CTU Journal of Innovation and Sustainable Development
ISSN 2588-1418 | e-ISSN 2815-6412

DOI:10.22144/ctujoisd.2024.302

Development of the intelligent traffic light system based on image processing and fuzzy control techniques

Hoang-Dung Nguyen*, Hoang-Dang Le, Van Khanh Nguyen, and Hung-Minh Lam

College of Engineering, Can Tho University, Viet Nam

**Corresponding author (hoangdung@ctu.edu.vn)*

Article info.

Received 27 Feb 2024
Revised 25 Mar 2024
Accepted 25 Jun 2024

Keywords

Fuzzy logic control, intelligent traffic light, objective recognition, YOLOv8

ABSTRACT

In Viet Nam's current traffic conditions, congestion and jams—especially at intersections during peak hours—present major challenges. Traditional traffic light systems, which rely on fixed timing principles, often fail to manage traffic flow efficiently, particularly when vehicle density varies significantly across different directions. This research aims to develop an intelligent traffic light system where the signal timings automatically adjust based on the vehicle density at intersections. The study uses an object recognition algorithm to identify, classify, and count vehicles. The data was then fed into a fuzzy logic model to calculate the optimal signal timings. Experimental results demonstrate an accuracy of approximately 88% in vehicle detection. The fuzzy logic model and the programmable logic controller were able to effectively compute reasonable signal timings based on real-time vehicle density. Future developments include expanding the system's functionalities, creating a user-friendly interface, and developing a management application for mobile devices.

1. INTRODUCTION

The widespread use of motorcycles and scooters has inevitably contributed to traffic congestion, especially during peak hours at major intersections and along main roads (Yousif et al., 2020; Damani & Vedagiri, 2021; Huu & Ngoc, 2021; Hidayati et al., 2022). This issue is not unique to Viet Nam; it is a global challenge that affects both developed and developing countries. The consequences of traffic congestion are far-reaching, impacting various aspects of social life (Gössling, 2018; Robinson & Thagesen, 2018).

Increased commute times, higher fuel consumption, vehicle wear and tear, elevated transportation costs, heightened emissions, noise pollution, reduced urban traffic flexibility, and a decline in the quality of urban living environments are among the negative effects (Chen & Kockelman, 2016; Van Dender, 2019). These repercussions extend beyond individual inconvenience, affecting both communal and economic aspects of life.

Furthermore, daily time losses due to traffic congestion reduce work efficiency and productivity for citizens (Sweet, 2014; Çolak et al., 2016). The increased fuel consumption and vehicle wear and tear burden drivers and impose financial and environmental pressures. Rising transportation costs add to the financial burdens of individuals and businesses, posing challenges in transportation expenses and commodity prices (Button, 2010; Dewita et al., 2018; Wensveen, 2023). Moreover, traffic congestion adversely affects the mental health of the community, creating a stressful and uncomfortable daily living environment.

Addressing the issue of traffic congestion requires a comprehensive approach involving infrastructure improvements, smart urban planning, and community engagement to create a positive and sustainable living and working environment (Khansari et al., 2014; Angelidou et al., 2018; Shamsuzzoha et al., 2021).

In advanced countries, an effective solution to alleviate traffic congestion is installing camera systems for automated traffic regulation at critical intersections (Datondji et al., 2016; Makino et al., 2018). Throughout the evolution of traffic control systems, numerous studies have concentrated on simulations and experiments to optimize traffic signal controllers (Namazi et al., 2019; Olayode et al., 2020; Qadri et al., 2020). Notably, successful applications have employed image-processing techniques in conjunction with fuzzy control.

Recently, various papers have emerged proposing intelligent solutions for controlling traffic light systems. Ngon (2010), introduced a system for crossroad traffic lights utilizing two cameras to monitor the traffic flow on two roads. Greenlight signal durations were determined based on traffic flow, with results from 450 experiments revealing a range of durations. In addition, Chabchoub et al., (2021) presented a smart traffic light controller employing fuzzy logic and MATLAB image processing to manage traffic in two directions, with simulation results reflecting efficient operation. An intelligent traffic light control system controlled based on the fuzzy logic algorithm and PLC (Programmable logic controller) was developed by Bouhedda et al., (2021). They found a significant reduction in waiting times compared to traditional algorithms, particularly at high flow rates. Additionally, Mujčić et al. (2022) proposed a fuzzy logic-based system incorporating surveillance cameras for vehicle detection and PIR sensors for pedestrian detection, with simulation analysis validating its effectiveness. A novel method utilizing MATLAB and cameras for real-time image processing to adjust green signal durations based on traffic density and congestion levels was also suggested by Wath et al. (2022).

The integration of You Only Look Once (YOLO) object detection with fuzzy logic control in intelligent traffic light systems marks a significant advancement in traffic management, as demonstrated by Huang et al. (2023) and Lin and Jhang (2022). YOLO, a real-time object detection algorithm, efficiently identifies vehicles, pedestrians, and other objects in traffic scenes (Wang et al., 2023). This information feeds into a fuzzy logic controller, which makes dynamic decisions about signal timings based on the current traffic environment (Jutury et al., 2023). These systems improve efficiency, reduce congestion, and enhance road safety by continuously adapting to changing conditions. The benefits extend to energy efficiency, as optimized signal timings can reduce idling and fuel consumption, leading to lower emissions.

The proposed solution advocates creating an intelligent traffic light system that utilizes image processing and fuzzy control techniques. Cameras stationed at two roadways at an intersection will collect data, allowing the system to adjust green light durations dynamically based on the traffic volume. Roads with higher traffic will enjoy extended green times, effectively optimizing traffic flow and mitigating congestion at these intersections.

2. MATERIALS AND METHOD

2.1. YOLOv8 model

YOLOv8, developed and maintained by the Ultralytics team, is the latest iteration in the YOLO series of real-time object detection models. Originating from Joseph Redmon's work, who created the initial YOLO models, YOLOv3 served as the basis for YOLOv5, adapted by Glenn Jocher in PyTorch. YOLOv8, introduced on January 10, 2023, continues to undergo active research and development (Terven & Cordova-Esparza, 2023).

