves correctly and promptly
[38]		Identification of disease in bell pepper plants	The comprehensive detection of bell pepper bacterial spot illness
[39]	YOLO: V7	Detecting and tracking numerous objects	Examine the USV videos motion cue
[40]	YOLO: V8	Identify helmets in pictures	Single-shot data collection
[41]		Crop detection	Principal component analysis is used for trunk and branch detection and segmentation
[42]		Detection of fractures	When analyzing X-ray pictures of pediatric wrist injuries, YOLO v8 is used to identify fractures

different versions of the YOLO (You Only Look Once) framework. These studies consistently focus on better comprehending the influence of various design decisions on performance metrics involving velocity, precision, and efficiency.

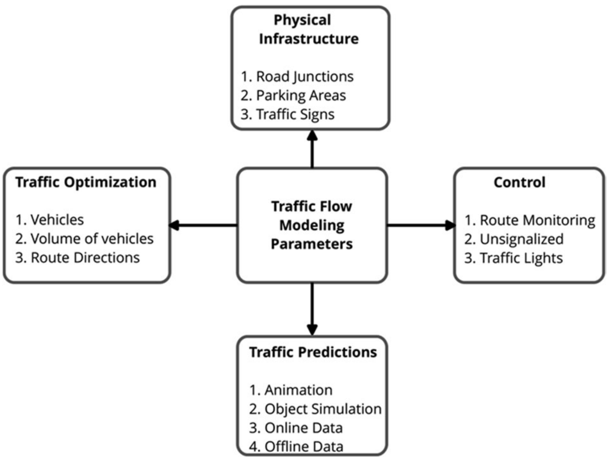
3 Traffic simulation and proposed framework

Using state-of-the-art vehicular traffic flow modeling, the interaction between drivers, vehicles, and infrastructure may characterize traffic dynamics. Hence, this self-adaptive intelligent traffic prediction model combines many data: driver behavior in automobile acceleration, street agglomeration, travel speeds, pedestrian movement, and road sign placement. It is desirable to obtain the optimum time to cross junctions by analyzing this data to alleviate traffic delays. Simulation models have been developed primarily using data collected from various traffic monitoring sensor networks. These models enable the prediction of traffic evolution and can identify intersections with the highest risk of obstruction. Setting the proposed model parameters' values to ensure the simulation matches the traffic statistics is worth noting. The culmination of this procedure is a calibrated model capable of predicting traffic flow, commonly referred to as model calibration. Figure 1 depicts a schematic depiction of a vehicular traffic flow model with all these factors.

The proposed self-adaptive smart traffic optimization system consists of several functional blocks depicted in Fig. 2. A sophisticated traffic management system that utilizes computer vision techniques like YOLOv3 for object detection and classification, along with algorithms to control traffic signals and analyze vehicle flow patterns. This system can be highly effective for monitoring traffic conditions, managing signal timings, and detecting abnormal behaviors like wrong-way driving. Here's a breakdown of the components and their functions:

- Smart Traffic Camera: Traffic cameras capture images of the road network, which are then processed by the system.
- Traffic Monitoring and Control System: This system receives input from the traffic cameras and processes it using the YOLOv3 model. It then analyzes the detected objects to monitor traffic conditions.

Fig. 1 State-of-the-art vehicular traffic flow modeling



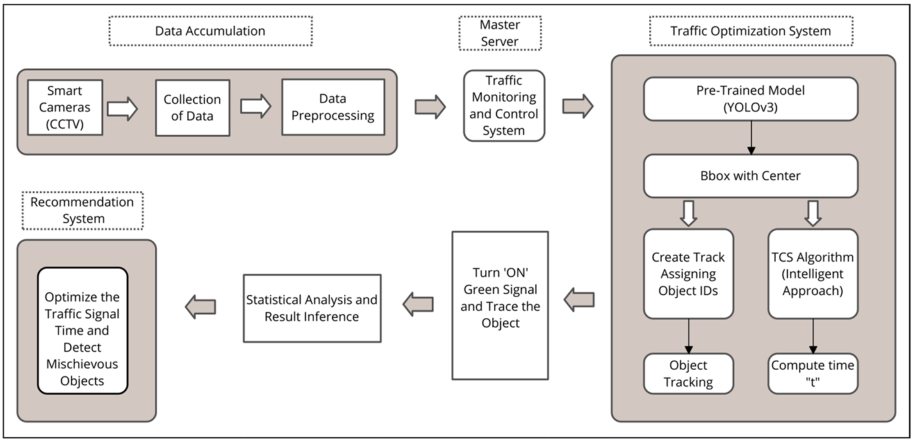


Fig. 2 The proposed framework of self-adaptive smart traffic optimization model

- Pre-Trained Model (YOLOv3): YOLOv3 is a deep learning-based model capable of detecting and classifying objects in images. In this context, it's used to identify vehicles and potentially other objects relevant to traffic management.
- Intelligent Approach: TCS algorithm uses the information gathered from the traffic monitoring system to determine the optimal timing for traffic signals. It computes the time 't' for each signal phase, deciding when to switch from red to green lights at intersections.
- Object Tracing: These are virtual representations of the paths taken by vehicles as they move through the road network. The system creates and analyzes these flow lines to understand the direction and volume of traffic flow.
- Statistical Analysis and Result Inference: By examining the flow lines, the system can identify instances where vehicles are traveling in the wrong direction. This could indicate a serious safety hazard, and appropriate actions can be taken, such as alerting authorities or adjusting signal timing to prevent further incidents.

4 Proposed methodology and algorithm

4.1 System architecture

The system necessitates the installation of four different cameras, each at the top of the road that intersects at the traffic road junction (Fig. 3). These cameras send their acquired pictures to an integrated controller. This controller counts the number of traffic objects, including TW and FW, by analyzing the real-time photographs. Subsequently, the timing of the green signal is modified to account for the number of objects detected.

Smart cameras have been installed atop traffic lights at many intersections, requiring connectivity to a controller. As depicted in Fig. 4, cameras typically cover only one side of the road. The controller might carry out the following operations:

1. Specify that the camera snaps a picture of the traffic scenario.
2. The number of objects in the camera's input images should be counted.

