Research Type (Original)

AI Based Traffic Management System

Bibek Shrestha[1*], Bishal Singh[2], Gaurav Darlami[3], Nabin Upadhayaya[4]

[1,2,3] National College of Engineering, Tribhuvan University

Correspondent author: bibekshrestha881@gmail.com

Article information	**Abstract**

Keywords:

Traffic flow
Optimization
Open CV
Artificial Intelligence
Waiting time
parallel processing
Traffic Management System

Background: Increased traffic flow in growing cities causes congestion, higher fuel consumption, and frequent traffic jams. Traditional traffic management systems rely heavily on traffic police and fixed traffic light controls, which lack real-time adaptability. This study addresses the need for an efficient, intelligent traffic management solution to reduce congestion and optimize vehicle flow.

Objective: The main objective of this study is to develop an Artificial Intelligence (AI)-based traffic management system using YOLO algorithms to detect vehicle density and dynamically adjust traffic signals.

Methods: The system utilizes phone cameras to capture video of traffic lanes, and YOLO algorithms are employed for real-time object detection and density evaluation. Data collected from the cameras are processed using Python to determine the vehicle count in each lane. Based on this analysis, Arduino microcontrollers are programmed to prioritize traffic signals for lanes with higher vehicle density. The proof-of-concept implementation includes a prototype setup with two phone cameras, Arduino, and LEDs.

Results: The system successfully detected vehicle density and adjusted traffic signals dynamically, demonstrating improved optimization of vehicle flow compared to traditional fixed-time signal systems. Real-time parallel processing ensured continuous monitoring and responsiveness to changing traffic conditions.

Conclusions: The implementation of this AI-based traffic management system demonstrates significant potential to enhance traffic flow and reduce congestion. Future improvements could include scaling the system to larger networks and integrating additional sensors for better performance under varied environmental conditions.

1. Introduction

The rapid increase in vehicle numbers has significantly contributed to road congestion, which presents major challenges to both human safety and the environment. In cities like Kathmandu, where the vehicle population exceeds the available road capacity, traffic congestion has become a persistent issue. Despite resource constraints, the Metropolitan Traffic Police have effectively managed traffic flow, focusing on improving safety, reducing accidents, and implementing cost-efficient traffic management strategies. However, traditional traffic control systems, which rely on fixed signal timings based on historical data, struggle to adapt to dynamic traffic conditions, particularly during peak hours or after accidents.

Artificial Intelligence (AI) provides a promising solution to these limitations. AI-based systems, utilizing YOLO (You Only Look Once) algorithms for real-time vehicle detection, offer the ability to dynamically adjust traffic signal timings based on real-time vehicle density [1]. Unlike conventional systems that rely on fixed-time or fuzzy control, AI-based traffic management systems process real-time data to optimize waiting times and reduce traffic congestion [2]. By adapting to fluctuating traffic volumes, AI-driven systems offer enhanced efficiency in managing urban traffic. Traffic congestion not only leads to wasted time and resources but also contributes to increased emissions, significantly harming the environment. YOLO-based algorithms, with their ability to detect, classify, and count vehicles in real-time, optimize traffic flow by adjusting signal durations to prioritize lanes with higher vehicle density [3]. Moreover, AI-driven systems can prioritize emergency vehicles like ambulances and fire trucks, ensuring their swift passage through intersections and improving response times during critical situations [4] .

Traditional traffic control systems, such as fixed-time signals, loop-based systems, and deep reinforcement learning (DRL) models, are limited in comparison to AI vision-based solutions. DRL-based systems enhanced by YOLO integration allow for real-time optimization that outperforms conventional methods [5]. This research explores the potential of AI to optimize traffic light control in congested urban environments, providing an adaptable, real-time solution to modern traffic challenges [6].

2. Literature Review

Intelligent traffic management systems are crucial for achieving smart city objectives, which aim to improve the efficiency of urban traffic networks. Numerous approaches have been proposed to optimize traffic flow, leveraging a wide array of technologies and methodologies. Haider [7] introduced a machine learning-based system that dynamically adjusts traffic signal timing based on real-time traffic data, reducing congestion and enhancing overall traffic efficiency. This system utilizes historical traffic patterns to optimize signal control, ensuring smoother traffic flow.

Liu et al. [8] explored the application of multi-agent Q-learning for intelligent traffic light control. Their approach, which relies on distributed agents, allows each agent to learn optimal signal timings based on real-time traffic conditions. This system can adapt to fluctuations in traffic volume, providing a notable improvement over traditional fixed-time traffic light system. Wang et al. [2] expanded on this concept by integrating a spatio-temporal multi-agent reinforcement learning (RL) approach to traffic light management. This method employs a graph-based model to represent the relationships between multiple intersections, enabling more effective coordination, particularly during peak traffic hours. The model's adaptability to varying traffic conditions makes it a promising solution for optimizing traffic flow in large urban areas.

Further advancements in RL-based traffic control were made by Zhang [9], who combined RL with partial vehicle detection. This approach focuses on real-time decision-making, adjusting signal timings based on vehicle counts and traffic flow, allowing for more dynamic signal control. Kumar [10] further enhanced this model by introducing fuzzy inference systems into RL, enabling adaptive control in situations where precise traffic modeling is challenging. Deep reinforcement learning (DRL) has gained significant attention in recent years for traffic control. Pan [11] demonstrated that DRL-based systems could dynamically adjust signal timings in response to real-time data, improving overall traffic efficiency and reducing waiting times at intersections. These systems adjust to fluctuating traffic patterns, offering an advanced alternative to traditional traffic management methods. [12] proposed a smart traffic signal control system designed for smart cities, where real-time data processing facilitates adaptive traffic management. This system, which uses sensor-based data to guide decision-making, enhances traffic flow and reduces congestion in urban environments. Similarly, zhao [13] introduced a two-stage fuzzy control scheme to optimize signal timing at intersections, effectively balancing traffic loads and minimizing delays across multiple lanes.

