Thanh-Lam BUI [0000-0003-2859-6670]*, *Ngoc-Tien TRAN* [0000-0001-5099-3758]*

NAVIGATION STRATEGY FOR MOBILE ROBOT BASED ON COMPUTER VISION AND YOLOV5 NETWORK IN THE UNKNOWN ENVIRONMENT

Abstract

The capacity to navigate effectively in complex environments is a crucial prerequisite for mobile robots. In this study, the YOLOv5 model is utilized to identify objects to aid the mobile robot in determining movement conditions. However, the limitation of deep learning models being trained on insufficient data, leading to inaccurate recognition in unforeseen scenarios, is addressed by introducing an innovative computer vision technology that detects lanes in real-time. Combining the deep learning model with computer vision technology, the robot can identify different types of objects, allowing it to estimate distance and adjust speed accordingly. Additionally, the paper investigates the recognition reliability in varying light intensities. When the light illumination increases from 300 lux to 1000 lux, the reliability of the recognition model on different objects also improves, from about 75% to 98%, respectively. The findings of this study offer promising directions for future breakthroughs in mobile robot navigation.

1. INTRODUCTION

A mobile robot consists of a chassis, or body, that houses the robot's hardware components. The chassis is typically designed to be sturdy, lightweight, and capable of accommodating various sensors and actuators. In addition, the software of a mobile robot encompasses the algorithms, control systems, and operating systems that enable it to function. The cameras also play a very important role; they provide visual perception and enable the robot to gather information about its environment. Thus, vision-based mobile robot navigation includes the design of robots, cameras, object recognition algorithms, and control systems. Intelligent mobile robots must possess the ability to navigate in complex environments. The field of mobile robot navigation is continuously evolving, with various technologies being developed. The development of intelligent robots capable of autonomous navigation in unknown environments is the primary focus of mobile robot navigation. It is a multidisciplinary field that combines techniques and technologies from robotics (Hichri et al., 2021; Nguyen & Vu, 2022; Thuan et al., 2023), computer vision (Marroquín et al., 2023), machine learning (Geng et al., 2018; Xiao et al., 2022), and control theory (Ren et al., 2022; Shang et al., 2021). The ultimate objective of mobile robot navigation research

---

* Hanoi University of Industry, Faculty of Mechanical Engineering, Department of Mechatronics Engineering, Vietnam, tientn@haui.edu.vn

is to create automated systems that can operate safely and efficiently in real-world environments, performing tasks such as exploration, surveillance, delivery, and transportation. This field has gained significant attention from researchers in recent years, with numerous studies focused on advancing mobile robot navigation techniques. A novel navigation model was proposed in the article, which not only establishes two-way communication between the cerebellum and basal ganglia but also allows for their co-development (Wang et al., 2020). This approach enabled the agent to autonomously develop its intelligence through hybrid learning techniques. An artificial potential field (APF) with brain signals through a brain-robot interface (BRI)-based control strategy, in conjunction with simultaneous localization and mapping (SLAM) techniques was developed in the study (Liu et al., 2020). This combination enabled the establishment of a relationship between the strength of EEG signals and the intensity of the potential field, facilitating the navigation and control of a mobile robot in uncertain environments. The framework for self-improving lifelong learning designed for a mobile robot was introduced in the research (Liu et al., 2021). This method could enable agents to operate in various environments. The approach for guiding mobile robots was proposed by (Ajeil et al., 2020). The core of this approach was based on hybridizing the Particle Swarm Optimization with the Modified Frequency Bat algorithm. The researchers in (Iqbal et al., 2020) discussed a mobile field robot that operates on a Robotic Operating System (ROS) and is capable of simultaneously navigating through occluded crop rows while performing various phenotyping tasks. The authors in (Gharajeh et al., 2020) proposed a method for collision-free navigation of autonomous mobile robots using a hybrid GPS-ANFIS approach. In the study (Ran et al., 2021), the authors developed a Convolutional Neural Network that exhibited superior scene classification accuracy and efficiency in processing monocular images. The research from Lagaza et al. (Lagaza, Kashyap, & Pandey, 2020) has identified navigation algorithms that can efficiently address path optimization challenges while minimizing the required time. Ehab Al Khatib et al. (Khatib et al., 2020) have identified navigation algorithms that can efficiently address path optimization challenges while minimizing the required time. A novel approach integrated with a knowledge-based neural fuzzy controller (KNFC) was proposed to control the navigation of mobile robots in the study (Chen et al., 2021). The authors in (Chen et al., 2021) proposed a multivariable, event-triggered, generalized super-twisting sliding-mode algorithm for the safe navigation of nonholonomic mobile robots in unknown indoor environments. The authors in (Wang et al., 2021) focused on the autonomous navigation of a wheeled mobile robot in a dynamic environment, utilizing a 3D point cloud map and the Creative Mechanism Design Methodology.