YOLOv8 introduces more parameters than its predecessors, like YOLOv5, but fewer than YOLOv6, as depicted in Figure 1. Despite the increased number of parameters, it delivers approximately 38% higher mean Average Precision (mAP) for models of size n and generally higher mAP. Additionally, YOLOv8 exhibits faster inference times compared to all other versions. This performance enhancement showcases its efficiency in balancing model complexity and accuracy, making it a compelling choice for real-time object detection tasks. The trade-off between parameters and mAP highlights YOLOv8's optimization for improved precision without sacrificing computational speed, reinforcing its position as an advanced solution within the YOLO series (Terven et al., 2023; Wang et al., 2023).

The latest version builds upon its predecessors, incorporating new features and optimizations for enhanced accuracy and processing speed. YOLOv8 is ideal for diverse object detection tasks across various application fields. Its evolution showcases the collaborative effort within the computer vision community to advance real-time object detection capabilities (Farooq et al., 2023; Talaat & ZainEldin, 2023).

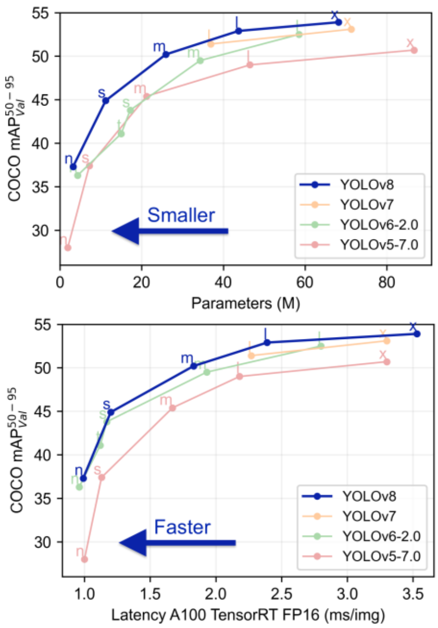


Figure 1. Comparion of YOLOv8 with previous versions (Terven & Cordova-Esparza, 2023)

In YOLOv8, there are various model sizes denoted as n (nano), s (small), m (medium), l (large), and x (extra-large). The details of each model size's parameters are presented in Table 1. This work provides flexibility for users, allowing them to choose a YOLOv8 version that suits their specific requirements, optimizing performance according to the desired model size (Farooq et al., 2023; Terven & Cordova-Esparza, 2023).

The model size correlates linearly with mAP (mean Average Precision) and inversely with inference time. This research indicates that larger models take more time for object detection but are more accurate due to higher mAP. On the other hand, smaller models have faster inference times but are less accurate as they exhibit lower mAP. However, since the Params and Flops of YOLOv8n models are smaller than others, they are suitable for tasks that do not require high computational resources. In summary, the choice of model size in YOLOv8 is a trade-off between accuracy (mAP) and inference speed. More extensive models offer higher accuracy at the cost of longer inference times, while smaller models provide faster inference but with reduced accuracy.

Table 1. Comparison of YOLOv8 model sizes (Terven & Cordova-Esparza, 2023)

Model	Size (pixels)	mAP^{val} 50-95	Speed CPU ONNX (ms)	Speed A100 TensorRT (ms)	Params (M)	Flops (B)
YOLOv8n	640	37.3	80.4	0.99	3.2	8.7
YOLOv8s	640	44.9	128.4	1.20	11.2	28.6
YOLOv8m	640	50.2	234.7	1.83	25.9	78.9
YOLOv8l	640	52.9	375.2	2.39	43.7	165.2
YOLOv8x	640	53.9	479.1	3.53	68.2	257.8

2.2. Fuzzy control

2.2.1. Introduction of fuzzy logic

Fuzzy logic is a system that handles uncertainty and vagueness in information and decision-making. Unlike traditional logic systems with binary values of true or false, fuzzy logic allows logic values and functions to range between 0 and 1, representing the degree of truth in a response within uncertain or ambiguous contexts (Lootsma, 2013; Zadeh, 2023).

Fuzzy logic finds applications in diverse fields, such as fuzzy assessment systems, robot control, image blurring, and natural language processing. It excels in handling situations where information is unclear and exhibits a fuzzy or uncertain nature. In fuzzy assessment systems, it adapts to evaluate based on fuzzy truth values, accommodating the ambiguous nature of information. In robot control, it aids in adapting to uncertain and dynamic environments (Bouteraa et al., 2020; Khairudin et al., 2020; Zangeneh et al., 2022). Fuzzy logic is employed in image blurring for unclear information and natural language processing to understand and process ambiguous language constructs (Dzyubanenko & Rabin, 2022).

2.2.2. Scikit-fuzzy tool (Skfuzzy)

Skfuzzy is a framework based on fuzzy logic in information processing. Specifically, it is a software library developed to support the modeling and implementation of systems using fuzzy logic for evaluation, decision-making, and control across various applications. This tool facilitates programmers and researchers in conducting analyses and experiments related to fuzzy logic. Skfuzzy is a library of fuzzy logic algorithms designed for use within the SciPy toolkit and written in Python (Chegini et al., 2021; Yazdinejad et al., 2023). It provides a set of fuzzy logic algorithms for users to apply to their projects.

The key features include:

– *Fuzzy Modeling:* Skfuzzy uses fuzzy language to represent variables and fuzzy relationships in a system.

– *Rule and Knowledge Management:* This tool defines and manages fuzzy rules as a set of rule sets.

– *Fuzzy Set Operations:* Skfuzzy supports operations on fuzzy sets such as union and intersection.

– *Programming Interface:* It offers an Application Programming Interface (API) for integrating fuzzy logic into other applications and systems.

– *Evaluation and Testing:* Enables testing and evaluating fuzzy logic systems to ensure correct and effective operation.

– *Versatile Application Support:* Skfuzzy can be used in various applications like automatic control, information systems, performance evaluation, and more.