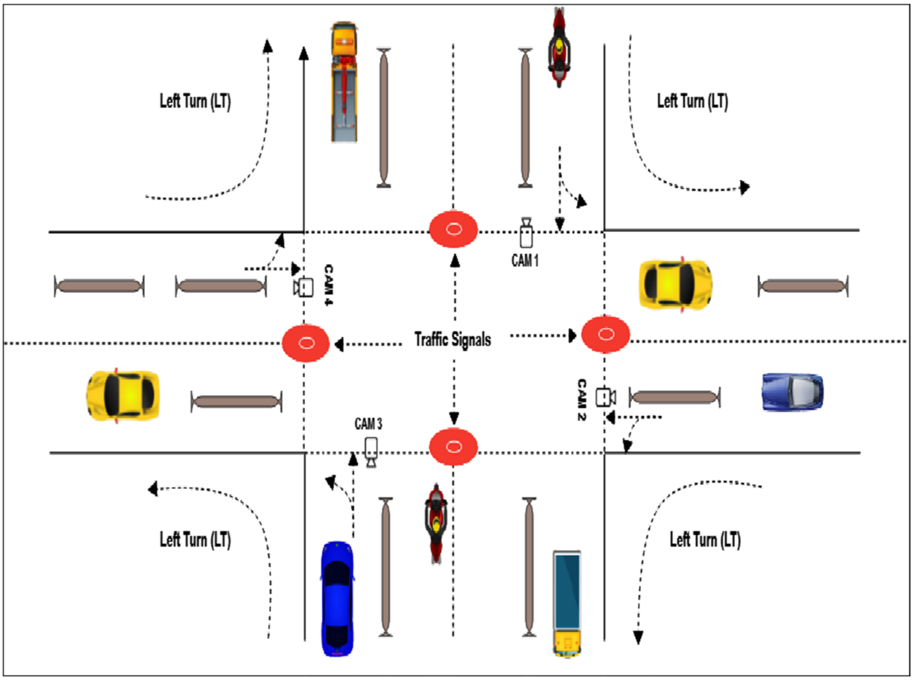


Fig. 3 The placement of self-adaptive smart traffic cameras: top view

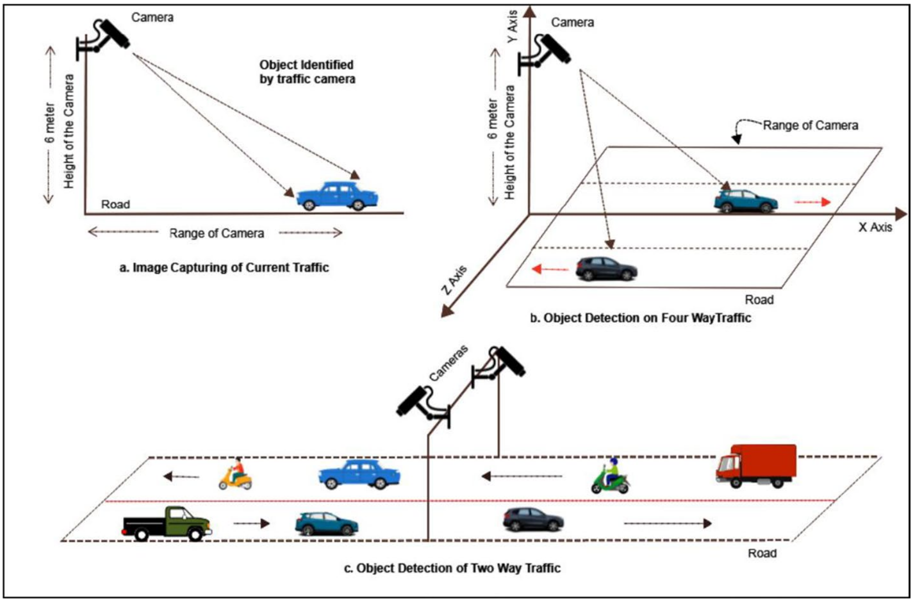


Fig. 4 The placement of self-adaptive smart traffic cameras: side view

3. To determine how long the green light should be on at a particular position based on the number of vehicles.
4. Turn on the red light of the other signals and the green light of the corresponding signal.

4.2 Data sets

The network has to be trained beforehand using the potential input set to get more accurate predictions. Before being put through actual testing, the YOLOv3 model was trained using many traffic junctions and vehicle images. The official Incubation Center Smart City of State Govt. MP, a public research and development agency located in Madhya Pradesh, India, provided the dataset used in this study, referred to as "Traffic Data." Three different picture types are included in the mobility aids dataset: RGB images, perfect depth images, and depth map images. Five distinct classes make up this dataset: persons with walking frames, transport items, TW objects, FW objects, and traffic lanes at intersections. Every image has tags in it that correspond to the image information found in the annotation file. It contains category information for every recognizable item (TW and FW) in the picture. The five categories listed above comprise the category information, and four coordinate points (x_min, x_max, y_min, y_max) are often used to convey the location information. Furthermore, we computed the evaluation indicators in the test set. We applied our object identification model to the standard test set, which consists of more than 1000 images and their corresponding tag data.

4.3 Platform

The TensorFlow framework developed the suggested system based on multiple object detection and tracking using CNN. The laboratory setting served as the training and deployment site for our concept. The system configuration is TITAN XP (GPU), with 64 GB of RAM, an Intel Xeon (R) CPU (processor), and a 64-bit operating system. The model needs an image with an input size of 416×416, 100 training iterations on the traffic dataset, an initial CNN 0.001 learning rate, and a 0.9 momentum factor. Furthermore, this research makes use of the size and quantity of anchors—116×90, 156×198, 373×326), (30×61), (62×45), and (59×119)—obtained by k-means clustering in Yolo v3. The pre-defined IOU's threshold is 0.76.

4.4 Traffic control system algorithm

Driving objects are approaching from the driving side of the signal. Correct positioning of the cameras is crucial to prevent objects from overlapping in the picture, as overlapping can lead to erroneous detection results. Figures 5 and 6 depict the various types of photos captured by the traffic cameras and object detection and analyzed by the prediction model. The amount of vehicle overlapping is significant in the images on the left, making them unsuitable for processing. On the other hand, the two images on the right half of Fig. 6 demonstrate the vehicle's accurate distribution and slight overlapping. As a result, the approach will benefit significantly from these images.

The study uses OpenCV to count the automobiles in the input picture. OpenCV is a C++and Python-based image processing library. OpenCV [7] includes modules for object detection using a variety of approaches, and OpenCV is used in this study.