The role of AI and computer vision in modern traffic management has become increasingly important. YOLO (You Only Look Once), an object detection algorithm, has been widely studied for vehicle detection and counting. According to Nvidia's documentation [14], YOLO provides real-time object detection with high accuracy, making it well-suited for traffic management systems. [15]further validated YOLO's ability to handle large datasets efficiently, showcasing its potential as a robust tool for real-time vehicle counting and classification. Additionally, integrating AI systems with emergency vehicle prioritization has been explored by Sharma and Bansal [16], who proposed intelligent traffic light systems that utilize deep learning for traffic environment analysis. These systems not only optimize traffic flow but also enhance emergency response times by prioritizing the passage of vehicles such as ambulances and fire trucks.

Lee et al.[17] proposed a smart traffic light controller using embedded systems, demonstrating the cost-effectiveness and feasibility of implementing intelligent traffic systems with low-cost hardware solutions. Their work highlights that intelligent traffic management can be achieved with embedded systems while maintaining system reliability and scalability. The use of intelligent agents in traffic control was further explored by Roozemond [18], who discussed the proactive role of agents in managing urban intersections in real-time. His work emphasized the importance of intelligent agents in managing traffic flow, a strategy that could enhance the flexibility and scalability of AI-based traffic systems. Kosonen [19] examined multi-agent fuzzy control, using real-time simulations to adjust signal timings based on fuzzy logic models. This approach improves decision-making in complex urban traffic networks, where precise modeling is difficult. Sutton and Barto [20] introduced reinforcement learning techniques, laying the groundwork for many advanced RL-based systems, including those used in traffic management.

Further developments in AI-based object detection for traffic management were made by Open Data Science [21], who provided an overview of YOLO's capabilities in object detection and classification. This work strengthened YOLO's role as a leading technology for real-time vehicle detection in intelligent transportation systems. Liu and Qin [22] emphasized the potential of intelligent traffic light systems to provide scalable solutions that not only manage traffic but also improve emergency vehicle prioritization through real-time decision-making and AI integration. Zhang et al. [23] supported this by demonstrating smart traffic systems using real-time monitoring and intelligent control. Chavan [24] proposed a Cognitive Road Traffic Management System (CTMS) based on the Internet of Things (IoT), exploring the integration of IoT technologies with AI for smarter traffic management. This approach demonstrated the potential of combining AI with IoT to enhance urban traffic efficiency.

The integration of deep learning and reinforcement learning for adaptive traffic signal systems was further explored by De Oliveira et al. [25], who discussed the importance of combining these methods with real-time vehicle data to optimize signal control in congested areas. Several authors [26], [27]contributed to enhancing the understanding of traffic signal optimization through AI, vehicle detection algorithms, and reinforcement learning techniques. These studies highlight the ongoing evolution of traffic management technologies, particularly in urban areas where congestion remains a major challenge.

Kosonen [28] proposed an intelligent traffic light control system that utilizes deep learning to optimize signal timing based on real-time traffic conditions. Their approach leverages advanced neural network architectures to analyze and predict traffic patterns, enabling the system to dynamically adjust signal phases. This methodology goes beyond traditional rule-based or fixed-time systems, which are often ineffective in handling fluctuating traffic volumes. By incorporating deep learning, the system is capable of processing vast amounts of real-time data, such as vehicle count, speed, and congestion levels, to make timely and efficient signal adjustments. Bansal and Sharma's work emphasizes the potential of AI to enhance the accuracy and responsiveness of traffic management systems, improving overall traffic flow and reducing delays.

Sharma [29] provided an extensive survey on the application of intelligence methods in urban traffic signal control, highlighting various approaches and technologies used in optimizing traffic signal systems. The paper categorizes these methods into several key areas, including heuristic approaches, genetic algorithms, and expert systems, alongside emerging machine learning techniques. Liu's review discusses the challenges faced by traditional signal control systems, such as fixed timing and limited adaptability, and emphasizes the advantages of intelligent systems in improving traffic efficiency. The study also explores how technologies like reinforcement learning, fuzzy logic, and multi-agent systems can be integrated into traffic control to provide more adaptive and scalable solutions. Liu's survey serves as an important reference for understanding the evolution of intelligent traffic systems and their potential for transforming urban mobility.

3. Methodology

The rapid growth in vehicle numbers has led to increased traffic congestion, particularly in urban areas, which significantly delays travel times and contributes to environmental pollution. In this study, two traffic lanes, each experiencing varying levels of vehicle density, are monitored to assess real-time traffic flow and adjust traffic light control accordingly. This approach is based on intelligent traffic management systems outlined in previous research, which employ real-time data processing and adaptive control strategies.

3.1. Data Collection with IP Webcam Software

To collect real-time traffic data, we employ IP webcam software that connects to the network and transmits video footage over Wi-Fi, enabling remote monitoring of traffic conditions in both lanes. The camera captures video footage, compresses the data using the MJPEG format, and converts it into a digital signal. The resolution and frame rate of the camera are adjusted to ensure optimal video quality for accurate vehicle detection and counting, as demonstrated by several studies. This technique ensures high-quality video capture, critical for real-time object detection systems like YOLO.