Skfuzzy empowers users to build and test fuzzy logic systems efficiently and easily, enhancing decision-making and control in situations characterized by ambiguity and uncertainty.

3. SMART TRAFFIC LIGHT SYSTEMS

3.1. Operating principle of smart traffic light system

The smart traffic light system utilizes two cameras installed on two roads at the same intersection, capturing real-time images of various objects. YOLOv8, detects key objects such as motorbikes and cars while simultaneously counting the number of vehicles. This information is then integrated into a simple fuzzy control system comprising 25 fuzzy rules to assess traffic conditions on both roadways at the intersection, as shown in Figures (3) and (4). Traffic conditions are categorized as “very low (V_low)”, “low”, “moderate”, “high”, and “very high (V_high)” based on the counted number of vehicles, as presented in Table 2.

Table 2. The number of vehicles corresponding to traffic levels

Number of vehicles	Levels
0 – 15	V_low
10 – 25	Low
16 – 44	Medium
35 – 50	High
45 – 60	V_high

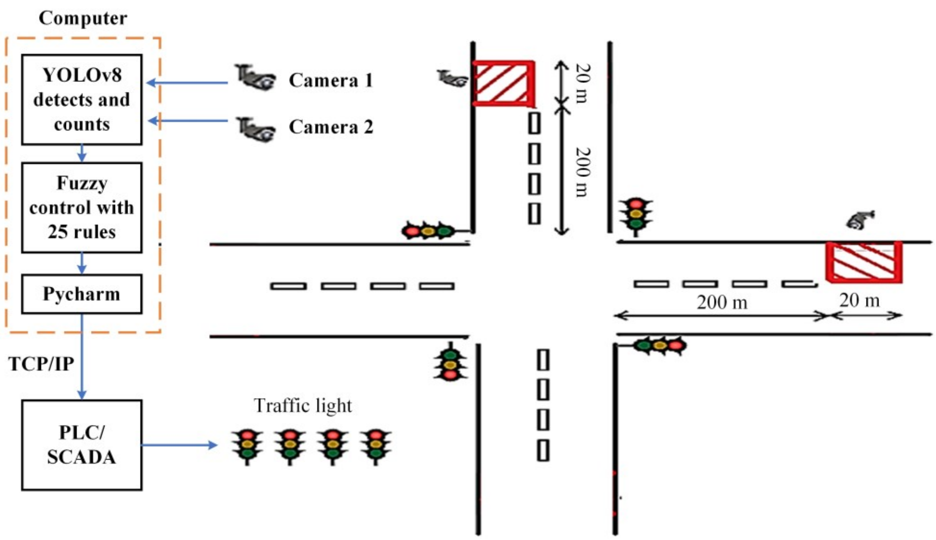


Figure 2. Schematic of smart traffic light system

The camera placement distance to the traffic light pole is 200 meters, and the detection area has a length of 20 meters, so the distance from the object detection to the traffic light pole is 220 meters (as shown in Figure 2). The reason for choosing 200 meters is to ensure that the counting start time is neither too long nor too short. If counting takes too long, the number of vehicles on both roadways will always be congested, potentially leading to traffic jams and affecting subsequent green light cycles. If counting takes too short, one roadway will always have fewer vehicles while the other remains congested.

The maximum speed in the city is 40 km/h, equivalent to 11 m/s. The start time for counting can be determined as 220/11=20 seconds. However, maintaining a constant speed of 11 m/s is impossible due to closely spaced vehicles and other situations. Therefore, the start time for counting vehicles was in the last 30 seconds of the green light on the first roadway (as during these 30 seconds, when the counted vehicles reach the traffic light pole, the light will change to red, meaning the counted vehicles are still within the red light stopping area) and during the red light cycle of the second roadway. During the yellow light time on the first roadway, the green light time for both roadways will be reset, and the counting variable will be reset. This decision aims to avoid inaccuracies in the green light time compared to the actual number of vehicles.

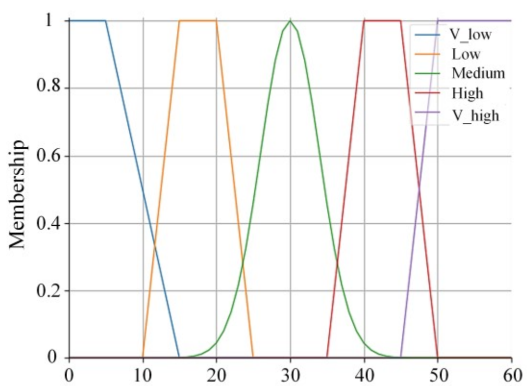


Figure 3. Input rules of fuzzy controller

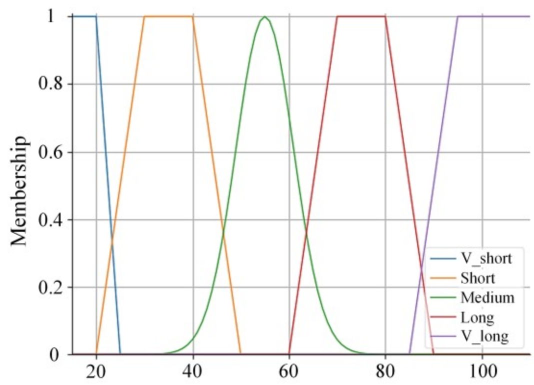


Figure 4. Output rules of fuzzy controller

Table 3. The green light time corresponds to the levels

Greenlight time (s)	Levels
15 - 25	V_short
20 – 50	Short
35 – 75	Medium
60 – 90	Long
85 – 110	V_long

After simultaneously assessing the traffic conditions on both roadways, the fuzzy controller activates a comparison based on established fuzzy rules and proposes solutions to prevent congestion. These solutions include adjusting the green and red light transition times on both roadways, with options such as "very short (V_short)", "short", "medium", "long", and "very long (V_long)", as described in Table 3. A specific fuzzy control system with 25 rules and traffic lanes (L1/L2) is described in Table 4.

