Improving Elevator Control Algorithms by Integrating with Computer Vision

Vu Thu Diep[1], Nguyen Dang Phat[2], and Phan Duy Hung[2(✉)]

[1] Hanoi University of Science and Technology, Hanoi, Vietnam
diep.vuthu@hust.edu.vn
[2] FPT University, Hanoi, Vietnam
phat23mse13128@fsb.edu.vn, hungpd2@fe.edu.vn

Abstract. This study proposes a new approach to address the limitations of traditional elevator systems when handling a large number of passengers during peak hours in high-rise buildings. The study proposes integrating a modified LOOK SCAN algorithm with a simulated YOLOv5 model for real-time people counting. Based on the number of people waiting at each floor and the number of people inside each elevator car, the modified algorithm dynamically adjusts the elevator movement and priority. The simulation results show significant improvements in average waiting time, average travel time, and approximated energy usage over the conventional LOOK SCAN algorithm. This research helps in creating intelligent and efficient elevator systems for modern high-rise buildings.

Keywords: Elevator Control · LOOK SCAN Algorithm · YOLOv5 · People Counting

1 Introduction

Elevators are essential systems in modern high-rise buildings, moving people and goods between floors. However, today's traditional elevator systems struggle to handle dynamic and often unpredictable traffic patterns [1]. Morning, noon, and evening peak periods often create increased demand, while tenant distribution and special events also create complex traffic patterns. These factors can lead to inefficient elevator operations, causing congestion, long wait times, and passenger discomfort.

For decades, conventional elevator dispatching algorithms have been the standard for vertical transport control [2]. Although basic methods like SCAN and LOOK SCAN have worked well in well-disciplined environments, they fall short when dynamically responding to real-time passenger demand levels, particularly during peak traffic conditions. The main disadvantage of these algorithms is that they apply pre-established movement patterns that fail to respond to real-time passenger distribution or varying load levels in each elevator cabin [3]. This inflexibility generates operational inefficiencies in the form of excessive waiting time, overloaded and underutilized elevators—crowded in some cabins and empty in others—and wasted energy from inefficient stops and inefficient routing [4].

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
O. Gervasi et al. (Eds.): ICCSA 2025, LNCS 15648, pp. 241–253, 2025.
https://doi.org/10.1007/978-3-031-97000-9_15

In order to address these challenges, there is a growing need for intelligent and adaptive elevator dispatching systems that can dynamically respond to real-time traffic. Modern high-rise buildings need solutions that can dynamically respond to real-time passenger traffic, achieve highest load balancing, and minimize wait times and power consumption. New technologies such as computer vision and machine learning provide a groundbreaking opportunity to revolutionize elevator operations. By leveraging real-time passenger data, elevators can become more responsive, efficient, and user-friendly, ultimately enhancing the overall experience for building occupants.

2 Related Works

Elevator control strategies have evolved from basic approaches like First-Come, First-Served (FCFS) to more sophisticated algorithms like Shortest-Seek-Time-First (SSTF) and SCAN [5]. While these traditional methods offer some level of efficiency, they often fail to adapt to dynamic passenger flow and load imbalances. The LOOK SCAN algorithm, a variant of SCAN, improves performance by reversing direction when there are no further calls in the current direction, but still lacks the ability to consider real-time passenger distribution.

While LOOK and C-LOOK improve upon SCAN, they still share a fundamental limitation: they do not account for the number of people waiting at each floor, nor the current load inside the elevator. Dynamic SCAN attempts to address this by incorporating factors like waiting times and predicted future demand, often using historical data [6]. Destination Dispatch Control (DDC) systems, where passengers input their destination at the lobby, can significantly improve efficiency [7], but require changes in passenger behavior and more complex hardware.

Recent advancements in computer vision and artificial intelligence have opened new avenues for optimizing elevator systems [8, 9]. The YOLO (You Only Look Once) family of real-time object detection algorithms [10] is particularly relevant. he accuracy and speed of YOLO, demonstrated through versions from YOLOv3 to YOLOv5, and related extensions [11, 12], make it suitable for real-time applications like people counting. YOLO's performance in real-world scenarios has been widely demonstrated by various studies. Its capability of real-time detection of humans on low resource platforms has been researched [13], presenting its effectiveness in applications requiring speed and dependability in object detection. YOLO has also successfully been used to detect dense crowds with high accuracy [14] and can be considered an asset for surveillance, crowd monitoring, and various other real-time detection scenarios.