Fig. 5 Object detection and analysis for traffic at cross-road (Training Data Set A)

The cdn [43] module is efficiently used to predict and count the number of objects. The YOLO v3 model receives the input picture through OpenCV [44, 45]. The method splits the image into regions and predicts bounding boxes and probabilities for each area using a single neural network applied to the entire image. The anticipated possibility is that these bounding boxes are weighed. Each object recognized in the picture produces a list of bounding box coordinates. The number of vehicles in the returned list would represent the number of objects [46]. Most studies employ the same lane clearance time for TW and FW objects. However, FW objects take longer to start and leave the signal in a real-time scenario than TW vehicles. As a result, it is crucial to factor in the time it takes to depart from the signal intersection [47, 48]. Figure 7 illustrates the entire process of traffic optimization, encompassing all stages and the processing of data captured by traffic cameras. This comprehensive approach ensures effective traffic management and optimization based on real-time data analysis.

Differently, FW and TW objects contribute to the processing time required to clear the road traffic in this proposed work. The controller then assigns the on-time green light after approximating the true time period based on the number of vehicles (TW and FW) and their timings to clear the road lane, which is symbolized by the text ‘t’. The time ‘t’ is calculated using Eq. (1).

$$t = \lceil \frac{TW}{a} \rceil * n + \lceil \frac{FW}{b} \rceil * m \quad //\text{time calculation} \tag{1}$$

where,

Fig. 6 Object detection and analysis for traffic at cross-road (Training Data Set B)

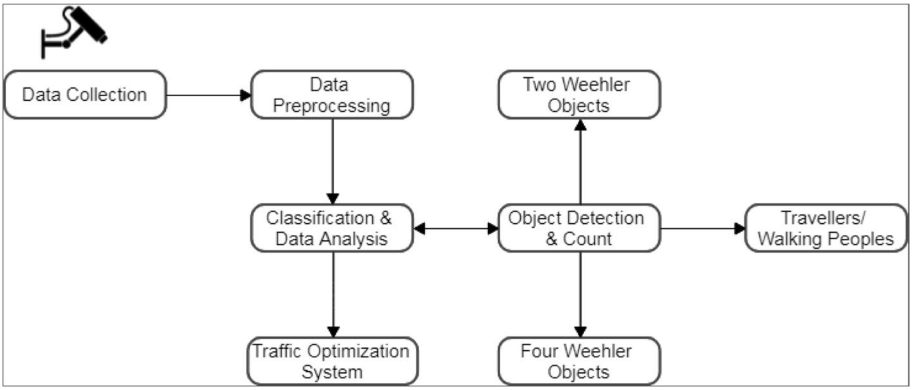


Fig. 7 Traffic Optimization Stages and Data Processing

TW stands for the total number of two-wheelers in the data.
FW is denoted by the number of four-wheelers/objects in the data.
n Denotes the number of seconds to clear a single two-wheeler.
m Denotes the time it takes to clear a single four-wheeler.
a The maximum number of TW objects available in the whole width of the road.
b Maximum number of FW objects available in the whole width of the road.

A vehicle's clearing time is determined by measuring how long it takes to pass the stop line from where it is currently standing. Consequently, the constant parameters n and m are determined by real-time observation of the traffic road junction. The measured values for n and m are 4.2 and 6.46 s, respectively. We adjusted the parameter value of n to 4 s because a driver's response time in most scenarios is more significant than 2 s.

More factors are needed to determine constants a and b, such as road width (Wroad), width of a TW (W2), width of a FW (W4), and the ground (G) spacing between any two objects. For different traffic crossroads, the width of the road may vary. We are focusing on the most prevalent scenario right now.

Wroad = 40 ± 5 feet, W2 = 3 feet, W4 = 7 feet, G = 2 feet, and finally, Eqs. (2) and (3) determine the values of a and b.

$$a = \lfloor \frac{W_{road}}{W_2+G} \rfloor \text{ //counting the most two – wheeled objects that can fit on the entire width of the road} \tag{2}$$

$$b = \lfloor \frac{W_{road}}{W_4+G} \rfloor \text{ //counting the most four – wheeled vehicles that can fit on the entire width of the road} \tag{3}$$

For, road width = 35 ft.

$$a = \lfloor \frac{35}{3+2} \rfloor = \lfloor 7.0 \rfloor = 7;\ b = \lfloor \frac{35}{7+2} \rfloor = \lfloor 3.88 \rfloor = 3$$

For, road width = 40 ft.

$$a = \lfloor \frac{40}{3+2} \rfloor = \lfloor 8.0 \rfloor = 8;\ b = \lfloor \frac{40}{7+2} \rfloor = \lfloor 4.44 \rfloor = 4$$

For, road width = 45 ft.

$$a = \lfloor \frac{45}{3+2} \rfloor = \lfloor 9.0 \rfloor = 9;\ b = \lfloor \frac{45}{7+2} \rfloor = \lfloor 5.0 \rfloor = 5$$

There is a high probability that many automobiles will be at the intersection and move slowly, starving the automobiles on the other side of the road. It is essential to set a maximum time restriction and determine the least practical number of objects within the time range the previously given algorithm predicted. Consequently, Eq. (4) provides the actual time period for computing the duration of the green light.

$$t_{\text{actual\_time}} = \min(\text{t}, \text{MAX\_GREEN\_TIME}) \text{//t – Actual time for max of green light time} \tag{4}$$

To assess the worst-case scenario result of Eq. (1), where all automobiles are four-wheelers (FW), we need to determine the MAX_GREEN _TIME. Given that we cannot remove more than 30 FW vehicles from any one side of the intersection and assuming no TW, we can set TW = 0 and FW = 30 in Eq. (1) to calculate the expected time:

For road width = 35ft

$$t = \lceil \frac{0}{7} \rceil * 4.2 + \lceil \frac{30}{3} \rceil * 6.46 = 64.6 \text{ sec}$$

For road width = 40ft

$$t = \lceil \frac{0}{8} \rceil * 4.2 + \lceil \frac{30}{4} \rceil * 6.46 = 48.45 \text{ sec}$$

For road width = 45ft

$$t = \lceil \frac{0}{9} \rceil * 4.2 + \lceil \frac{30}{5} \rceil * 6.46 = 38.76\,\text{sec}$$

With a minimum road width of 35 feet, we utilized the worst-case scenario, corresponding to a MAX_GREEN_TIME of 64.6 s, for future calculations. Figure 8 depicts Eq. (1) concerning the number of objects in three different kinds of scenarios: (1) when traffic only has TW (TW = 30 & FW = 0), (2) when traffic only has FW (TW = 0 & FW = 30), and (3) when traffic has both TW and FW in the same ratio (TW = 15 & FW = 15). The curve with a MAX_GREEN_TIME of 64.6 s approaches a flat irrespective of the proposed three conditions, as shown in Fig. 8.