3.2. Vehicle Detection Using YOLO Algorithm

The captured video data is processed using the YOLO (You Only Look Once) algorithm, a widely used real-time object detection method. YOLO is known for its ability to detect and classify objects (in this case, vehicles) within a single frame, providing high efficiency and real-time vehicle counting. YOLO's framework divides an image into a grid and predicts bounding boxes for detected objects, thus enabling dynamic traffic management. The efficiency of YOLO in processing frames quickly makes it ideal for real-time traffic light control applications, as demonstrated by previous research in the field. YOLO's application in traffic management systems has been studied in multiple contexts, including urban traffic flow optimization, vehicle classification, and real-time monitoring. This real-time processing allows for immediate adjustments to the traffic light cycle based on vehicle density, optimizing traffic flow and reducing congestion.

3.3. Integration with Arduino for Traffic Control

Once the vehicle count is determined, the data is transmitted to an Arduino microcontroller, which controls the traffic lights. The Arduino adjusts signal timing based on the vehicle density data received from the YOLO algorithm. This integration of YOLO with Arduino allows for dynamic adjustments to the traffic signal cycle, ensuring that the lane with higher vehicle density receives more green light time. The Arduino microcontroller is well-suited for real-time applications like this, as it can process and act on input data efficiently. This approach is based on the principles of adaptive traffic control systems, which continuously adjust signal timing in response to real-time data. Previous studies have shown the effectiveness of Arduino in managing traffic signals for smart cities and optimizing vehicle flow.

3.4. System Implementation and Evaluation

The combination of YOLO and Arduino creates an adaptable, real-time traffic management system. By processing data in parallel and adjusting the signals promptly, the system can improve traffic flow, reduce congestion, and minimize environmental impact. Similar intelligent traffic control solutions have been implemented in various smart city projects, demonstrating their effectiveness in reducing wait times and optimizing traffic flow. This methodology offers a scalable and efficient solution for urban traffic management, enabling real-time responses to fluctuating traffic conditions.

3.5. Algorithm and Flowchart

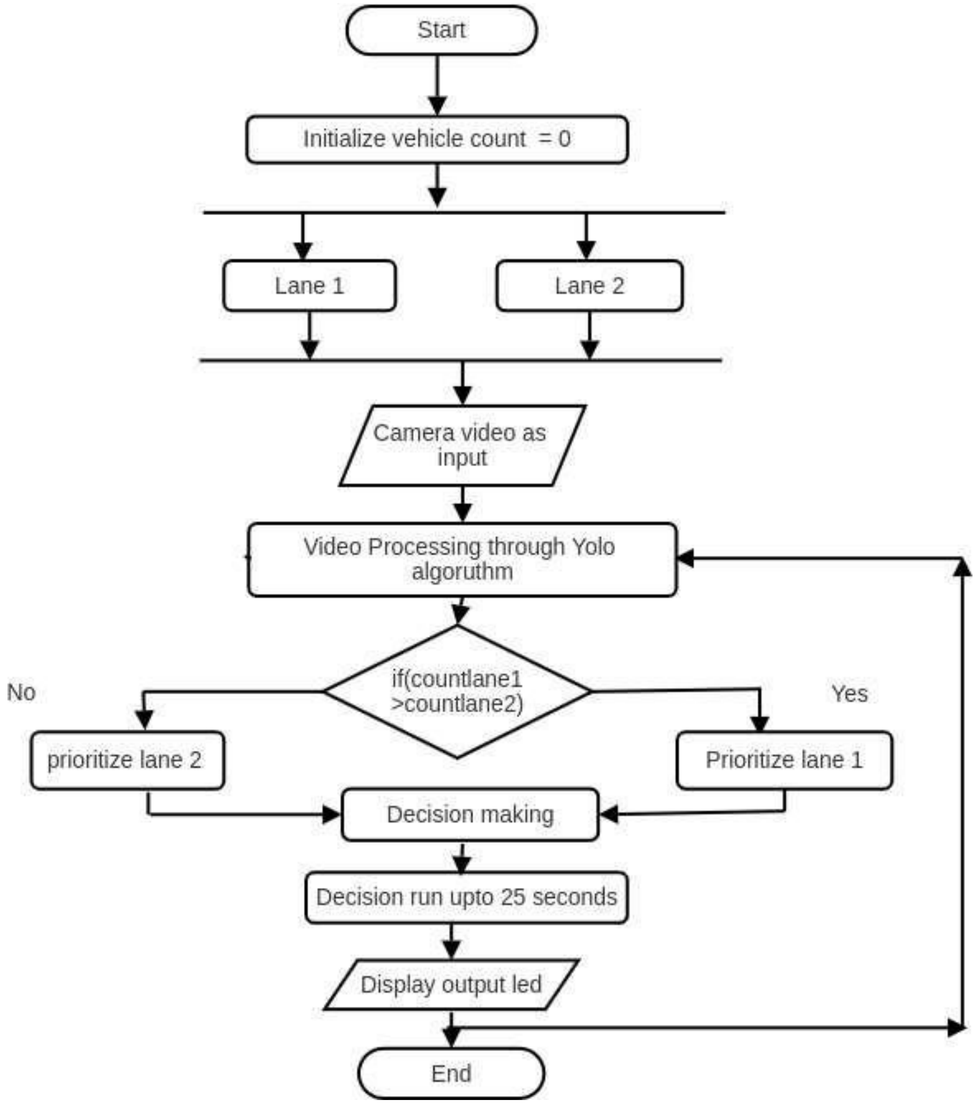

Fig. 1: Workflow Diagram

The proposed system employs the YOLO (You Only Look Once) algorithm to manage traffic flow at an intersection by dynamically adjusting traffic light signals based on real-time vehicle density. The process begins with the capture of live video data through a phone camera, which is then transmitted via an IP address over the same network to the system for processing.