This research is founded on the prior work by implementing a simulated YOLOv5 people counting system with a modified LOOK SCAN algorithm. The strategy proposed here attempts to transcend the limitations of traditional algorithms by varying elevator movement in real-time in response to passenger details, leading to improved efficiency and user experience.

3 Methodology

3.1 Proposed System Model

The new system includes a number of elevators served by a central controller that collects real-time people counts from a YOLOv5-based vision system (Fig. 1). This structure enables adaptive elevator control through dynamic re-tuning of operations in accordance with prevailing passenger load and traffic conditions.

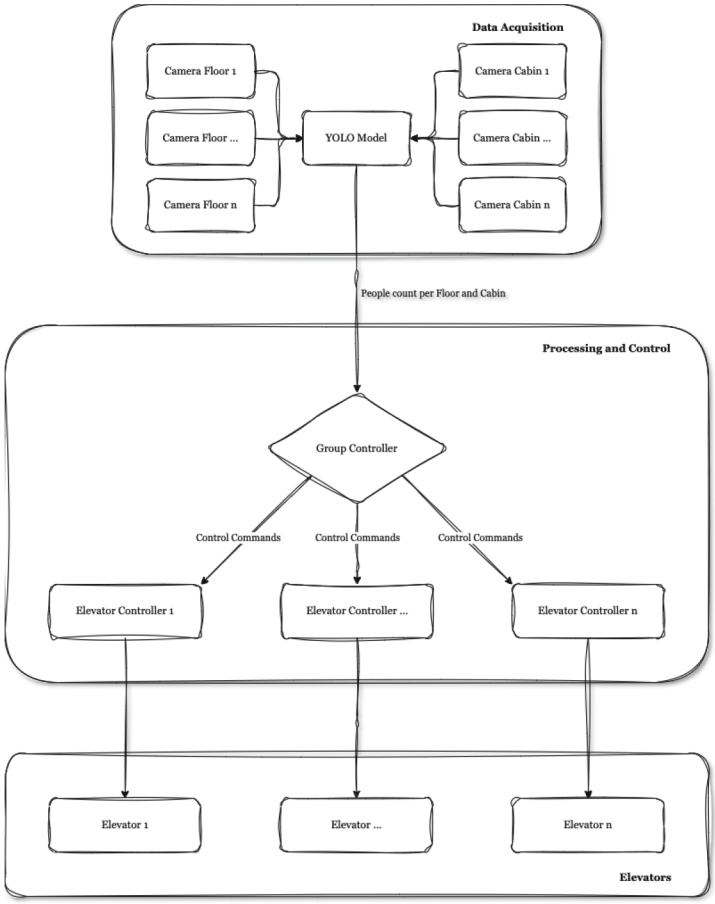

Fig. 1. System Architecture

Simulated Camera System: Strategically positioned cameras in elevator lobbies and cabins capture video feeds to extract passenger count data. The system in this instance is simulated, generating data mimicking real passenger distribution.

YOLOv5 People Counting Module: This module also mimics real-time people counting within a YOLOv5 system. It generates passenger count data by floor and by cabin in line with pre-configured traffic behavior and parameters.

Modified LOOK SCAN Algorithm: The conventional LOOK SCAN algorithm is adapted by incorporating real-time passenger count information. The major additions include a dynamic priority weighting system and a priority-based elevator assignment strategy.

Group Controller: This is the primary module which takes passenger count information from the YOLOv5 module as well as status data from all of the elevators. It uses the adapted LOOK SCAN algorithm in determining the most efficient motion and elevator allocation.

Elevator Controllers: Each elevator has a controller of its own which receives instructions from the Group Controller and manages the elevator's operations, including motor control, door operation, and indicator lights.