Table 4. Fuzzy control with 25 rules

L1 \ L2	V_low	Low	Medium	High	V_high
V_low	L1: Medium L2: Medium	L1: Short L2: Long	L1: Short L2: Long	L1: V_short L2: V_long	L1: V_short L2: V_long
Low	L1: Long L2: Short	L1: Medium L2: Medium	L1: Short L2: Long	L1: V_short L2: Long	L1: V_short L2: V_long
Medium	L1: Long L2: Short	L1: Long L2: Short	L1: Medium L2: Medium	L1: Short L2: Long	L1: Short L2: Long
High	L1: V_long L2: V_short	L1: V_long L2: Short	L1: Long L2: Short	L1: Medium L2: Medium	L1: Short L2: Long
V_high	L1: V_long L2: V_short	L1: V_long L2: V_short	L1: Long L2: Short	L1: Long L2: Short	L1: Medium L2: Medium

3.2. Data collection and manually assign labels

Images captured from traffic surveillance cameras are sourced from the Internet or extracted frames from videos. Before labeling in Figure 5, resizing is a crucial step. Images need to be adjusted to a standardized size of 640×640 pixels to achieve uniformity in the data and optimize the labeling process. This work helps reduce data complexity, establishing a standardized foundation for identifying and classifying traffic objects. This process ensures that the machine learning model can effectively process the data, providing accurate and reliable results in monitoring and managing traffic.

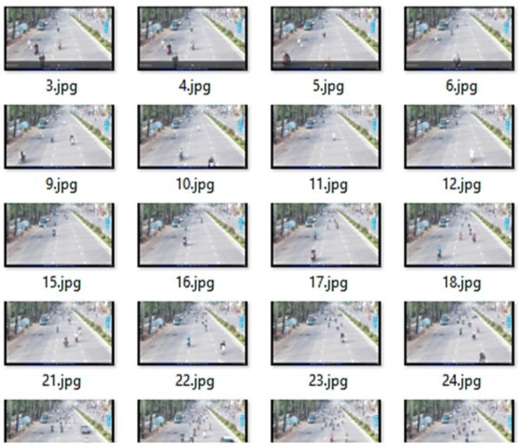


Figure 5. Images data for training

Manually labeling over 800 collected data using tools like LabelImg or MakeSense is crucial, as shown in Figure 6. First, prepare the data by ensuring they are stored appropriately. Install a labeling tool like LabelImg and upload the images to a new project. Manually label by drawing bounding boxes around objects and assigning labels. Save label information and meticulously check for errors. Finally, the labeled data will be stored to train a machine-learning model.

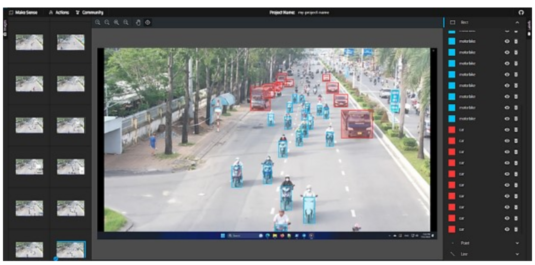


Figure 6. Manually assign labels using the Makesense tool

3.3. Train and test results

Performing 99 training iterations for the YOLOv8n model on Google Colab, using pre-labeled images and labels, is critical. After training, the model's accuracy can be assessed using a confusion matrix. The matrix evaluates the accuracy of predictions (from 0 to over 800) between predicted (Predicted) and actual (True) objects. Since the training objects included only motorbikes and cars, squares comparing the two objects in Predicted and True, ranging from 700 to over 800, signify the model's effective object detection. Additionally, training results can be directly assessed using images containing trained objects, as shown in Figure 7. Once the model met the desired standards, a program can be developed to access video data and control a smart traffic light system.

Figure 7. Image test using the YOLOv8n model

3.4. Connection to SCADA software

Several steps need to be taken to deploy the YOLO tracking application and integrate it with PLC along with image processing/fuzzy controller shown in Figure 8: (1) Firstly, Windows and basic parameters must be established to prepare for the tracking process; (2) next, connecting to the PLC through Modbus TCP (Transmission control protocol) protocol to access and control necessary devices; (3) identifying the area for tracking and create sets to store the IDs of recognized objects and choosing a suitable YOLO model and develop a user interface to display information about recognized objects and control actions;

(4) subsequently, setting an antecedent for the fuzzy controller to adjust behavior based on input data from the PLC and image processing program; (5) defining the input and output variables for the fuzzy controller to execute the adjustment process; (6) utilizing the YOLO model to detect and track objects, then extract and display information on the user interface; (7) finally, the values of train variables and control data transmission through Modbus TCP are checked based on values read from registers. Record the number of counted objects in registers and display this information on the user interface to monitor and analyze results.