Inspired by the research mentioned, this paper proposes a detection framework for mobile robot navigation based on computer vision and deep learning. To be more detailed, a segmentation algorithm is designed to extract the information of movement lane for the robot. After that, the YOLOv5 network (Luo et. al, 2021) is built up and integrated automatically on the robot controller. Hence, the robot could detect and cluster the necessary information for navigation process. The experimental results indicate the working performance of the proposed method under different circumstances.

The rest of the content of this paper is included as follows: The next section illustrates the detailed structure of the proposed approach. In the third section, the experimental results are conducted and analyzed via different scenarios. Finally, the conclusion is drawn up in depth in Section 4.

2. THE PROPOSED METHOD

In general, the proposed method is designed based on the following criteria: the ability to identify lanes; the ability to detect different groups of objects and control reactions corresponding to them; and the ability to modify instantaneous speed. The sub-sections below analyze in depth all the mentioned criteria.

2.1. Lane detection algorithm

To determine the movement line for the robot model, a real-time image processing system is designed as shown in Fig. 1. There are six steps in total of this procedure. To begin with, the system takes the images extracted from cameras mounted at the front of the robot model. After that, the system determines the color threshold of the lane and calibrates the precision lane by reducing noise on the binary images. To be more specific, histogram equalization is used to eliminate the outline pixels from the photos, and then a morphological filter is employed for Hole Filling process (Cho et al., 2022). The latency for the computing phase after taking the photo and sending data to the drives is estimated at 1 seconds on average.
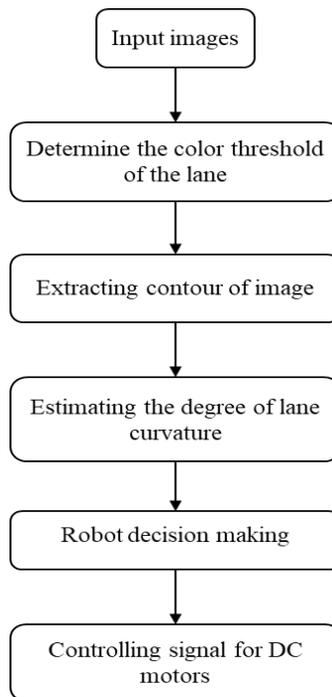
```
┌─────────────────────┐
│    Input images     │
└─────────────────────┘
           │
           ▼
┌─────────────────────┐
│ Determine the color │
│  threshold of the   │
│        lane         │
└─────────────────────┘
           │
           ▼
┌─────────────────────┐
│  Extracting contour │
│      of image       │
└─────────────────────┘
           │
           ▼
┌─────────────────────┐
│ Estimating the      │
│ degree of lane      │
│     curvature       │
└─────────────────────┘
           │
           ▼
┌─────────────────────┐
│  Robot decision     │
│      making         │
└─────────────────────┘
           │
           ▼
┌─────────────────────┐
│ Controlling signal  │
│   for DC motors     │
└─────────────────────┘
```

Fig. 1. The lane detection algorithm

Fig. 2. The pre-processing region of the lane images

As a result, the region for the robot motion is clearly separated from the input images as shown in Fig. 2. In the next step, the contours of the binary images are taken out to get an idea of how much the lanes curve. The average steering angle for the robot is used to figure out the value of the lane curvature. Also, in order to conserve computing resources for the processor, the authors severely reduce the required movement lanes on the image region. While evaluating the road's curve, the algorithm simply examines the curvature immediately in front of the robot model. So, the binary images are now cropped to a new image and limited by the orange circle dot that can be seen in Fig. 3.

Fig. 3. The principle of extracting images to the processing area

The mesh of lane curvature is established by aiding the method of summing pixels. As indicated in Fig. 4, the lane curvature included the pixels with a value of 255. All pixels of this type on the left and right sides of the y-axis, which are separated by the red line, are summarized. Thus, the yellow boundaries are drawn to display the lane for the robot to move in.