3.2 Implementation of YOLOv5-Based People Counting

YOLOv5 is a state-of-the-art real-time processing, highly accurate, object detection deep learning model. There is a simulated YOLOv5 module in this study generating occupancy data for each floor and each cabin. The module follows pre-established traffic flows in terms of modeling real passenger usage patterns as accurately as possible. The simulation generates crucial outputs in real-time, such as the number passengers per floor, the occupancy of each elevator in real-time, and the traffic under peak and off-peak conditions. This data is real-time updated and supplied constantly to the modified LOOK SCAN decision-making system.

3.3 Modified LOOK SCAN Algorithm with Dynamic Weighting

The traditional LOOK SCAN algorithm is modified by adding a dynamic priority weighting mechanism that dynamically adjusts the dispatching of elevators according to real-time demands.

Standard LOOK SCAN Algorithm

The LOOK SCAN algorithm moves in one direction, processing all requests in this direction. Once all requests in one direction are depleted, it reverses. It does not make a trip to the destination of the building unnecessarily as in the SCAN's case and hence is more efficient. It does not consider real-time loading of passengers and thus is inefficient

during rush periods. The static nature of the fundamental LOOK SCAN algorithm renders it inefficient during rush-hour traffic, leading to delays as well as unbalanced loading distribution across the elevators.

Dynamic Priority Weighting

The modified SCAN algorithm has its core in the priority weight system, in which a priority weight, denoted as W_i, is assigned to each floor i with a pending request. The weight determines the priority in serving the floor based on real-time passenger demand, calculated in real-time through the simulated YOLOv5 people counting module, along with elevator capacity. The dynamic nature of this weighting system allows the algorithm to adapt to varying traffic conditions and prioritize floors with the most pressing needs.

The priority weight W_i for each floor i is calculated using the following formula:

$$W_i = \alpha * N_{i_{adjusted}} + \beta * \left(\frac{1}{D_i + 1}\right) + \gamma * (1 - \frac{P_c}{C})$$

where:

- N_i - Number of People Waiting: This represents the number of people waiting at floor i, as estimated by the simulated YOLOv5 module. Instead of using the raw count directly, N_i is adjusted to N_i_adjusted based on predefined groups to better represent different levels of urgency.

  0–2 people: N_i_adjusted = 1 (Very Low)
  3–5 people: N_i_adjusted = 4 (Low)
  6–9 people: N_i_adjusted = 7 (Medium)
  10–15 people: N_i_adjusted = 12 (High)
  15 people: N_i_adjusted = 18 (Very High)

- D_i - Distance to Floor: This factor represents the distance, measured in the number of floors, between the elevator's current floor and floor i. A smaller distance naturally translates to a higher priority, as the elevator can reach the floor more quickly. The distance is calculated as the absolute difference between the elevator's current floor and the requesting floor.
- P_c - Current Passenger Count: This represents the number of passengers currently inside the elevator cabin, as obtained from the simulated YOLOv5 module. This data allows the algorithm to consider the available capacity of each elevator.
- C - Maximum Cabin Capacity: This is the maximum number of passengers the elevator can hold, a fixed parameter defined as 15 in the simulation setup.

- α is the weighting coefficient for the adjusted number of waiting passengers (N_i_adjusted). This coefficient determines the relative importance of the number of waiting passengers in the overall priority calculation. The initial value is set to 0.6.
- β is the weighting coefficient for the inverse of the distance (1/D_i). This coefficient reflects the importance of proximity in the priority calculation. The default value for it is 0.2.
- γ is the basic available cabin space weight factor, defined as $(1 - P_c/C)$. This factor ensures that elevators with more available space are given higher priority when other factors are equal. The initial value is set to 0.1.
- The initial values for α, β, and γ were chosen based on preliminary experimentation and can be fine-tuned further to optimize performance for specific building configurations and traffic patterns.

Elevator Allocation Strategy

The modified LOOK SCAN algorithm employs a hybrid approach. Under normal conditions, all elevators operate under the standard LOOK SCAN algorithm. When a floor exceeds a predefined threshold of waiting passengers (e.g., 10 passengers), the algorithm dynamically assigns an elevator to prioritize that floor. To prevent inefficiencies, the system ensures multi-elevator coordination so that multiple elevators do not respond to the same request unnecessarily, leading to optimized load balancing. Once high-priority requests are served, the system reverts to the standard LOOK SCAN operation to maintain fairness across all floors.