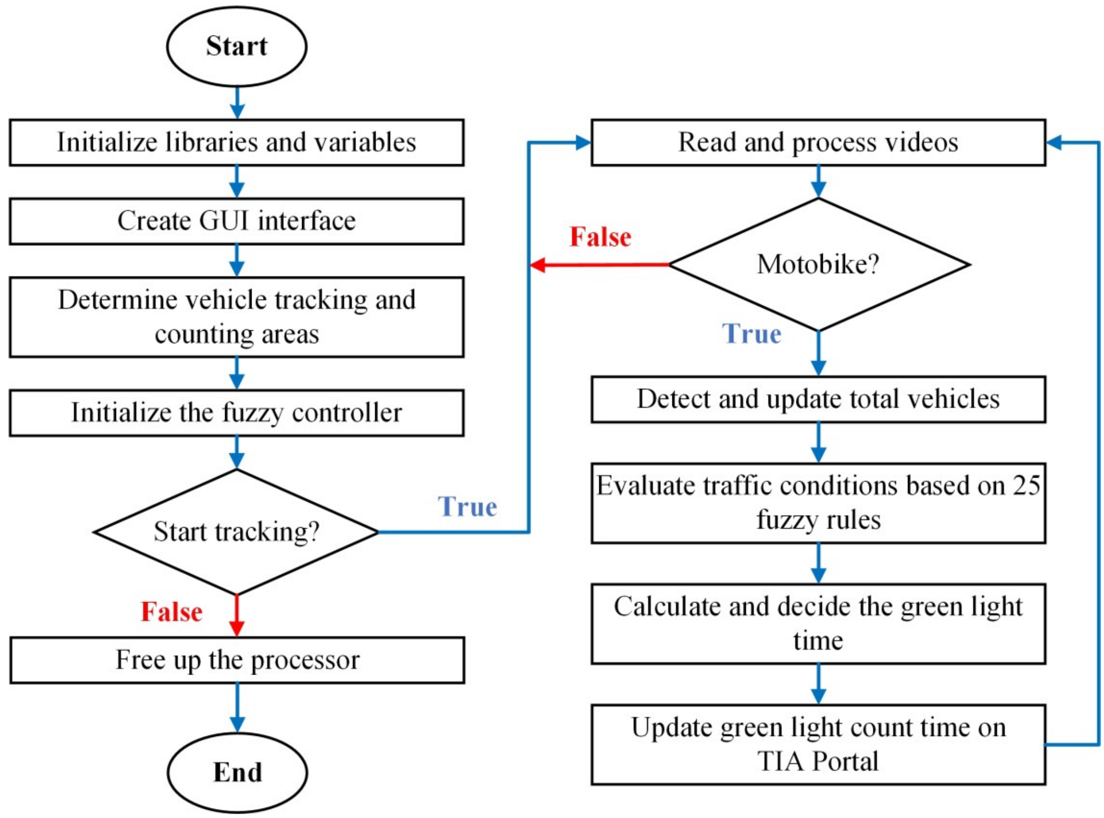


Figure 8. The flow chart on Pycharm connection to PLCs

After completing the programs in PyCharm and the traffic light program in the TIA Portal, the creation of a SCADA interface in Figure 9 has allowed for easy observation and monitoring of the smart traffic light system. Through Ethernet, PLC devices can efficiently send and receive data with the SCADA screen, optimizing the management and supervision of the system. This interface facilitates effective visualization, enabling real-time monitoring of the intelligent traffic light system's activities. The integration of PyCharm and TIA Portal, along with the SCADA interface, enhances the overall efficiency and control of the system.

Connecting to PLC devices can efficiently send and receive data with the SCADA screen, optimizing system management and monitoring. The actual hardware connection between 2 PLCs and SCADA. This work enables continuous communication between components, allowing SCADA to gather data from PLCs and control them as needed. This paper improves system performance and enhances flexibility and adjustability in intelligent traffic light systems.

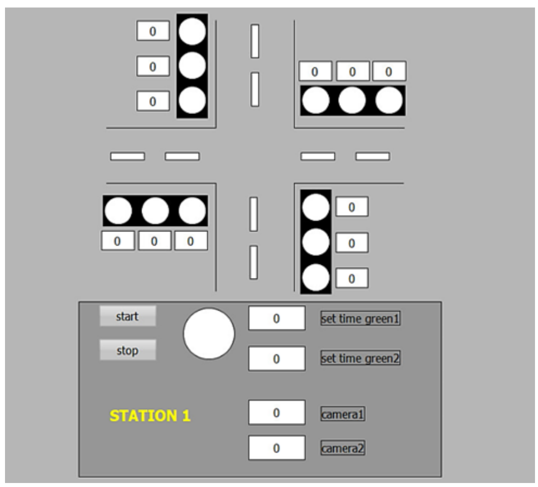
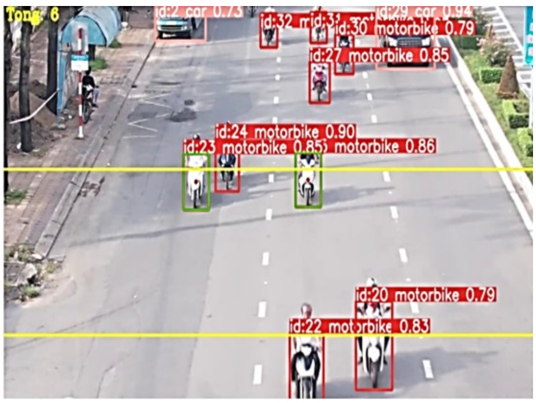


Figure 9. SCADA interface for smart traffic light system

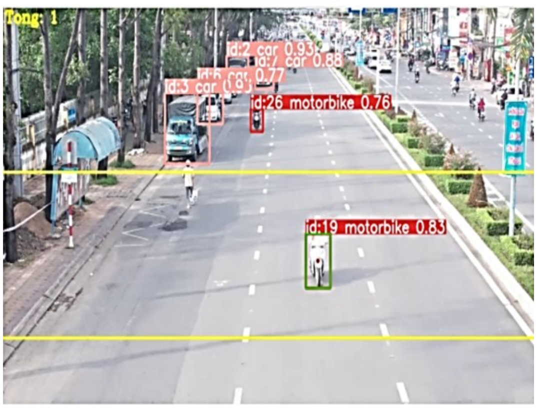
4. RESULTS AND DISCUSSION

Figure 10 presents the results of detecting and counting various traffic vehicles, including cars and motorcycles, with no missed objects. The model efficiently detects vehicles from a distance, incrementing the count only when they enter the designated counting area (highlighted in yellow). This approach reduces counting errors and improves system accuracy. The results highlight the model's flexibility in counting traffic vehicles, demonstrating its effectiveness in real-world conditions and its ability to accurately track the movement of objects.

Figure 10. Boundary and detection of the objective



Total = 06



Total = 01

Figure 11. Display two cameras at the same time

The solution can simultaneously run and display feeds from two cameras at the intersection of the same junction. It can efficiently detect, count, and store data into separate counters for each roadway. This functionality is illustrated in Figure 11, showcasing the system's ability to manage and process data from both cameras concurrently, providing a comprehensive overview of traffic dynamics at the intersection.