Fig. 4. The lane curvature visualized by pixel values

However, when the robot runs through roadways with varying curves and intricacy, the numbers of pixels to the left and right of the y-axis are different. For instance, in the case of turning left, the pixels on the left are obviously greater than those on the right. The only case where the number of pixels is evenly distributed in both directions is when the robot is running straight.
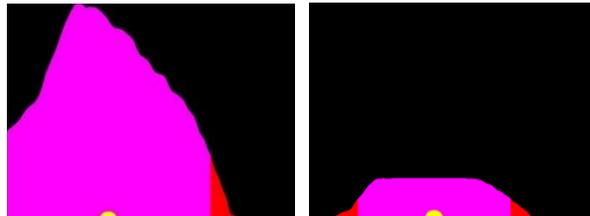

Fig. 5. The lane with yellow dot instructions

For the purpose of ensuring the continuity and the precision of the robot moving decision every time step, the signal for the DC motor is unified and depicted by the yellow point as shown in Fig. 5. As the lane bends to the right, the yellow dot deviates to the right; when going in a straight line, the yellow dot is centered on the screen. The final results of lane determination are presented in Fig. 6.
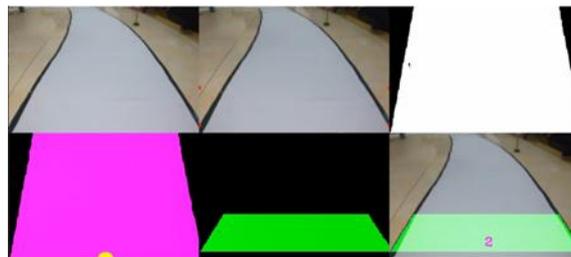
Fig. 6. The final results of lane determination

2.2. Object Detection by Applying YOLOv5 Network

In this study, a YOLOv5 network is designed to detect several types of objects: static objects (traffic lights, speed indicators) and dynamic objects (human target). The detailed structure of the proposed model is illustrated in Fig. 7. There are 1200 images captured in total for the mentioned target groups. These obtained data sets are then imported and split for training the YOLOv5 model at the default ratios: seven parts of the data size for the training phase and three parts of the data size for the evaluation phase.

The architecture of the YOLOv5 network is composed of three main components: the Backbone, the Neck and the Head. To be more specific, the Backbone is combined between the Cross Stage Partial (H. Lin, & J. Yang, 2022) and the Darknet53 network (Wang et al., 2020). The purpose of this combination is for the Cross Stage Partial to address the redundant gradient problem by truncating the gradient flow, while the Darknet53 addresses the vanishing gradient issue. Thus, the number of calculated parameters is decreased, and the inference speed, a key metric in real-time object detection models, is dramatically boosted. The second component of the proposed network, the Neck, is built on the Path Aggregation

Network (PANet) (Zhou et al., 2022) to enhance the information flow and help in the accurate localization of pixels in the process of mask prediction. Hence, the possibility of creating features based on previously unseen data is improved. Finally, the Head is made of the convolutional sub-network to generate predictions from the anchor boxes for the detection process.
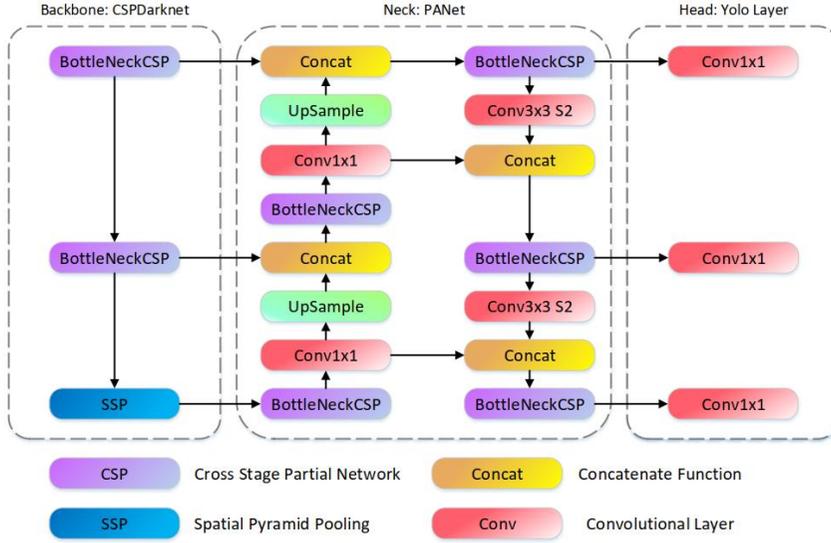