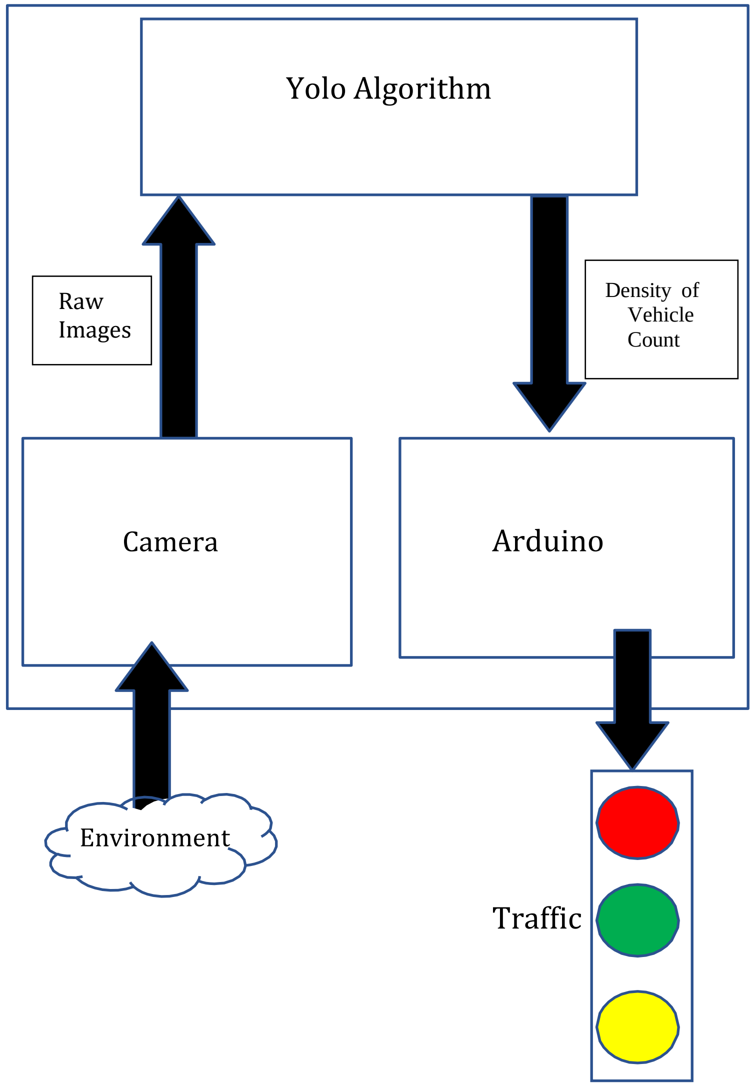

Fig. 2: System Block Diagram

3.6. Video Capture and Data Transmission:

The phone camera acts as a sensor that captures real-time video from two distinct lanes. The video data is transmitted over the network using the IP webcam application, which allows for efficient video feed transmission. The video frames are compressed in the MJPEG format and converted into a digital signal, which can be processed by the system.

3.7. YOLO Algorithm and Image Processing

Upon receiving the video feed, the system processes the data using the YOLO deep learning algorithm. YOLO is a real-time object detection system capable of quickly analyzing each video frame. The captured video frames are resized to 448x448 pixels, which is the required input size for YOLO. YOLO then applies a single convolution operation to detect and classify objects within each frame. The algorithm identifies various objects, but in this context, it focuses on detecting vehicles, including cars, buses, trucks, and motorcycles, based on a pre-trained dataset.

3.8. Vehicle Detection, Classification, and Counting

Once YOLO detects an object in a frame, it classifies the object as a vehicle and counts the number of vehicles in each lane. The system uses a dataset model that includes various classes of vehicles. After classifying the objects, YOLO counts the number of vehicles present in each lane and records these counts as countlane1 and countlane2 for the respective lanes.

3.9. Decision-Making Process for Traffic Light Control

The system is designed to operate at an intersection where four different lanes converge. The system continuously evaluates the traffic conditions in each lane, which is influenced by two main parameters: queue length and average waiting time. The queue length represents the number of vehicles waiting at the traffic light, while the average waiting time reflects how long vehicles have been waiting at the signal.

The system operates with predefined thresholds for both queue length and waiting time. Based on the real-time data obtained from YOLO, the system determines whether one lane has a higher vehicle density than the other. If lane 1 (countlane1) has a higher vehicle density than lane 2 (countlane2), the system prioritizes lane 1 by keeping the green light active for a longer duration. If lane 2 has more vehicles, the system will prioritize lane 2 instead.

3.10. Traffic Light Cycle Calculation

The traffic light cycle is influenced by the vehicle density and calculated using the formula proposed by Webster (1958). The optimal cycle time for the traffic light is derived based on the current traffic conditions, with the system calculating a cycle time of 128 seconds. However, the system operates more efficiently with an optimized cycle of 68 seconds, which results in a better flow of traffic compared to the traditional approach.