To study the worst-case situation further, we only assessed FW vehicles and computed clearance times. It is calculated that it will take 775 s for all 120 FW objects in traffic to clear. The round time shown in Fig. 9 denotes the time a vehicle (TW and FW) is granted a second clearing chance after the MAX GREEN TIME threshold time has been reached.

For each of the four directions that cross at the intersection:

1. Take the input as an image before turning on the green light for two seconds.
2. Prepare the picture for vehicle detection by preprocessing it.
3. Count the (a) TW and (b) FW in the picture using object detection.
4. Using the numbers in the equation, calculate the total time ‘t’.
5. Continue to have the green light on the entire time ‘t’.
6. Repetition of step 1 for the following camera.

4.5 Object tracing algorithm

As described in this section, estimating vehicle trajectory involves calculating the difference in pixels within a range of XF-frames. Here, ’F’ represents the number of frames used to monitor a specific object. This difference in pixels is then analyzed to determine the flow direction of a tracked object. First, we establish the object’s starting

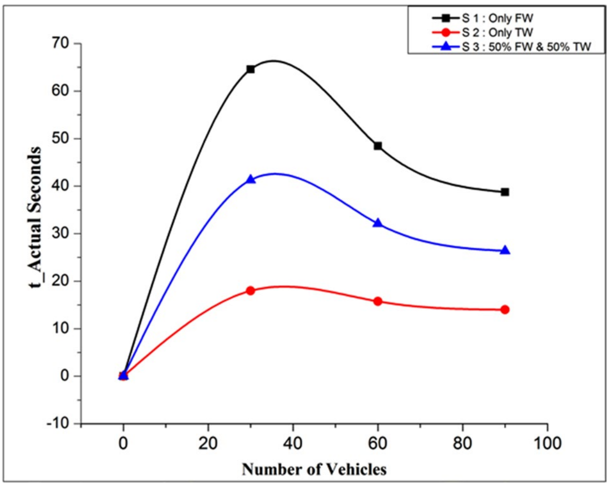


Fig. 8 Compute the time for the green light, which has a peculiar behavior

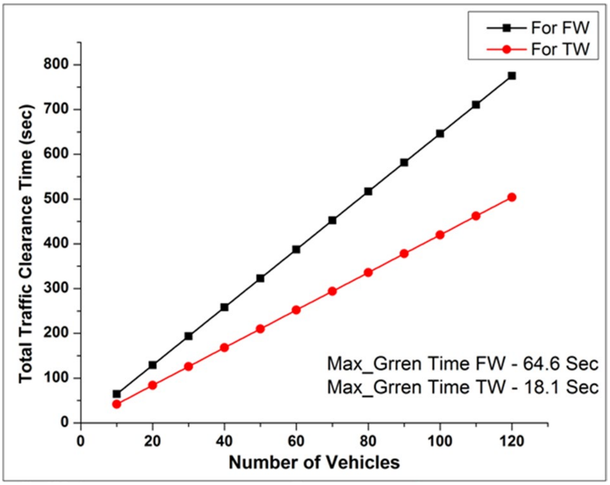


Fig. 9 In the worst-case scenario, total traffic clearance time is proportional to the number of vehicles

coordinates (X0, Y0) in the frame (F0) by computing its beginning location from the bounding box of a detected automobile. As the automobile drives, the starting coordinates' area changes from frame to frame. Therefore, for each frame, we determine the pixel difference between (X1, Y1) and (XN, YN), where X1, Y1 denotes an object's (Initial + 1 Frame) X0 + 1 location in a frame (F1) and XN and YN in a frame (FN). Let us use six frames as an example and utilize them for tracking. The difference in pixel position would be Do = X5—X1, where Do is the direction of the object. The second frame is the initial starting point, while the sixth is the final stopping point. The ID assignment phase is carried out on the first frame when the automobile is spotted.

Algorithm for object tracing

{

If (length of Initial Position X0) exceeds 0,

For (iterate _ number of detected initial objects),

For (iterate _ N, where N number of tracking frames for unique objects)

Take X_1, Y_1 positions from tracker // (from initial X_0, Y_0 to $X_{1,2,3\ldots N}$, $Y_{1,2,3\ldots N}$)

Take ID for distinct objects // Assigned unique ID for each objects)

$D_O = X_N \Delta X_1$ if (check D_O (Difference) and sign with X0, X1...X_N)

Assign direction

}

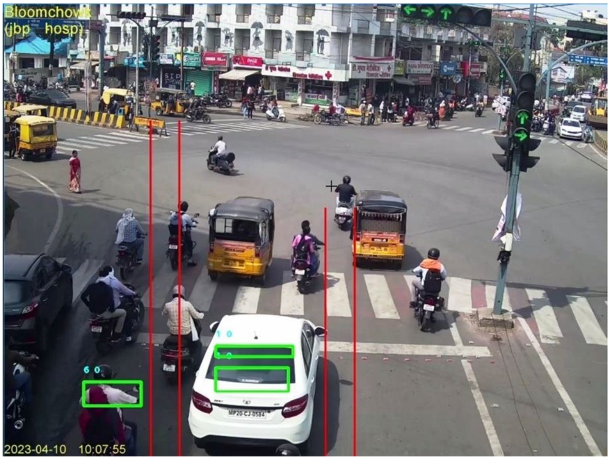
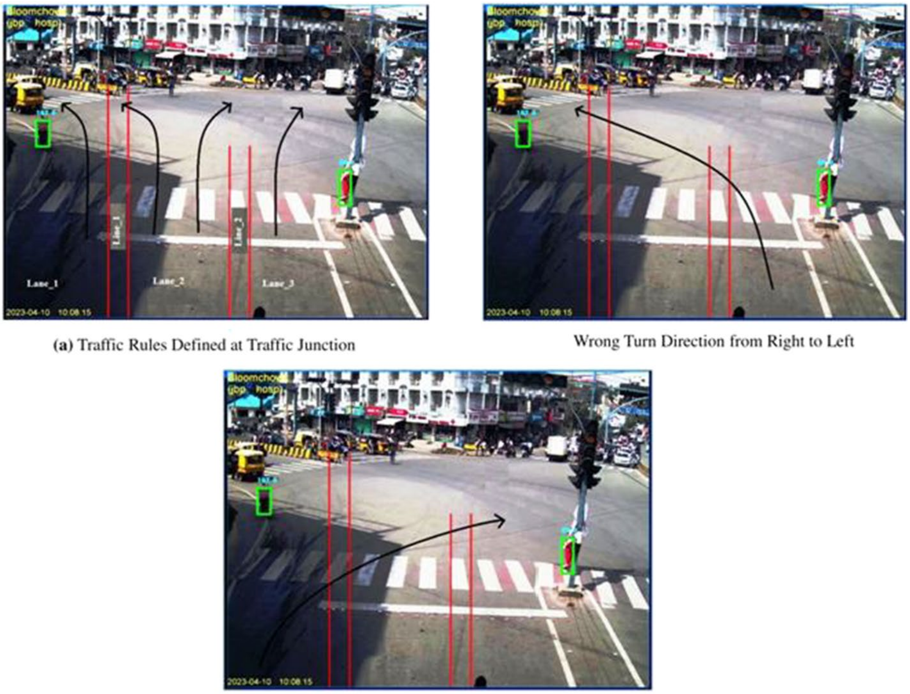