Fig. 7. The architecture of YOLOv5 model (Luo et. al, 2021)

To evaluate the the accuracy of the classified object in the real – time video stream, a synthetic loss function is established. It could be seen that the synthetic loss function is the sum of three kinds of losses: the confidence loss, the localization loss, and the classification loss. The formula of the synthetic loss function is written as follows:

$$\lambda_{Synthetic} = \lambda_{Confidence} + \lambda_{Localization} + \lambda_{Classification} \tag{1}$$

which: $\lambda_{Synthetic}$ is the synthetic loss function; $\lambda_{Confidence}$ is the confidence loss; $\lambda_{Localization}$ is the localization loss; and $\lambda_{Classification}$ is the classification loss.

According to (Hu et al., 2022), the confidence loss, the localization loss, and localization loss are calculated as in Eq. (2) to Eq. (4):

$$\lambda_{Confidence} = \sum_{i=0}^{M^2} \sum_{j=0}^{K} R_{ij}^{obj} (C_i - \hat{C}_i)^2 + \varphi_{coord} \sum_{i=0}^{M^2} \sum_{j=0}^{K} R_{ij}^{obj} (C_i - \hat{C}_i)^2 \tag{2}$$

$$\lambda_{localization} = \varphi_{coord} \sum_{i=0}^{M^2} \sum_{j=0}^{K} \square_{ij}^{obj} \left[\left(x_i - x_l \right)^2 + \left(y_i - y_l \right)^2 \right]$$
$$+ \varphi_{coord} \sum_{i=0}^{M^2} \sum_{j=0}^{K} \left[\left(\sqrt{w_i} - \sqrt{w_l} \right)^2 + \left(\sqrt{h_i} - \sqrt{h_l} \right)^2 \right] \tag{3}$$

$$\lambda_{classfication} = \sum_{i=0}^{M^2} \sum_{c \in classes} \square \, _{ij}^{obj} \left(p_i(c) - p_l(c) \right)^2 \qquad (4)$$

whereas: R_{ij}^{obj} means to judge whether there is an object center in grid mesh of the image. If the grid contains an object center, it is responsible for predicting the category probability of the object, C_i represents the confidence score, $\hat{C}_i$ represents the intersection of the prediction boundary box and the basic facts, and φ_{coord} represents the weight of the classification error.

For the activation stage of the proposed network, the SiLU function (Elfwing et al., 2018) is applied in the hidden layers, while the Sigmoid function (Alexandris et.al, 2019) is integrated in the output layers. To save computational time, the ADAM function (Cao et al., 2023) is applied, which is the default in the original YOLOv5 network.

3. RESULTS AND DISCUSSION

To inspect the stability as well as the working performance of the proposed method, a differential wheeled robot integrated with a smartphone at the front of the model is designed. The completed concept of the model is shown in Fig. 8.

a) The concept of the robot model
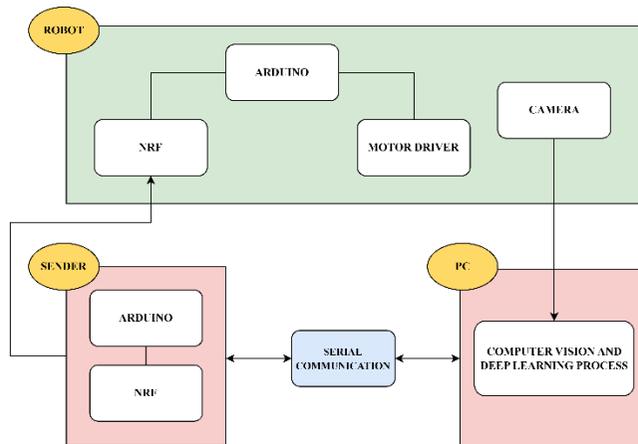
b) Scheme of communication between devices

Fig. 8. The concept of the mobile robot and communication between devices

The study uses an ultrasonic sensor HC-SR04 to estimate the distance from the robot to the object. From there, the system will transmit control signals to navigate the robot in different situations.