The design solution has been validated through experimental models in 9 diverse scenarios. These situations assessed the solution's performance in various traffic contexts. In Table 5, we illustrated both cases: when the traffic flow on both roadways is the same and when the traffic flow on two different roads is considered. Results from the experimental model provide robust evidence for the effectiveness and broad applicability of the design solution. The diversity of scenarios helped confirm the solution's adaptability, ensuring it can be applied effectively across various traffic contexts. The proposed enhances the reliability and practicality of the proposed design solution.

Table 5. Experimental green light time results

Cases	**Green light line 1 (s)**	**Green light line 2 (s)**
L2 V_high, L1 V_low	19 ± 1	98 ± 1
L2 High, L1 Low	23 ± 3	74 ± 3
L2 High, L1 Medium	35 ± 3	72 ± 3
L2 Medium, L1 Low	39 ± 3	71 ± 3
L1, L2 Medium	55 ± 1	55 ± 1
L1 Medium, L2 Low	71 ± 3	39 ± 3
L1 High, L2 Medium	72 ± 3	35 ± 3
L1 High, L2 Low	74 ± 3	23 ± 3
L1 V_high, L2 V_low	98 ± 1	19 ± 1

The experimental results highlight the effectiveness of the proposed algorithm, which relies on the CNN architecture YOLOv8. This algorithm achieved an impressive 88% accuracy in detecting, counting, and classifying traffic vehicles, as shown in Figure 11. The integration of YOLOv8 with the system demonstrated its robust capabilities in accurately identifying and categorizing diverse traffic scenarios. This high level of precision is crucial for the reliable operation of the smart traffic light system. The combination of the fuzzy model and PLC controller further enhances the system's adaptability, ensuring that adjustments in timing align with the real traffic density. In general, the experiments confirm the algorithm's capability of significantly improving the efficiency and accuracy of traffic management in the context of smart traffic light systems.

In intelligent traffic light systems, the standard deviation time represents the variability in signal change intervals at intersections (Kamal et al., 2019). Minimizing this variability enhances traffic flow, reduces congestion, and improves safety. Achieved through sensors, algorithms, and real-time data, these systems dynamically adjust signal timings based on traffic conditions and pedestrian activity. By reducing standard deviation time, drivers experience more predictable signal changes, leading to smoother traffic flow and potentially shorter wait times. The standard deviation (σ) is calculated using the following formula:

$$\sigma = \sqrt{\frac{\sum_{i=1}^{N}\left(x_i - \overline{x}\right)^2}{N}} \qquad (1)$$

Where N is the total number of time results, x_i is the number of values, $\overline{x}$ is the average value.

The accuracy of the results is assessed by comparing continuous timestamp data displayed in the PyCharm terminal window. The duration of the green light on both roadways showed variability, with time stamps differing by 1 to 3 seconds. The standard deviation time, a measure of data dispersion, is determined to be 0.506 seconds. This standard deviation time is computed using the Excel function, with the shortest green light duration chosen as the reference point. These findings are detailed in Table 6, providing a clearer understanding of the accuracy and stability of the system in measuring and monitoring green light durations.

The longest green light duration is 98 ± 1 seconds, indicating a range within one second of variability. Conversely, the shortest green light duration is measured at 19 ± 1 second, suggesting a similarly narrow range of potential variability. These specific values precisely characterize the system's performance, emphasizing its ability to consistently manage and control the traffic signal timings within tight tolerances.

The current smart traffic light system exhibits shortcomings in functionality and interface simplicity. However, there is potential for improvement through various enhancements. Firstly, essential features such as monitoring traffic density, integrating safety sensors for pedestrians and cyclists, and prioritizing traffic for specific vehicles should be added. Secondly, the system should calculate the synchronization time between traffic lights to optimize traffic flow, preventing consecutive red lights at intersections. Additionally, developing a more user-friendly interface is crucial, incorporating features like graphical statistics and performance reports for administrators to monitor and make informed decisions effectively. Lastly, creating a mobile application for remote system management, configuration updates, status tracking, and alert notifications enhances convenience. This paper facilitates improved monitoring and control capabilities, ultimately enhancing the efficiency and safety of urban traffic.

Table 6. Standard deviation time results

Desired value (s)	Actual value (s)
19	18
19	18
19	18
19	18
19	19
19	19
19	19
19	19
19	19
19	19
19	19
19	19
19	18
Average value	18.615
Standard deviation	0.506

Different methods have been developed to accurately detect and count vehicles in real-time in various intelligent traffic light systems, as shown in Table 7. Xiang et al. (2018) utilized multi-threading techniques, achieving an 85% accuracy rate. Whilst, Yang and Qu (2018) employed background subtraction algorithms, resulting in 82.35% accuracy. Moranduzzo and Melgani (2013) focused on automatic car counting, reaching 76.61% accuracy. Bouvié et al. (2013) utilized particle filtering, achieving an accuracy of 78.6%. Froidevaux et al. (2020) introduced Tiramisu, a deep learning architecture, with an accuracy of 86.2%. Ghosh et al. (2019) utilized adaptive video-based methods, resulting in 82% accuracy. However, the most recent work presented in this comparison, which incorporates YOLOv8 object detection and fuzzy control, demonstrates the highest accuracy of 88%. This proposed system outperforms previous methods, making it promising for enhancing intelligent traffic light systems.