Fig. 10 Two-line approach framework for traffic flow at traffic junctions

4.6 Two-line approach framework

We implemented the two-line method using three fictitious lanes to verify the proper direction of objects. The concept is clear and practical, illustrated by drawing two lines, Line_1 and Line_2, along with three simulated lanes on the video frame, as depicted in Fig. 10.



(a) Traffic Rules Defined at Traffic Junction

Wrong Turn Direction from Right to Left

(c) Wrong Turn Direction from Left to Right

Fig. 11 Defined Framework for traffic flow and validation of directions at traffic junctions

Following the initial setup, we crop the video frame to focus on the desired area of interest. Traffic is directed to pass through the intersection of Lines 1 and 2, denoted by red lines. As shown in Fig. 11, we provide an innovative traffic rule based on a two-line technique to enhance the efficiency of traffic movement.

4.6.1 Assigning object IDs

Vehicle Unique IDs are stored in a list upon arrival at Line_1 or Line_2 and removed when passing through the traffic area.

4.6.2 Designated traffic lanes

- Line_1: Reserved for vehicles making left turns or proceeding straight from their starting point.
- Line_2: Exclusively designated for vehicles moving straight ahead from their starting point or executing right turns.

4.7 Wrong-Directions and reckless driving object detection

The object crosses both Line_1 and Line_2 simultaneously; it is identified as traveling in the wrong direction, triggering a warning signal. Additionally, objects with an angle exceeding 60 degrees concerning a particular Line_1 or Line_2 are flagged as engaging in reckless driving, as shown in Fig. 12.

Figure 13 illustrates how the location of the moving object changes throughout $N = 4$ frames. The green box represents the object's displacement across each frame. In this instance, the word objects ID indicates that the object travels from right to left or left to right in the video.

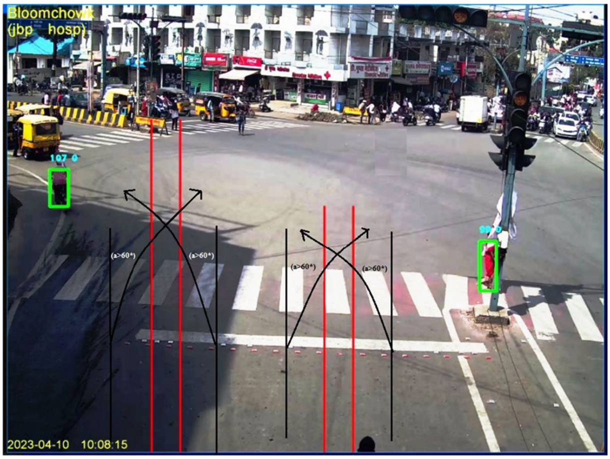


Fig. 12 Reckless driving angle (a) deviation > 60* for each line

Fig. 13 Displacement of an object within $N=4$ frames

5 Results and discussion

To assess the effectiveness of our effort, we carried out three different sorts of experiments. The first included 45 genuine real-time traffic photos from the two main junctions of Jabalpur (Shastri Bridge), Madhya Pradesh, which have been used to train the YOLOv3 object identification model. We have taken nine real-time traffic images preserved from the training data set. Table 2 displays the results of the test photographs. The second included evaluating the accuracy of object identification in various illumination scenarios, and the third involved wrong-way driving detection. We have made a real-world camera available and used it to simulate an automobile traveling in the opposite direction.

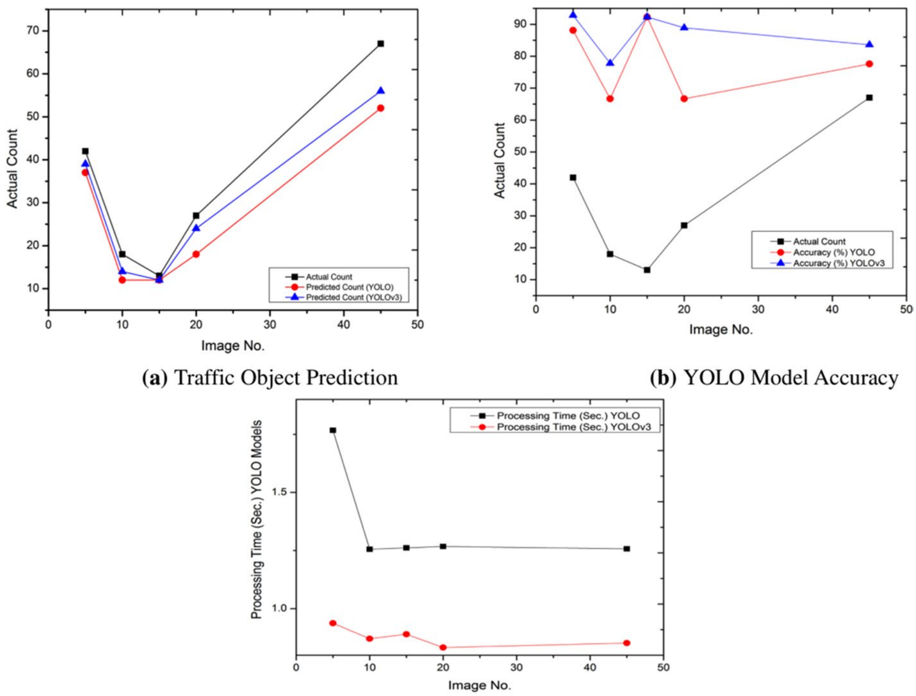
5.1 Objects detection and counting validation

First, we collected data from smart cameras mounted at traffic intersections to test the performance of object recognition. The numbers of objects in the image data set are manually computed. We then routinely choose a picture dataset with various time intervals. We selected a range of time windows, including dawn, dusk, and nighttime hours. We determined the mean average precision (mAP) with an IOU threshold of more than 0.76 to assess detection accuracy.