After that, the case studies are investigated via several scenarios: human detection, signal traffic classification and navigation based on the line and the types of traffic light.

3.1. Case study 1: Human detection

The ability of the proposed approach to detect several human targets is shown in Fig. 9. It is evident that the model could perform well with the complicated attire of individuals dressed in white or black suits. Besides, the distance to each target is computed and displayed in real-time.

Fig. 9. Human detection of the proposed model

3.2. Case study 2: Signal traffic classification

Fig. 10 indicates the classification possibility of the designed model under different circumstances. To be more detailed, in the case of a stop sign, the robot will begin to detect and decelerate when the estimated distance to the sign reaches 0.5 meters. About 0.2 meters away from the sign, the robot will come to a complete stop. In addition, in the case of speed indicator signs, the robot will compute and update the motor velocity in order to maintain the permitted speed indicated by the signs. If the maximum speed is higher than the current speed, the robot could increase the velocity value. In contrast, the robot will slow down the current speed.

a)	*b)*	*c)*

Fig. 10. Signal traffic classification rate of the designed model under different scenarios: a) stop sign; b) speed indicator sign with 0.2 *m/s*; c) speed indicator sign with 0.5 *m/s*

3.3. Case study 3: Navigation in complicated traffic situations

In case study 3, the hypothetical situation is that the traffic light is green but there are still some people crossing the road. The control commands are executed in order of priority, from human objects to light signals. Hence, the robot has to stop from a safe distance in this situation, as shown in Fig. 11.
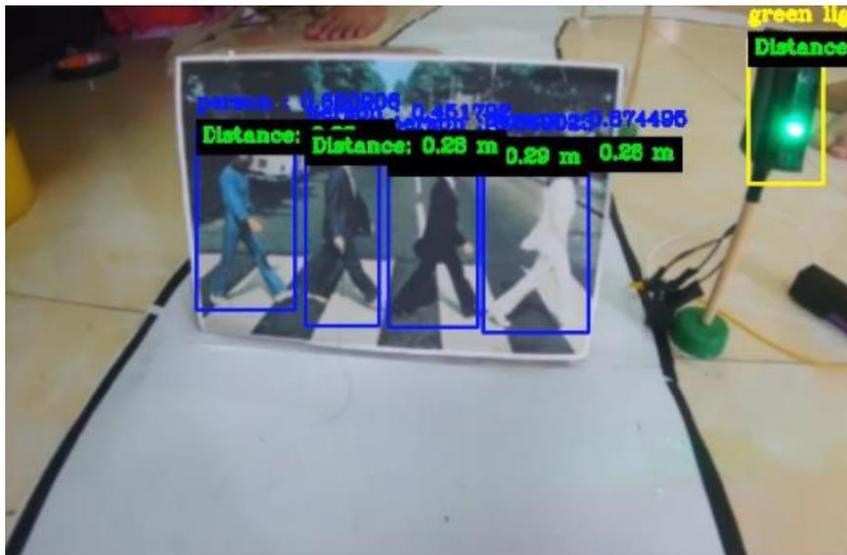

Fig. 11. Navigation for mobile robot in complicated traffic situations

3.4 Analyze the experimental statistics with various levels of illumination

For robots to be useful, they need to be smart enough to see what is going on around them and make decisions based on what they see. Object recognition is one of the critical tasks that robots must perform to function correctly. In this sub-section, all case studies mentioned above are conducted with different rate of illumination variables (Full bright – 1000 lux,

Normal bright – 750 lux, Weak light – 300 lux). The normal speed of the robot is still set at 0.2 m/s and 0.5 m/s, respectively. Tab. 1 to Tab. 6 show the results of detection under different illumination conditions. The x-y figure generated from the tests revealed that the proposed algorithm had a high accuracy rate of over 75% in recognizing objects. The algorithm performed exceptionally well in recognizing traffic lights, with an accuracy rate of between 86.5% and 98.4%. The recognition of human objects, however, had a lower accuracy rate compared to that of traffic lights, with values ranging from 81.8% to 94.1%. The lowest accuracy rate was observed in the recognition of signs, with values ranging from 76.1% to 93%. The results showed that as the light intensity increased, the robot's ability to identify objects improved. However, we observed a stable recognition ability for the robot. Interestingly, the robot was less accurate in recognizing objects when it moved at a speed of 0.5 m/s than when it moved at a speed of 0.2m/s. This finding suggests that the target recognition of the robot model improves when it moves slowly. Additionally, the largest amplitude fluctuation in accuracy was found in the recognition of the stop sign at a light intensity of 1000 lux in both speed scenarios. Furthermore, it is noticed that when the robot encountered the speed indicator 20, it tended to slow down, which led to an improvement in identification accuracy. The results demonstrate that environmental conditions, such as light intensity levels, can impact object recognition ability in general. As the light intensity increased, the robot identification ability improved. This finding is consistent with previous studies on object recognition, where an increase in light intensity improved the recognition ability of algorithms.