Table 7. A comparison of accuracy in real-time vehicle detection and counting

References	Methods	Accuracy (%)
(Xiang et al., 2018)	Multi-threading	85
(Yang & Qu, 2018)	Background subtraction	82.35
(Moranduzzo & Melgani, 2013)	Automatic car counting	76.61
(Bouvié et al., 2013)	Particle filtering	78.6
(Froidevaux et al., 2020)	Tiramisu	86.2
(Ghosh et al., 2019)	Adaptive video-based	82
This work	YOLOv8 and Fuzzy control	88

The research aims to enhance the accuracy of an intelligent traffic light system by improving various components. Data quality is improved through high-resolution cameras and preprocessing techniques to enhance accuracy. The algorithm is continuously refined, potentially incorporating deep learning methods for better classification. The fuzzy logic model, responsible for determining optimal signal timings based on vehicle density, undergoes refinement by adjusting parameters and incorporating additional factors like pedestrian flow. Optimization algorithms are explored to fine-tune signal timings, potentially integrating machine learning to predict traffic patterns. Rigorous testing and validation are conducted using real-time data, with a feedback mechanism implemented to iteratively improve the system based on performance observations.

5. CONCLUSIONS

This study successfully achieves its objectives, including the recognition of traffic vehicles, the implementation of a fuzzy controller, and the establishment of communication between PyCharm and the PLC. Utilizing the precise decision-making abilities of the fuzzy controller, the research develops an algorithm for efficient vehicle detection and counting, catering to the smart traffic light system. The interconnected PLC S7-1500s visualized through SCADA, enhances monitoring and operational efficiency. Experimental results reveal an 88% accuracy in detecting, counting, and classifying traffic vehicles using the proposed YOLOv8 CNN-based algorithm. Combining the fuzzy model and the PLC controller ensures timely and reasonable adjustments based on real traffic density, contributing to effective and accurate smart traffic light system operations.

ACKNOWLEDGEMENT

We would like to extend our sincere thanks to Mr. Minh Triet Tran and Mr. Truong Thanh Duong for dedicating their valuable time to collecting data and labeling vehicle objects.

REFERENCES

Angelidou, M., Psaltoglou, A., Komninos, N., Kakderi, C., Tsarchopoulos, P., & Panori, A. (2018). Enhancing sustainable urban development through smart city applications. *Journal of Science and Technology Policy Management*, *9*(2), 146-169.

Bouhedda, M., Benyezza, H., Toumi, Y., & Rebouh, S. (2021). Fuzzy traffic lights controller based on PLC. Artificial Intelligence and Renewables Towards an Energy Transition 4,

Bouteraa, Y., Abdallah, I. B., ElMogy, A., Ibrahim, A., Tariq, U., & Ahmad, T. (2020). A fuzzy logic architecture for rehabilitation robotic systems. *International Journal of Computers Communications & Control*, *15*(4).

Bouvié, C., Scharcanski, J., Barcellos, P., & Escouto, F. L. (2013). Tracking and counting vehicles in traffic video sequences using particle filtering. 2*013 IEEE international instrumentation and measurement technology conference (I2MTC)*.

Button, K. (2010). *Transport economics*. Edward Elgar Publishing.

Chabchoub, A., Hamouda, A., Al-Ahmadi, S., & Cherif, A. (2021). Intelligent traffic light controller using fuzzy logic and image processing. *International Journal of Advanced Computer Science and Applications*, *12*(4), 396-399.

Chegini, H., Beltran, F., & Mahanti, A. (2021). Fuzzy logic based pasture assessment using weed and bare patch detection. In *Smart and Sustainable Agriculture: First International Conference, SSA 2021, Virtual Event, June 21-22, 2021, Proceedings 1* (pp. 1-18). Springer International Publishing.

Chen, T. D., & Kockelman, K. M. (2016). Carsharing's life-cycle impacts on energy use and greenhouse gas emissions. *Transportation Research Part D: Transport and Environment*, *47*, 276-284.

Çolak, S., Lima, A., & González, M. C. (2016). Understanding congested travel in urban areas. *Nature Communications*, *7*(1), 10793.

Damani, J., & Vedagiri, P. (2021). Safety of motorised two wheelers in mixed traffic conditions: Literature review of risk factors. *Journal of Traffic and Transportation Engineering (English edition)*, *8*(1), 35-56.

Datondji, S. R. E., Dupuis, Y., Subirats, P., & Vasseur, P. (2016). A survey of vision-based traffic monitoring of road intersections. *IEEE transactions on intelligent transportation systems*, *17*(10), 2681-2698.

Dewita, Y., Yen, B. T., & Burke, M. (2018). The effect of transport cost on housing affordability: Experiences from the Bandung Metropolitan Area, Indonesia. *Land Use Policy*, *79*, 507-519.

Dzyubanenko, A. A., & Rabin, A. V. (2022). Application of the method of metaontologies in the intellectual analysis of text resources considering the fuzziness and blurring of images of natural language units. In *AIP Conference Proceedings* (Vol. 2647, No. 1). AIP Publishing.

Farooq, J., Muaz, M., Khan Jadoon, K., Aafaq, N., & Khan, M. K. A. (2023). An improved YOLOv8 for foreign object debris detection with optimized architecture for small objects. *Multimedia Tools and Applications*, 1-27.

Froidevaux, A., Julier, A., Lifschitz, A., Pham, M. T., Dambreville, R., Lefèvre, S., ... & Huynh, T. L. (2020). Vehicle detection and counting from VHR satellite images: efforts and open issues. In *IGARSS 2020-2020 IEEE International Geoscience and Remote Sensing Symposium* (pp. 256-259). IEEE.

Ghosh, A., Sabuj, M. S., Sonet, H. H., Shatabda, S., & Farid, D. M. (2019, June). An adaptive video-based vehicle detection, classification, counting, and speed-measurement system for real-time traffic data collection. In *2019 IEEE region 10 symposium (TENSYMP)* (pp. 541-546). IEEE.

Gössling, S. (2018). ICT and transport behavior: A conceptual review. *International journal of Sustainable Transportation*, *12*(3), 153-164.

Hidayati, N., Harimanana, R. R., Sunarjono, S., Dhia, H., & Ghalib, A. (2022). Bottleneck Effect Caused by Motorcyclist Presence in the Traffic Flow on Kerten Signalized Intersection. *Civil Engineering and Architecture*, *10*(6), 2338-2351, https://doi.org/10.13189/cea.2022.100609.