3.11. Dynamic Adjustment and Real-Time Response

Once the system determines which lane to prioritize, it dynamically adjusts the duration for which the green light stays on for that lane. The decision-making process continues until a predefined period, typically 25 seconds, after which the system evaluates the new conditions and adjusts the signal accordingly. The traffic light will remain green for the prioritized lane until the queue length and vehicle density have decreased sufficiently.

The system also accounts for situations when the queue length falls below a specified threshold or when the average waiting time is within acceptable limits. In such cases, the system follows a fair switching pattern, ensuring that traffic flow remains balanced. If the system detects that the threshold values have been exceeded, priority-based switching occurs to optimize traffic movement at the intersection.

3.12. Optimization and Parallel Processing

The system operates efficiently by utilizing kernel multiprocessing for data processing. This allows for the concurrent execution of multiple tasks, such as detecting vehicles, counting them, and adjusting the traffic light in real-time. The ability to process video data quickly and handle decision-making tasks in parallel is key to maintaining smooth traffic flow and minimizing delays.

3.13. Handling Traffic Congestion

When traffic congestion occurs, vehicles will inevitably experience delays due to the red light. The system's dynamic and adaptive nature ensures that the light cycle is continually adjusted based on the current vehicle density, thereby minimizing waiting times and reducing congestion. Additionally, the system can help prioritize emergency vehicles such as ambulances or fire trucks by keeping the light green for them, ensuring they can pass through the intersection quickly and reach their destination.

3.14. YOLOv8 and COCO Dataset

YOLOv8 (You Only Look Once version 8) is an object detection model developed by Ultralytics. It is part of the YOLO family of models, which are known for their speed and efficiency in object detection tasks.

YOLOv8 is specifically designed to work with large-scale datasets like the COCO (Common Objects in Context) dataset, which consists of over 330,000 images, with 200,000 annotated images for tasks like object detection, segmentation, and captioning.

The annotation format in YOLOv8 is relatively simple. Each image in the dataset is accompanied by a .txt file containing annotations for the objects present in the image. Each annotation corresponds to a single bounding box in the format:

- Class-id: the identifier for the object class (e.g., car, bicycle, truck).
- Center-x and center-y: the coordinates for the center of the bounding box.
- Width and height: the dimensions of the bounding box.

The COCO dataset is organized into three subsets:

- Train: Contains 118,000 images used for training models.
- Validation (Val): Includes 5,000 images used for validation during training.
- Test: Comprises 20,000 images used for testing and benchmarking model performance.

YOLOv8 achieves a mean average precision (mAP) of 50.2% on the COCO dataset, using a resolution of 640 pixels. It also demonstrates impressive speed, processing at 234.7 frames per second (FPS) on the TensorRT framework, with 25.9 billion floating-point operations per second (FLOPs).

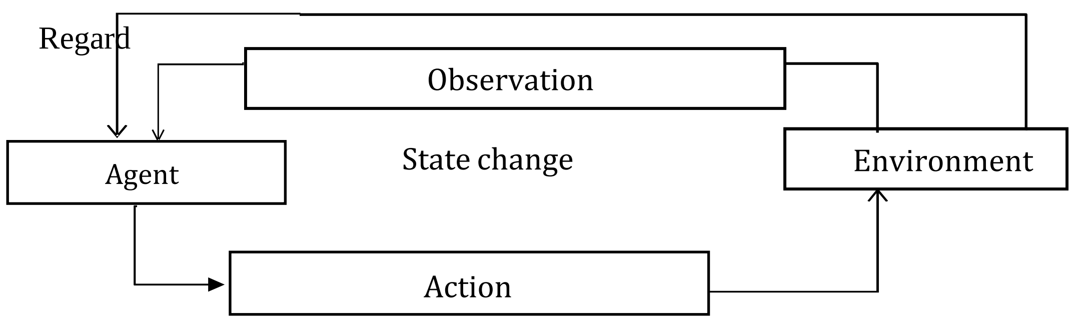

Fig. 3: State Flow Diagram

In a traffic management system, an Agent is responsible for deciding the actions that will manage traffic flow at intersections. The Agent assesses the current traffic scenario and determines what action needs to be taken (e.g., changing traffic lights). The Environment in which the agent operates refers to the real-world traffic conditions, including vehicles, signals, and junctions.

The Action refers to the steps taken by the agent. For instance, an action could be turning the traffic light green for a specific lane. There are two states for action:

- Action 0: No change to the traffic light.
- Action 1: Turn the traffic light green for the next sequence.

The Action Space is the set of all possible actions that the agent can take, and the State refers to the current condition the agent perceives in the environment, such as vehicle positions or speed.

The Reward is the feedback mechanism that informs the agent about the effectiveness of its actions. The goal of the agent in a traffic management system is to minimize average waiting times, delays, and queue lengths, while maximizing vehicle speed.

3.15. Traffic Signal Optimization (– mathematical optimization)

Traffic signals can be classified as:

- Pre-timed: Fixed cycle lengths for all time periods.
- Semi-actuated: Traffic detectors trigger changes in signal phases when vehicles approach.
- Fully actuated: Traffic detectors on all approaches dynamically adjust signal phases based on traffic flow.

The Cycle Length refers to the total time for a complete sequence of traffic lights at an intersection. In a fully actuated system, the cycle length is influenced by the traffic volume and the critical lane group (the lane that requires the greenest time).