$$\mu = \frac{No.of\ Predicted\ Objects}{Actual\ No.of\ Objects\ in\ Image\ DataSet} \geq 0.76$$

Table 3 Results of object detection and counting obtained from training sample data

Image No	No. of Actual Objects	Predicted Count	Two Wheeler	Four Wheeler	mAP > 0.75 (%)	Processing Time
5	42	39	17	22	92.85	0.937559
10	18	14	6	6	77.77	0.87109
15	13	12	8	4	92.30	0.890319
20	27	24	7	17	88.88	0.833055
25	37	33	21	12	89.18	0.849508
30	22	20	11	9	90.90	0.98238
35	47	42	36	16	89.36	0.842146
40	84	80	41	39	95.23	0.897944
45	67	52	21	28	77.61	0.852594



(a) Traffic Object Prediction

(b) YOLO Model Accuracy

(c) Processing Time of YOLO Models

Fig. 14 Comparative analysis of the YOLO model with other alternative object detection models

To establish the locations of the objects in each picture frame, we manually counted the number of objects inside the specified intervals and marked them. Only when the IOU rate was over 0.76 were the objects regarded as having been detected. Table 3 shows that the detection system recognized objects at various time slots with an average mAP of 88.51%. We tested our program, focusing on accuracy and speed, with an eye on its intended use in real-time traffic events. Despite the program being created to identify just a single class, we acquired superior performance compared to the creators of the classic YOLOv3 model in terms of the mAP score.

As shown in Fig. 14, YOLOv3 outperforms the original YOLO model regarding processing time. Changes in network architecture, such as convolutional layers and feature pyramid networks (FPN), which enable speedier inference with respectable accuracy, enable this development. Compared to YOLOv3, the original YOLO model's processing time is noticeably slower. Despite being the first to use real-time object recognition, YOLO's processing speed may not be as fast as that of more recent versions because of architectural constraints. Regarding object identification tasks, YOLOv3 outperforms the original YOLO model's accuracy. This improvement is ascribed to developments in training methodologies, multi-scale prediction, and feature pyramid networks, which lead to more accurate object localization and classification. Although YOLO provides acceptable accuracy, its single-scale prediction and network design restrictions may limit its ability to recognize tiny objects and cases with heavy occlusion. YOLOv3 has better object recognition performance, particularly when objects have different sizes, orientations, and levels of occlusion. The original YOLO model remains helpful for object detection across different environments and is praised for initiating real-time object identification. However, its performance may be restricted, especially in challenging scenarios involving tiny objects or severe occlusion. In comparison, YOLOv3 displays higher performance in processing speed, accuracy, and object-detecting capabilities. Its network design and training procedure improvements identify it as a favored alternative for real-time object recognition applications needing high accuracy and robustness.

Table 4 depicts the evaluation results of the proposed model, encompassing a confusion matrix and significant performance metrics like accuracy, recall, precision, and F1 score. The confusion matrix offers a granular breakdown of the model predictions compared to the actual ground truth across multiple image data sets.

5.2 Traffic rules violator's objects

The system's output for the detection portion was the total number of objects that passed through each lane of the traffic junction (the suggested framework) in a video frame. Roadside smart cameras record videos with a resolution of 1280 × 720 pixels. We saw moving

Table 4 The confusion matrix is used to evaluate the performance of training datasets

Image No	Actual Objects	Predicted Count	TP	FP	FN	TN	Accuracy	F1 Score	Recall	Precision
5	42	39	38	1	2	1	0.929	0.962	0.950	0.974
10	18	14	13	1	1	4	0.895	0.929	0.929	0.929
15	13	12	11	1	1	1	0.857	0.917	0.917	0.917
20	27	24	23	1	1	3	0.929	0.958	0.958	0.958
25	37	33	28	5	2	2	0.811	0.889	0.933	0.848
30	22	20	18	2	1	2	0.870	0.923	0.947	0.900
35	47	42	39	3	2	3	0.894	0.940	0.951	0.929
40	84	80	77	3	7	5	0.891	0.939	0.917	0.963
45	67	52	48	4	3	6	0.885	0.932	0.941	0.923

objects that were traveling in the incorrect direction at the same time, simultaneously crossing the right and left lines. Our technology also has object-detection capabilities. Additionally, our system recognizes an object if it deviates more than 60 degrees from the line's centroid position, indicating reckless driving. The images of the objects inside the video frame shown in Fig. 15 have been captured. Objects outlined in orange have entered the designated area in the incorrect direction, while those highlighted in dark green are behaving recklessly while traversing the road. To safeguard their identities, the vehicle license plates have been obscured.

The segmentation results for object driving direction and the accuracy (confusion matrix) of reckless driving identification that we acquired from our experiment are shown in Table 5. Our system correctly identified objects moving in the incorrect direction. In particular, every occurrence of a car traveling in the incorrect direction in each of the four video modules was correctly recognized. The accuracy rate of the suggested framework is outstanding, averaging around 90.45%.

6 Conclusions and future work

Self-adaptive road traffic management encompasses a range of strategies to reduce traffic congestion's occurrence and adverse effects, including minimizing delays, reducing object waiting times, and recognizing wrong turn-taking objects. Additionally, it involves improving the efficient flow of traffic on road networks and dynamically allocating traffic flows to optimize efficiency. Through these measures, road traffic management seeks to enhance overall traffic effectiveness and mitigate congestion-related challenges for motorists and the community. The proposed algorithm aims to optimize traffic flow by reducing vehicle waiting times at traffic junctions and mitigating congestion by managing the flow of vehicles on roads. The system can use the YOLO technique, a machine learning-driven object detection algorithm, to identify and track vehicle objects in real time. An AI-based self-adaptive traffic optimization system can dynamically adjust traffic signal timings based on the current traffic conditions. The approach allows the system to respond in real-time to traffic volume and pattern changes, improving overall traffic flow efficiency and reducing congestion. To

Fig. 15 Wrong-Turn taken and reckless driving objects are detected

Table 5 The confusion matrix and the results of wrong-turn-taken objects and reckless driving objects from the training datasets

S. No	Cases	Total number of object Count	Reckless Driving Object (angle deviation > 60* From the Base line)	Wrong-Turn taken Objects	TP	FP	FN	TN	Accuracy	F1 Score	Recall	Precision
1	Case-1	74	8	1	65	9	1	2	0.870	0.929	0.985	0.878
2	Case-2	167	6	1	160	7	2	3	0.948	0.973	0.988	0.958
3	Case-3	192	8	0	184	8	2	8	0.950	0.974	0.989	0.958
4	Case-4	75	7	1	67	8	2	5	0.878	0.931	0.971	0.893
5	Case-5	85	9	0	76	9	3	9	0.876	0.927	0.962	0.894

determine the optimal duration for green lights, we analyzed various temporal factors, including traffic volume, road width, and intersection processing times. We trained a neural network to enhance vehicle detection accuracy using extensive datasets of authentic traffic images. The method involved integrating two state-of-the-art YOLOv3 methods to effectively monitor and identify objects, particularly vehicles.