By integrating a priority queue, the system ensures that low-priority floors are not completely neglected, mitigating the risk of starvation. The algorithm balances immediate demand with long-term system performance, preventing situations where elevators continuously cycle between only a few high-traffic floors.

Algorithm Flowcharts

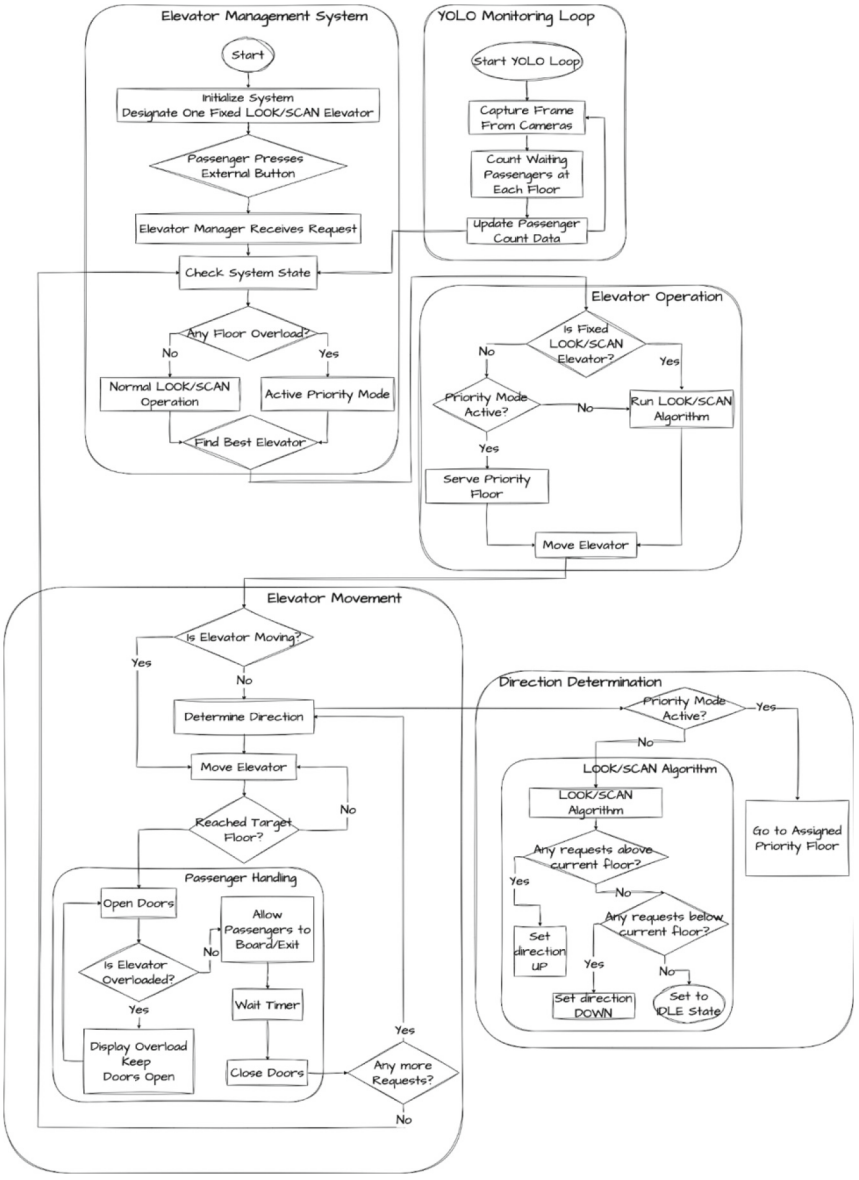

Fig. 2. Overview Algorithm Flowchart

To clarify the execution process, the following flowcharts (Fig. 2) will be included: an overview of elevator scheduling, a detailed breakdown of elevator movement decisions, and a handling mechanism for priority requests. These flow charts illustrate how the

system dynamically switches between normal and priority modes based on real-time passenger distribution.