A mathematical formula for calculating the minimum cycle length necessary for an intersection is:

$$C_{min}= \frac{L \cdot Xc}{Xc-\sum_{i=0}^{n} Yi} \qquad \text{--Equation (1)}$$

C_{min} = minimum necessary cycle length

L= Lost time (in seconds) per signal cycle due to factors like start-up delays

X_c = Critical – ratio for intersection $= \frac{v}{s}$

v= actual traffic volume in the lane

s=saturation flow rate

Y_i= Flow ratio for the i^{-th} critical lane group= $\frac{v}{c}$

c=maximum capacity of vehicle

n=number of Critical Lane Group

To minimize vehicle delay, Webster's formula for the optimal cycle length Copt is used:

$$C_{opt}= 1.5L + 5\times (1- \sum (\frac{Y_i}{X_i})) \qquad \text{-------------------------------------Equation (2)}$$

Where:

- L is the total lost time for the cycle
- Yiand Xi are the saturation and flow rates for each lane group

3.16. Comparison of Traffic Cycle Lengths

In the previous system, a cycle length of 140 seconds was calculated for a 35-second cycle at a junction. In the latest system, a 25-second cycle was considered, resulting in an optimal cycle length of 106.25 seconds. This reduction in cycle time could significantly improve the efficiency of the traffic signal system, reducing waiting times and congestion.

By applying this mathematical framework to traffic signal control, the agent-based model ensures smoother traffic flow, reduced delays, and optimized vehicle throughput at intersections.

$$C_{opt} = Optimal\ Cycle\ length\ for\ Miniming\ Delay$$

$$L = \sum_{i=1}^{n} (t_L)_{ci} \qquad \text{Equation (3)}$$

$$Lost\ Time\ equal\ (L) = Waiting\ time\ such\ as\ Signal\ Change, pedestrain, vehicle\ accerelation$$

Generally, four Intersections may contain 70-90sec cycle to overall complete the traffic light. But only the junction of two refers to half of the complete cycle of traffic Light. So, we take 35sec of cycle.

But in our project, we have considered it as 25sec cycle to complete the traffic cycle.

Table 1 Data in previous system and Data in latest System

Data in previous system	Data in latest System
$C_{min} = \frac{35*0.8}{0.8-2*0.3} = 140sec$	$C_{min} = \frac{25*0.8}{0.8-2*0.3} = 100sec$
$C_{opt} = \frac{(1.5*35)+5}{1.0-2*0.3} = 143.55sec$	$C_{opt} = \frac{(1.5*25)+5}{1.0-2*0.3} = 106.25sec$

3.17. Hardware Tools

- Phone Camera: Phone cameras, equipped with network video recording capabilities, are among the most advanced options for real-time surveillance. These cameras are connected to a network, enabling remote access and continuous output via an IP address. They offer higher resolution and superior performance compared to standard cameras but come with a higher cost and require robust network connectivity.
- Arduino: Arduino is an open-source electronic platform that integrates both hardware and software to create interactive systems. It simplifies the process of programming microcontrollers, allowing users to write, compile, and upload code to an Arduino board using open-source software. Arduino supports various programming languages, including C/C++ and Micro Python, and is commonly used to control sensors, actuators, and other components in embedded systems.
- LEDs (Light Emitting Diodes): LEDs are used in various colors (red, yellow, and green) as traffic light indicators in this system. These LEDs are controlled using the Arduino IDE framework, making them easy to install and set up through the through-hole technology. The LEDs serve as visual traffic signals for the traffic management system.
- Jumper Wires: Jumper wires are used to transmit signals between the digital pins of the Arduino and the LEDs. They are essential for circuit prototyping and design. In this system, male-to-male jumper wires are utilized to transmit the signals that control the LED traffic lights.
- Frame: A rectangular wooden frame is used for prototyping the project. This frame emulates the setup of four different traffic lanes and houses the placement of LEDs as traffic signals, as well as the phone camera for traffic monitoring.

3.18. Software Tools:

- Python: Python is a versatile programming language that combines the capabilities of general-purpose languages with domain-specific ease of use. It includes a wide range of libraries for data loading, visualization, statistical analysis, natural language processing, image processing, and more. Python is well-suited for machine learning (ML) and data analysis tasks, where it enables iterative processes driven by data. It also allows the creation of complex graphical user interfaces (GUIs) and web services, as well as integration with existing systems. Python interacts seamlessly with code via tools like the terminal and Jupyter Notebook.
- Arduino IDE: The Arduino Integrated Development Environment (IDE) is a software tool designed for programming and managing Arduino microcontrollers. It provides an intuitive interface for writing, compiling, and uploading code to Arduino boards. In this project, the Arduino IDE is used to control three LEDs—red, yellow, and green—connected to the Arduino board. The Python programming language interfaces with the Arduino using the PyFirmata library for communication and control.

4. Results

The project presents an intelligent Traffic Management System designed to optimize vehicle flow at intersections. The system focuses on two primary intersections and is capable of detecting and counting vehicles to manage traffic light signals accordingly.

The model operates by detecting the number of vehicles in two lanes (Lane 1 and Lane 2). Based on vehicle count, the traffic light for each lane is activated. If the vehicle density in Lane 1 exceeds that in Lane 2, a green signal is triggered for Lane 1; otherwise, the green signal is given to Lane 2.

The traffic light cycle is optimized using the Webster formula, which initially provides a traffic light duration of 68 seconds. However, after optimization, the cycle time is adjusted to a more efficient 60 seconds. Upon completing one cycle, the system reassesses vehicle density and continues to adjust the traffic light flow, ensuring optimal vehicle passage. This dynamic flow allows the system to transition from a fixed-time scheduling method to an adaptive approach that continuously optimizes traffic management based on real-time data.