Huang, Z.-J., Patel, B., Lu, W.-H., Yang, T.-Y., Tung, W.-C., Bučinskas, V., Greitans, M., Wu, Y.-W., & Lin, P. T. (2023). Yeast cell detection using fuzzy automatic contrast enhancement (FACE) and you only look once (YOLO). *Scientific Reports*, *13*(1), 16222.

Huu, D. N., & Ngoc, V. N. (2021). Analysis study of current transportation status in Vietnam's urban traffic and the transition to electric two-wheelers mobility. *Sustainability*, *13*(10), 5577.

Jutury, D., Kumar, N., Sachan, A., Daultani, Y., & Dhakad, N. (2023). Adaptive neuro-fuzzy enabled multi-mode traffic light control system for urban transport network. *Applied Intelligence*, *53*(6), 7132-7153.

Kamal, M. A. S., Hayakawa, T., & Imura, J.-i. (2019). Development and evaluation of an adaptive traffic signal control scheme under a mixed-automated traffic scenario. *IEEE transactions on intelligent transportation systems*, *21*(2), 590-602.

Khairudin, M., Refalda, R., Yatmono, S., Pramono, H., Triatmaja, A., & Shah, A. (2020). The mobile robot control in obstacle avoidance using fuzzy logic controller. *Indonesian Journal of Science and Technology*, *5*(3), 334-351.

Khansari, N., Mostashari, A., & Mansouri, M. (2014). Impacting sustainable behavior and planning in smart city. *International Journal of Sustainable Land Use and Urban Planning*, *1*(2), 46-61.

Lin, C.-J., & Jhang, J.-Y. (2022). Intelligent traffic-monitoring system based on YOLO and convolutional fuzzy neural networks. *IEEE Access*, *10*, 14120-14133.

Lootsma, F. A. (2013). *Fuzzy logic for planning and decision making* (Vol. 8). Springer Science & Business Media.

Makino, H., Tamada, K., Sakai, K., & Kamijo, S. (2018). Solutions for urban traffic issues by ITS technologies. *IATSS Research*, *42*(2), 49-60.

Moranduzzo, T., & Melgani, F. (2013). Automatic car counting method for unmanned aerial vehicle images. *IEEE Transactions on Geoscience and Remote Sensing*, *52*(3), 1635-1647.

Namazi, E., Li, J., & Lu, C. (2019). Intelligent intersection management systems considering autonomous vehicles: A systematic literature review. *IEEE Access*, *7*, 91946-91965.

Ngon, N. C. (2010). Research on smart traffic light system design. *Can Tho University Journal of Science*, (15b), 56-63 (in Vietnamese).

Olayode, O., Tartibu, L., & Okwu, M. (2020). Application of artificial intelligence in traffic control system of non-autonomous vehicles at signalized road intersection. *Procedia CIRP*, *91*, 194-200.

Qadri, S. S. S. M., Gökçe, M. A., & Öner, E. (2020). State-of-art review of traffic signal control methods: challenges and opportunities. *European Transport Research Review*, *12*, 1-23.

Robinson, R., & Thagesen, B. (2018). *Road engineering for development*. CRC Press.

Shamsuzzoha, A., Nieminen, J., Piya, S., & Rutledge, K. (2021). Smart city for sustainable environment: A comparison of participatory strategies from Helsinki, Singapore and London. *Cities*, *114*, 103194.

Sweet, M. (2014). Traffic congestion's economic impacts: Evidence from US metropolitan regions. *Urban Studies*, *51*(10), 2088-2110.

Talaat, F. M., & ZainEldin, H. (2023). An improved fire detection approach based on YOLO-v8 for smart cities. *Neural Computing and Applications*, *35*(28), 20939-20954.

Terven, J., Córdova-Esparza, D.-M., & Romero-González, J.-A. (2023). A Comprehensive Review of YOLO Architectures in Computer Vision: From YOLOv1 to YOLOv8 and YOLO-NAS. *Machine Learning and Knowledge Extraction*, *5*(4), 1680-1716.

Terven, J., & Cordova-Esparza, D. (2023). A comprehensive review of YOLO: From YOLOv1 to YOLOv8 and beyond. *arXiv preprint arXiv:2304.00501*.

Van Dender, K. (2019). Taxing vehicles, fuels, and road use: Opportunities for improving transport tax practice.

Wang, X., Gao, H., Jia, Z., & Li, Z. (2023). BL-YOLOv8: An Improved Road Defect Detection Model Based on YOLOv8. *Sensors*, *23*(20), 8361.

Wath, A. K., Daigavane, P. M., & Naidu, H. K. (2022, December). A technique of image and fuzzy logic to solve traffic congestion at signal. In *2022 International Conference on Smart Generation Computing, Communication and Networking (SMART GENCON)* (pp. 1-6). IEEE.

Wensveen, J. (2023). *Air transportation: A global management perspective*. Routledge.

Xiang, X., Zhai, M., Lv, N., & El Saddik, A. (2018). Vehicle counting based on vehicle detection and tracking from aerial videos. *Sensors*, *18*(8), 2560.

Yang, H., & Qu, S. (2018). Real-time vehicle detection and counting in complex traffic scenes using background subtraction model with low-rank decomposition. *IET Intelligent Transport Systems*, *12*(1), 75-85.

Yazdinejad, A., Dehghantanha, A., Parizi, R. M., Srivastava, G., & Karimipour, H. (2023). Secure intelligent fuzzy blockchain framework: Effective threat detection in IoT networks. *Computers in Industry*, *144*, 103801.

Yousif, M. T., Sadullah, A. F. M., & Kassim, K. A. A. (2020). A review of behavioural issues contribution to motorcycle safety. *IATSS Research*, *44*(2), 142-154.

Zadeh, L. A. (2023). Fuzzy logic. In *Granular, Fuzzy, and Soft Computing* (pp. 19-49). Springer.

Zangeneh, M., Aghajari, E., & Forouzanfar, M. (2022). A review on optimization of fuzzy controller parameters in robotic applications. *IETE Journal of Research*, *68*(6), 4150-4159.