Additionally, we developed a novel technique to detect vehicles traveling in the wrong direction. Our experiments, conducted across two simulated scenarios, achieved an average object detection accuracy of 88.43% and a remarkable 90.45% accuracy in tracking vehicles engaging in wrong turns or reckless driving behaviors. Our technique proved highly effective in identifying vehicles moving in the incorrect direction, contributing to safer, more efficient traffic management and hardware advantages compared to traditional traffic-clearing strategies. By leveraging AI-based technology and innovative methodologies, our solution provides a cost-effective, efficient means of optimizing traffic flow and enhancing road safety. Upcoming research will explore using cameras positioned at higher elevations to provide panoramic and comprehensive road coverage. This approach aims to improve the accuracy of the data collected, enabling more precise detection and tracking of vehicle objects. By expanding the coverage area and leveraging advanced camera technologies, we anticipate significant improvements in the effectiveness and reliability of our traffic management system.

Acknowledgements The authors gratefully acknowledge the official Incubation Center Smart City of the State Government of Madhya Pradesh, India, a public research and development organization, for providing the dataset in this study.

Funding No funding was received for this work.

Data availability The datasets analyzed during the current study are available from https://github.com/hkphd20/Traffic-Data-Set.

Declarations

Conflict of interest The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Humayun M, Afsar S et al (2022) Smart traffic management system for metropolitan cities of kingdom using cutting edge technologies. J Adv Transp 2022:1–13. https://doi.org/10.1155/2022/4687319
2. Khan H, Kushwah K, Maurya MR, Singh S et al (2022) Machine learning driven intelligent and self adaptive system for traffic management in smart cities. Computing 104(5):1203–1217. https://doi.org/10.1007/s00607-021-01038-1
3. Redmon J and Farhadi A (2018) YOLOv3: An incremental improvement. Comput Sci arXiv:1804.02767. http://arxiv.org/abs/1804.02767
4. Rath M (2018) Smart traffic management system for traffic control using automated mechanical and electronic devices. IOP Conf Ser: Mater Sci Eng 377:012201. https://doi.org/10.1088/1757-899x/377/1/012201
5. De Souza AM, Brennand CA, Yokoyama RS et al (2017) Traffic management systems: a classification, review, challenges, and future perspectives. Int J Distrib Sens Netw 13(4):155014771668361. https://doi.org/10.1177/1550147716683612
6. Komsiyah S, Desvania E (2021) Traffic lights analysis and simulation using fuzzy inference system of mamdani on three-signaled intersections. Procedia Comput Sci 179:268–280. https://doi.org/10.1016/j.procs.2021.01.006
7. Toh CK, Sanguesa JA, Cano JC, Martinez FJ (2020) Advances in smart roads for future smart cities. Proc Royal Soc A: Math Phys Eng Sci 476(2233):20190439. https://doi.org/10.1098/rspa.2019.0439

8. Allstrom A, Barcelo J et al (2016) Traffic management for smart cities. Designing, Developing, and Facilitating Smart Cities 211–240. https://doi.org/10.1007/978-3-319-44924-1_11
9. Asha CS, Narasimhadhan AV (2018) Vehicle counting for traffic management system using yolo and correlation filter. In: 2018 IEEE International conference on electronics, computing and communication technologies (CONECCT). https://doi.org/10.1109/conecct.2018.8482380
10. Alpatov BA, Babayan PV, Ershov MD (2018) Vehicle detection and counting system for real-time traffic surveillance. In: 2018 7th Mediterranean conference on embedded computing (MECO). https://doi.org/10.1109/meco.2018.8406017
11. Basil E, Sawant S (2017) IoT based traffic light control system using Raspberry Pi. In; 2017 International conference on energy, communication, data analytics and soft computing (ICECDS).https://doi.org/10.1109/icecds.2017.8389604
12. Corovic A, Ilic V, Duric S et al (2018) The real-time detection of traffic participants using YOLO algorithm. In: 2018 26th Telecommunications forum (TELFOR). https://doi.org/10.1109/telfor.2018.8611986
13. Pop MD (2018) Traffic lights management using optimization tool. Procedia Soc Behav Sci 238:323–330. https://doi.org/10.1016/j.sbspro.2018.04.008
14. Genders W, Razavi S (2018) Evaluating reinforcement learning state representations for adaptive traffic signal control. Procedia Comput Sci 130:26–33. https://doi.org/10.1016/j.procs.2018.04.008
15. Thunig T, Scheffler R, Strehler M, Nagel K (2019) Optimization and simulation of fixed-time traffic signal control in real-world applications. Procedia Comput Sci 151:826–833. https://doi.org/10.1016/j.procs.2019.04.113
16. Ma Z, Cui T, Deng W, Jiang F, Zhang L (2021) Adaptive optimization of traffic signal timing via deep reinforcement learning. J Adv Transp 2021:1–14. https://doi.org/10.1155/2021/6616702
17. Bosse S (2020) Self-adaptive traffic and logistics flow control using learning agents and ubiquitous sensors. Procedia Manuf 52:67–72. https://doi.org/10.1016/j.promfg.2020.11.013
18. Malecki K, Iwan S (2019) Modeling traffic flow on two-lane roads with traffic lights and countdown timer. Trans Res Procedia 39:300–308. https://doi.org/10.1016/j.trpro.2019.06.032
19. Bandaragoda T, Adikari A et al (2020) Artificial intelligence based commuter behaviour profiling framework using Internet of things for real-time decision-making. Neural Comput Appl 32(20):16057–16071. https://doi.org/10.1007/s00521-020-04736-7
20. Garcia-Nieto J, Alba E, Carolina OA (2012) Swarm intelligence for traffic light scheduling: Application to real urban areas. Eng Appl Artif Intell 25(2):274–283. https://doi.org/10.1016/j.engappai.2011.04.011
21. Jia H, Lin Y, Luo Q, Li Y, Miao H (2019) Multi-objective optimization of urban road intersection signal timing based on particle swarm optimization algorithm. Adv Mech Eng 11(4):168781401984249. https://doi.org/10.1177/1687814019842498
22. Li D, Wu J, Xu M, Wang Z, Hu K (2020) Adaptive traffic signal control model on intersections based on deep reinforcement learning. J Adv Transp 2020:1–14. https://doi.org/10.1155/2020/6505893
23. Wang Y, Yang X, Liang H, Liu Y (2018) A review of the self-adaptive traffic signal control system based on future traffic environment. J Adv Transp 2018:1–12. https://doi.org/10.1155/2018/1096123
24. Sangeetha SKB, Kushwah VS, Sumangali K, Sangeetha R, Raja KT, Mathivanan SK (2023) Effect of urbanization through land coverage classification. Radio Sci 58(11). https://doi.org/10.1029/2023rs007816
25. Hilmani A, Maizate A, Hassouni L (2020) Automated real-time intelligent traffic control system for smart cities using wireless sensor networks. Wirel Commun Mob Comput 2020:1–28. https://doi.org/10.1155/2020/8841893
26. De Beer D, Matthee M (2020) Approaches to identify fake news: a systematic literature review. Integr Sci Digit Age 2020:13–22. https://doi.org/10.1007/978-3-030-49264-9_2
27. Wu Z, Li H, Wang X et al (2022) New benchmark for household garbage image recognition. Tsinghua Sci Technol 27(5):793–803. https://doi.org/10.26599/tst.2021.9010072
28. Zheng Y, Ge J (2021) Binocular intelligent following robot based on YOLO-LITE. MATEC Web Conf 336:03002. https://doi.org/10.1051/matecconf/202133603002
29 Husein AM, Noflianhar LK et al (2024) Computer vision-based intelligent traffic surveillance: multi-vehicle tracking and detection. Sinkron 9(1):384–391. https://doi.org/10.33395/sinkron.v9i1.13204
30. Zhang X, Qiu Z et al (2018) Application research of Yolo v2 combined with color identification. In: 2018 International conference on cyber-enabled distributed computing and knowledge discovery (CyberC). https://doi.org/10.1109/cyberc.2018.00036
31. Gubbi J, Varghese A, Balamuralidhar P (2017) A new deep learning architecture for detection of long linear infrastructure. In: 2017 Fifteenth IAPR international conference on machine vision applications (MVA). https://doi.org/10.23919/mva.2017.7986837
32. Chen S, Lin W (2019) Embedded system real-time vehicle detection based on improved Yolo network. In: 2019 IEEE 3rd advanced information management, communicates, electronic and automation control conference (IMCEC). https://doi.org/10.1109/imcec46724.2019.8984055