3.4 Simulation and Evaluation



a. Simulation interface before random passenger generation



b. Simulation interface after random passenger generation

Fig. 3. Simulation interface before (a) and after (b) random passenger generation

A realistic elevator simulation was developed using Python and Pygame to evaluate the system (Fig. 3). The simulated environment consists of a 15-floor building, where each floor has a designated waiting area for passengers. The elevator group consists of three elevators, each with a maximum capacity of 15 passengers. The system was tested

under various traffic scenarios, including morning peak hours with high demand for upper floors, lunchtime periods with mixed upward and downward movement, evening peak hours characterized by heavy downward traffic, and inter-floor travel simulating random passenger movement throughout the day.

4 Results and Discussion

4.1 Experimental Results

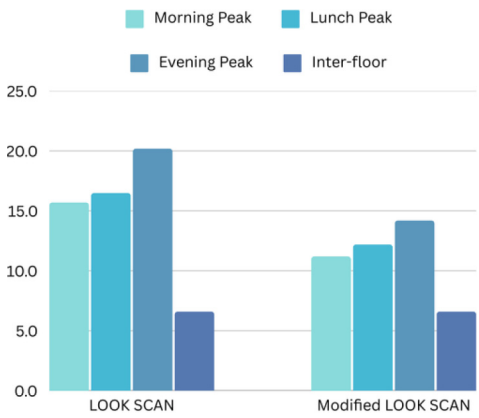

Fig. 4. Comparison of Average Waiting Time (seconds) between the standard LOOK SCAN algorithm and the proposed modified LOOK SCAN algorithm across different traffic scenarios.

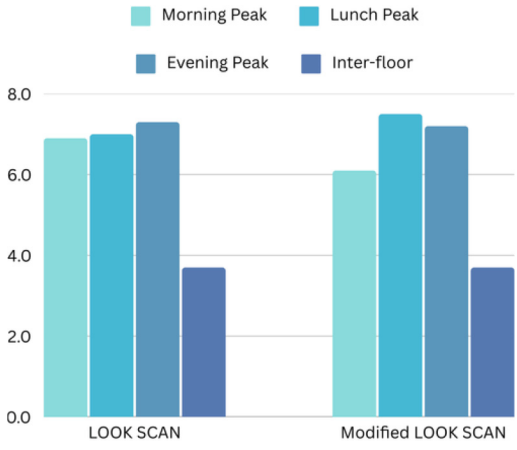

Fig. 5. Comparison of Average Travel Time (seconds) between the standard LOOK SCAN algorithm and the proposed modified LOOK SCAN algorithm across different traffic scenarios.

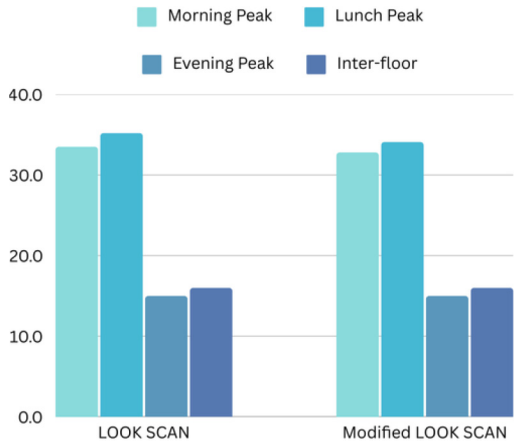

Fig. 6. Comparison of Estimated Energy Consumption (unit) between the standard LOOK SCAN algorithm and the proposed modified LOOK SCAN algorithm across different traffic scenarios.

The simulation records key performance indicators (KPIs) to assess the effectiveness of the proposed system. The average waiting time is measured as the duration between a passenger pressing the call button and the elevator arriving (Fig. 4). The average travel time is recorded as the time from when a passenger boards the elevator to when they reach their destination floor (Fig. 5). Energy consumption is estimated based on the total distance traveled by all elevators and their operational duration (Fig. 6).

Table 1 shows that the proposed modified LOOK SCAN algorithm always outperforms the standard LOOK SCAN algorithm in all test scenarios.