Fig. 5: Hardware and Image detection

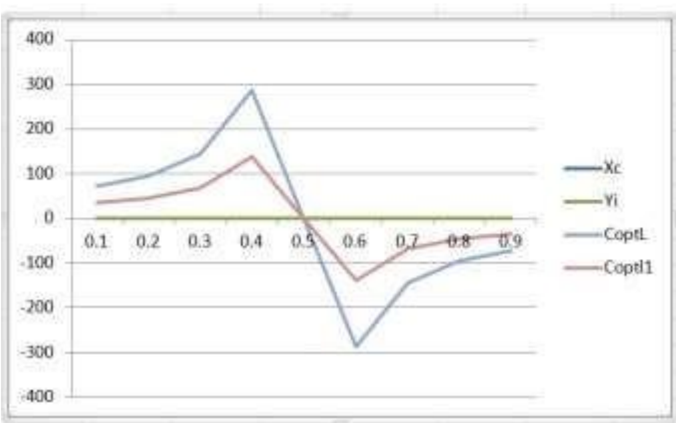

Fig. 4: Optimal cycle length for minimizing delay vs. flow Ratio (Yi)

5. Discussion

In the context of Nepal, 50% of traffic lights are powered by the ATmega328p microcontroller, while a small number use the STM32 microcontroller, and 8 traffic lights are based on a Programmable Logic Control (PLC) system. All these systems operate on fixed-time scheduling, which is predefined and static. These fixed-time algorithms lead to increased congestion, particularly when there is an imbalance in vehicle density between lanes. In some cases, one lane may experience unnecessary waiting times, even if the opposite lane has a lower vehicle count.

This study proposes an intelligent traffic management system that aims to address these issues by dynamically adjusting the traffic light signal based on real-time vehicle counting. The system uses cameras to detect and count the number of vehicles in each lane, adjusting the traffic light accordingly. By considering vehicle density and flow, the system can significantly reduce waiting times and optimize traffic management.

The system is particularly beneficial in areas where road expansion is not feasible due to limited space or heavy traffic demand. In places like Kathmandu Valley, this intelligent system can optimize traffic flow without the need for widening roads, which makes it highly suitable for urban areas with limited infrastructure.

6. Conclusions

The development of this intelligent traffic system demonstrates a promising solution for managing traffic congestion through dynamic vehicle count-based adjustments to traffic signals. Unlike traditional fixed-time scheduling systems, which operate on predefined time cycles, this adaptive system offers real-time responses to traffic conditions, improving the flow of vehicles. It is especially advantageous in regions where road expansion is impractical.

8. References

[1] Y. Liu, L. Liu, and W. P. Chen, "Intelligent traffic light control using distributed multi-agent Q learning," *IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC*, vol. 2018-March, pp. 1–8, Jul. 2017, doi: 10.1109/ITSC.2017.8317730.

[2] D. Zhao, Y. Dai, and Z. Zhang, "Computational intelligence in urban traffic signal control: A survey," *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews*, vol. 42, no. 4, pp. 485–494, Jul. 2012, doi: 10.1109/TSMCC.2011.2161577.

[3] V. Miz and V. Hahanov, "Smart traffic light in terms of the cognitive road traffic management system (CTMS) based on the Internet of Things," *Proceedings of IEEE East-West Design and Test Symposium, EWDTS 2014*, Jan. 2014, doi: 10.1109/EWDTS.2014.7027102.

[4] Z. Liu, "A Survey of Intelligence Methods in Urban Traffic Signal Control," *IJCSNS International Journal of Computer Science and Network Security*, vol. 7, no. 7, 2007.

[5] M. Bani Younes and A. Boukerche, "Intelligent Traffic Light Controlling Algorithms Using Vehicular Networks," *IEEE Trans Veh Technol*, vol. 65, no. 8, pp. 5887–5899, Aug. 2016, doi: 10.1109/TVT.2015.2472367.

[6] R. Zhang, A. Ishikawa, W. Wang, B. Striner, and O. K. Tonguz, "Using Reinforcement Learning with Partial Vehicle Detection for Intelligent Traffic Signal Control," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 1, pp. 404–415, Jul. 2018, doi: 10.1109/TITS.2019.2958859.

[7] M. B. Natafgi, M. Osman, A. S. Haidar, and L. Hamandi, "Smart Traffic Light System Using Machine Learning," *2018 IEEE International Multidisciplinary Conference on Engineering Technology, IMCET 2018*, Jul. 2018, doi: 10.1109/IMCET.2018.8603041.

[8] Y. Liu, L. Liu, and W. P. Chen, "Intelligent traffic light control using distributed multi-agent Q learning," *IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC*, vol. 2018-March, pp. 1–8, Jul. 2017, doi: 10.1109/ITSC.2017.8317730.

[9] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2016-December, pp. 779–788, Jun. 2015, doi: 10.1109/CVPR.2016.91.

[10] M. Sharma, A. Bansal, V. Kashyap, P. Goyal, and T. H. Sheikh, "Intelligent traffic light control system based on traffic environment using deep learning," *IOP Conf Ser Mater Sci Eng*, vol. 1022, no. 1, Jan. 2021, doi: 10.1088/1757-899X/1022/1/012122.