33. Li J, Gu J, Huang Z, Wen J (2019) Application research of improved YOLOv3 algorithm in PCB electronic component detection. Appl Sci 9(18):3750. https://doi.org/10.3390/app9183750
34. Yue T, Yang Y, Niu JM (2021) A light-weight ship detection and recognition method based on YOLOv4. In: 2021 4th International conference on advanced electronic materials, computers and software engineering (AEMCSE). https://doi.org/10.1109/aemcse51986.2021.00137
35. Cheng Z, Zhang F (2020) Flower end-to-end detection based on YOLOv4 using a mobile device. Wirel Commun Mob Comput 2020:1–9. https://doi.org/10.1155/2020/8870649
36. Kasper-Eulaers M, Hahn N et al (2021) Short communication: Detecting heavy goods vehicles in rest areas in winter conditions using YOLOv5. Algorithms 14(4):114. https://doi.org/10.3390/a14040114
37. Kumar VS, Jaganathan M, Viswanathan A et al (2023) Rice leaf disease detection based on bidirectional feature attention pyramid network with YOLOv5 model. Environ Res Commun 5(6):065014. https://doi.org/10.1088/2515-7620/acdece
38. Dai M, Sun W et al (2023) Pepper leaf disease recognition based on enhanced lightweight convolutional neural networks. Front Plant Sci14. https://doi.org/10.3389/fpls.2023.1230886
39. Liang Z, Xiao G et al (2023) Motion track: Rethinking the motion cue for multiple objects tracking in USV videos. Vis Comput 40(4):2761–2773. https://doi.org/10.1007/s00371-023-02983-y
40. Lin B (2024) Safety helmet detection based on improved YOLOv8. IEEE Access 12:28260–28272. https://doi.org/10.1109/access.2024.3368161
41. Shi J, Bai Y et al (2023) Multi-crop navigation line extraction based on improved YOLOv8 and Threshold-DBSCAN under complex agricultural environments. Agriculture 14(1):45. https://doi.org/10.3390/agriculture14010045
42. Ju RY, Cai W (2023) Fracture detection in pediatric wrist trauma X-ray images using YOLOv8 algorithm. Sci Reports 13(1). https://doi.org/10.1038/s41598-023-47460-7
43. Elsagheer Mohamed SA, AlShalfan KA (2021) Intelligent traffic management system based on the internet of vehicles (IoV). J Adv Transp 2021:1–23. https://doi.org/10.1155/2021/4037533
44. Martinez-Alpiste I, Golcarenarenji G et al (2021) A dynamic discarding technique to increase speed and preserve accuracy for YOLOv3. Neural Comput Appl 33(16):9961–9973. https://doi.org/10.1007/s00521-021-05764-7
45. Qadri SSSM, Gokce MA, Oner E (2020) State-of-art review of traffic signal control methods: challenges and opportunities. European Trans Res Rev 12(1). https://doi.org/10.1186/s12544-020-00439-1
46. Liu X, Wang H, Dong C (2021) An improved method of nonmotorized traffic tracking and classification to acquire traffic parameters at intersections. Int J Intell Transp Syst Res 19(2):312–323. https://doi.org/10.1007/s13177-020-00247-w
47. Wang Z, Cui J, Zha H, Kagesawa M et al (2014) Foreground object detection by motion-based grouping of object parts. Int J Intell Transp Syst Res 12(2):70–82. https://doi.org/10.1007/s13177-013-0074-8
48. Saleem M, Abbas S et al (2022) Smart cities: Fusion-based intelligent traffic congestion control system for vehicular networks using machine learning techniques. Egyptian Inform J 23(3):417–426. https://doi.org/10.1016/j.eij.2022.03.003

Authors and Affiliations

Hameed Khan[1] · Jitendra Singh Thakur[1]

✉ Hameed Khan
hameed.khan20@gmail.com

[1] Deptt of Computer Science & Engineering, Jabalpur Engineering College, Jabalpur, Madhya Pradesh, India