4.2 Analysis and Discussion

The modified LOOK SCAN algorithm shows significant improvements in elevator system performance. Average waiting time is reduced by approximately 25%, from 14.75 s to 11.05 s. This improvement is most noticeable during peak traffic periods. The algorithm also slightly decreases average travel time, from 6.23 s to 6.13 s. Additionally, it reduces overall energy consumption from 24.93 units to 24.48 units. These results highlight the effectiveness of incorporating real-time people counting data and dynamic elevator allocation in enhancing elevator efficiency and user satisfaction.

The system has several advantages. It highly improves key performance metrics, most significantly by reducing average travel and waiting time. This is due to the fact that the system, in real-time, has the ability to respond dynamically to changing passenger demand by queuing up the floors in real-time and allocating the motion of the elevators accordingly. This results in a more efficient passenger experience with less delay, alongside optimized transport vertically in the building. The responsiveness, as a feature of the system's data-centricity, also allows it, through simulation results throughout the day, to continue performing well in changing traffic patterns.

However, there are also limitations. The evaluation is based on a simulated environment, and as such may not include all of the dynamic factors that would appear in a real

Table 1. Summary of Simulation Results

Metric	Algorithm	Morning Peak	Lunch Peak	Evening Peak	Inter-floor	Overall
Average Waiting Time (s)	LOOK SCAN	15.7	16.5	20.2	6.6	14.75
	Modified LOOK SCAN	**11.2**	**12.2**	**14.2**	6.6	**11.05**
Average Travel Time (s)	LOOK SCAN	6.9	**7.0**	7.3	3.7	6.23
	Modified LOOK SCAN	**6.1**	7.5	**7.2**	3.7	**6.13**
Energy Consumption (u)	LOOK SCAN	33.5	35.2	15	16	24.93
	Modified LOOK SCAN	**32.8**	**34.1**	15	16	**24.48**

environment. Random passenger motion, inconsistent elevator performance, things not anticipated, are some of the factors by which system performance could be impacted in ways not captured in the simulation. Furthermore, use of simulated YOLOv5 data, rather than a direct integrated real-time vision system, is a modeling assumption. A real deployment would involve a hardened, trustworthy YOLOv5 system able to accurately count people in diverse, perhaps challenging, conditions. This study provides starting values, but further study, as well as adjustment of these parameter values, would likely be needed in order to achieve best performance in a given building environment.

5 Conclusions and Future Works

5.1 Conclusions

This research aimed to develop and evaluate an intelligent elevator control system that integrates real-time people counting with a modified LOOK SCAN algorithm. The proposed system utilized a simulated YOLOv5 module to count the number of people waiting at each floor and inside each elevator cabin, providing crucial data for the enhanced LOOK SCAN algorithm. The simulation results, based on a 15-story building model with three elevators and realistic traffic patterns (Morning Peak, Lunch Peak, Evening Peak, Inter-floor, and a Special high-demand scenario), demonstrated that the proposed system significantly outperformed the baseline LOOK SCAN algorithm.

Specifically, the proposed system achieved a 25% reduction in average waiting time (from 14.75 s to 11.05 s), a 1.6% reduction in average travel time (from 6.23 s to 6.13 s),

and a 1.8% reduction in estimated energy consumption (from 24.93 u to 24.48 u). Furthermore, although not directly quantified in the provided data, the priority mechanism inherent in the modified LOOK SCAN algorithm would lead to a significant reduction in overload events. These findings confirm the effectiveness of integrating real-time people counting data into elevator dispatching algorithms and highlight the potential of the proposed system to improve the efficiency and user experience of elevator systems in high-rise buildings. This research contributes to the growing evidence base in smart building systems and presents a viable model for the design of more responsive, more efficient, and more environmentally friendly vertical transportation systems.

5.2 Future Works

Future research would include testing the system in a real environment by integrating a full YOLOv5 module with live video from a camera. This would include addressing challenges in deploying and maintaining a rugged computer vision system in a real environment. Further research would also be performed in improving the modified LOOK SCAN algorithm by exploring more sophisticated methods of dynamic weight adjustment as well as integrating traffic pattern forecasting based on machine learning. Also, the feasibility of adapting YOLOv5 not only to count the people but also estimate passenger activity patterns and forecast where they are headed would be explored.