[11] T. Pan, "Traffic Light Control with Reinforcement Learning", Accessed: Jan. 09, 2025. [Online]. Available: https://github.com/OscarTaoyuPan/TrafficLightControl_QS.

[12] ultralytics, "COCO Dataset: All You Need to Know to Get Started." Accessed: Jan. 09, 2025. [Online]. Available: https://www.v7labs.com/blog/coco-dataset-guide

[13] D. Zhao, Y. Dai, and Z. Zhang, "Computational intelligence in urban traffic signal control: A survey," *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews*, vol. 42, no. 4, pp. 485–494, Jul. 2012, doi: 10.1109/TSMCC.2011.2161577.

[14] Y. Wang, T. Xu, X. Niu, C. Tan, E. Chen, and H. Xiong, “STMARL: A Spatio-Temporal Multi-Agent Reinforcement Learning Approach for Cooperative Traffic Light Control,” *IEEE Trans Mob Comput*, vol. 21, no. 6, pp. 2228–2242, Aug. 2019, doi: 10.1109/TMC.2020.3033782.

[15] S. S. Chavan, R. S. Deshpande, and J. G. Rana, “Design of intelligent traffic light controller using embedded system,” *2009 2nd International Conference on Emerging Trends in Engineering and Technology, ICETET 2009*, pp. 1086–1091, 2009, doi: 10.1109/ICETET.2009.76.

[16] A. N. Abood, “Traffic Light Control in Vehicular Network Systems using Fuzzy Logic,” *European Journal of Information Technologies and Computer Science*, vol. 4, no. 4, pp. 1–6, Sep. 2024, doi: 10.24018/COMPUTE.2024.4.4.132.

[17] W. H. Lee and C. Y. Chiu, “Design and Implementation of a Smart Traffic Signal Control System for Smart City Applications,” *Sensors (Basel)*, vol. 20, no. 2, Jan. 2020, doi: 10.3390/S20020508.

[18] Y. Liu, L. Liu, and W. P. Chen, “Intelligent traffic light control using distributed multi-agent Q learning,” *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*, vol. 2018-March, pp. 1–8, Jul. 2017, doi: 10.1109/ITSC.2017.8317730.

[19] I. Kosonen, “Multi-agent fuzzy signal control based on real-time simulation,” *Transp Res Part C Emerg Technol*, vol. 11, no. 5, pp. 389–403, Oct. 2003, doi: 10.1016/S0968-090X(03)00032-9.

[20] D. Roozemond, “Using intelligent agents for pro-active, real-time urban intersection control,” *Eur J Oper Res*, vol. 131, pp. 293–301, 2001, Accessed: Jan. 09, 2025. [Online]. Available: https://research.tudelft.nl/en/publications/using-intelligent-agents-for-pro-active-real-time-urban-intersect

[21] R. S. Sutton and A. G. Barto, “Reinforcement Learning: An Introduction Second edition, in progress,” 2014.

[22] J. Liu *et al.*, “Traffic signal control using reinforcement learning based on the teacher-student framework,” *Expert Syst Appl*, vol. 228, p. 120458, Oct. 2023, doi: 10.1016/J.ESWA.2023.120458.

[23] R. Zhang, A. Ishikawa, W. Wang, B. Striner, and O. K. Tonguz, “Using Reinforcement Learning With Partial Vehicle Detection for Intelligent Traffic Signal Control,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 1, pp. 404–415, Jan. 2021, doi: 10.1109/TITS.2019.2958859.

[24] S. S. Chavan, R. S. Deshpande, and J. G. Rana, “Design of intelligent traffic light controller using embedded system,” *2009 2nd International Conference on Emerging Trends in Engineering and Technology, ICETET 2009*, pp. 1086–1091, 2009, doi: 10.1109/ICETET.2009.76.

[25] L. F. P. De Oliveira, L. T. Manera, and P. D. G. Da Luz, “Development of a Smart Traffic Light Control System with Real-Time Monitoring,” *IEEE Internet Things J*, vol. 8, no. 5, pp. 3384–3393, Mar. 2021, doi: 10.1109/JIOT.2020.3022392.

[26] M. M. Gandhi, D. S. Solanki, R. S. Daptardar, and N. S. Baloorkar, “Smart Control of Traffic Light Using Artificial Intelligence,” *2020 5th IEEE International Conference on Recent Advances and Innovations in Engineering, ICRAIE 2020 - Proceeding*, Dec. 2020, doi: 10.1109/ICRAIE51050.2020.9358334.

[27] A. S. Salama, B. K. Saleh, and M. M. Eassa, “Intelligent cross road traffic management system (ICRTMS),” *ICCTD 2010 - 2010 2nd International Conference on Computer Technology and Development, Proceedings*, pp. 27–31, 2010, doi: 10.1109/ICCTD.2010.5646059.

[28] I. Kosonen and K.-L. Bång, “Multi-Agent Fuzzy Signal Control Based on Real-Time Simulation,” 2001, *ITS Australia*. Accessed: Jan. 09, 2025. [Online]. Available: https://research.aalto.fi/en/publications/multi-agent-fuzzy-signal-control-based-on-real-time-simulation-2

[29] M. Sharma, A. Bansal, V. Kashyap, P. Goyal, and T. H. Sheikh, “Intelligent Traffic Light Control System Based On Traffic Environment Using Deep Learning,” *IOP Conf Ser Mater Sci Eng*, vol. 1022, no. 1, p. 012122, Jan. 2021, doi: 10.1088/1757-899X/1022/1/012122.