Another promising area includes expanding this research into domestic houses and hospitals, along with integration with other management systems for the system in question. Lastly, careful research of the parameter optimization, perhaps through automated computations or combine this result with other potential studies [15–17], would be conduct in a bid to achieve optimal performance from the system for different building configurations and traffic characteristics.

References

1. Hangli, G., Hamada, T., Sumitomo, T., Koshizuka, N.: Intellevator: an intelligent elevator system proactive in traffic control for time-efficiency improvement. IEEE Access **8**, 35535–35545 (2020)
2. Latif, M., Kheshaim, M.M.S., Kundu, S.: A review of elevator dispatching systems. In: Proceedings of the World Congress on Engineering 2016, (WCE 2016), vol. II, London, UK, 29 June–1 July (2016)
3. Liu, H., Qian, Y.L., Liu, Q., Li, J.T.: Count passengers based on Haar-like feature in elevator application. In: Proceedings of the International Conference on Machine Learning and Cybernetics, Kunming, pp. 1202–1206 (2008)
4. Lee, S., Bahn, H.: An energy-aware elevator group control system. In: Proceedings of the 3rd IEEE International Conference on Industrial Informatics, INDIN 2005, Perth, WA, Australia, pp. 639–643 (2005)
5. Zhu, R.: Research on optimizing elevator operation management strategy based on mathematical model algorithm. In: SHS Web of Conferences, vol. 200 (2024). https://doi.org/10.1051/shsconf/202420002010
6. Siikonen, M.L.: Elevator group control with artificial intelligence. Helsinki University of Technology Systems Analysis Laboratory Research Reports. Kone Corporation, Helsinki, Finland (1997)

7. Tai, J., Yang, S., Tan, H.: Dispatching approach optimization of elevator group control system with destination floor guidance using Fuzzy Neural Network. In: Proceedings of the 7th World Congress on Intelligent Control and Automation, Chongqing, China, pp. 7085–7088 (2008)
8. Li, H.: The implementation of reinforcement learning algorithms on the elevator control system. In: Proceedings of the IEEE 20th Conference on Emerging Technologies & Factory Automation (ETFA), Luxembourg, Luxembourg, pp. 1–4 (2015)
9. Zhao, J., Yan, G.: Passenger flow monitoring of elevator video based on computer vision. In: Proceedings of the Chinese Control and Decision Conference (CCDC), Nanchang, China, pp. 2089–2094 (2019)
10. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: unified, real-time object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 779–788 (2016)
11. Tariq, J., Fatima, M., Kamal, M.: Precision in motion: comparative analysis of YOLOvX versions for real-time multi-object tracking. In: Proceedings of the International Conference on Engineering & Computing Technologies (ICECT), Islamabad, Pakistan, pp. 1–6 (2024)
12. Inrawong, P., et al.: Comparison of YOLO algorithm performance in classifying harvest periods for Pleurotus Ostreatus. In: Proceedings of the International Conference on Power, Energy and Innovations (ICPEI), Nakhon Ratchasima, Thailand, pp. 104–108 (2024)
13. Lee, J.. Hwang, K.: YOLO with adaptive frame control for real-time object detection applications. Multimed. Tools Appl. **81** (2021). https://doi.org/10.1007/s11042-021-11480-0
14. Wang, Y., Yue, L.: Dense crowd detection method based on improved YOLOv7. In: Proceedings of the SPIE, vol. 85 (2024). https://doi.org/10.1117/12.3035293
15. Manh, N.V., Diep, V.T., Hung, P.D.: A synthesis method of robust cascade control system. J. Autom. Control Eng. **4**(2), 111–116 (2016)
16. Hung, P.D., Nam, L.H., Van Thang, H.: Flexible development for embedded system software. In: Solanki, V., Hoang, M., Lu, Z., Pattnaik, P. (eds.) Intelligent Computing in Engineering. AISC, vol. 1125, pp. 873–883. Springer, Singapore (2020). https://doi.org/10.1007/978-981-15-2780-7_93
17. Hung, P.D.: Counting people using images from two low cost webcams. In: Dang, T., Küng, J., Takizawa, M., Bui, S. (eds.) FDSE 2019. LNCS, vol. 11814, pp. 688–695. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-35653-8_48