Tab. 1. Detection rates under 1000-lux condition at speed 0.5 m/s

Time (s)	Person	Red Light	Green Light	Stop Sign	Speed Indicator 60	Speed Indicator 20
1	89.6	93.8	96.6	88.2	87.7	88.9
2	89.3	92.5	96.7	92.8	88.8	87.3
3	90.1	95.8	96.5	90.7	87	87.3
4	91	97.3	97.1	85.4	87.3	88.6
5	91.5	92.6	95.8	93	88.4	88.8
6	91.7	94.5	94.4	92.5	87.9	88.3
7	89.4	94.6	95.7	84	87.6	87.5
8	91.6	93.3	97.2	92.3	88.1	87.8
9	89.6	93.1	96.1	88.9	87	87.5
10	89.2	96.9	96.4	91.5	87.6	87.1

Tab. 2. Detection rates under 750-lux condition at speed 0.5 m/s

Time (s)	Person	Red Light	Green Light	Stop Sign	Speed Indicator 60	Speed Indicator 20
1	84.5	93.6	90.4	86.5	80.5	81.2
2	84.1	90.6	90.2	87.5	81.4	81.9
3	83.4	91.1	90.5	87.2	81.2	82
4	85.6	93	90	85.9	81.6	81.3
5	84.1	90.6	91.9	85.7	80.4	81
6	84	91.9	91.7	85.5	81.1	81.3
7	85	93	91.1	86.4	81.5	80.7
8	83	93.6	91.4	86.1	80.2	80.1
9	84	93.7	90.9	85.7	81.2	81.3
10	83.8	91.5	91.4	87.6	81.8	81.7

Tab. 3. Detection rates under 300-lux condition at speed 0.5 m/s

Time (s)	Person	Red Light	Green Light	Stop Sign	Speed Indicator 60	Speed Indicator 20
1	81.8	87.1	89.6	77.3	77.6	79.2
2	82.7	90.1	87.7	76.2	78.1	78.6
3	83.2	88.7	88.5	76.1	77.4	76.6
4	83.1	87.6	89.9	77.2	76.3	77.2
5	83.2	90.3	87.5	76.8	79.7	77.7
6	82.8	86.8	88.1	79.1	79.5	76.1
7	82.8	89.4	90.9	78.7	77.8	78.1
8	83.4	86.6	89.5	79.8	79.8	77.3
9	84.1	88.4	89.1	80	79.3	76.8
10	84.1	89.3	86.5	78.3	79.9	76.6

Tab. 4. Detection rates under 1000-lux condition at speed 0.2 m/s

Time (s)	Person	Red Light	Green Light	Stop Sign	Speed Indicator 60	Speed Indicator 20
1	90.9	98.4	94.9	88.2	89.1	89.4
2	91.9	97.4	95.3	92.8	87.5	88.6
3	92.5	93.8	91.8	90.7	87	89.2
4	93.1	95.1	93.1	85.4	89	89.7
5	89	95.5	94.6	93	88.6	88.7
6	91.5	92.7	95.1	92.5	89.7	89.2
7	88.1	92.1	93.7	84	88.1	89.6
8	94.1	94.5	96.7	92.3	88.7	88.6
9	92.9	98.2	94.7	88.9	87.8	89.4
10	92.9	93.1	95.5	91.5	88.9	88.7

Tab. 5. Detection rates under 750-lux condition at speed 0.2 m/s

Time (s)	Person	Red Light	Green Light	Stop Sign	Speed Indicator 60	Speed Indicator 20
1	88.7	93.6	94.6	86.5	86.9	86.1
2	88.4	90.6	95	87.5	85.6	85.6
3	87.5	91	91.9	87.2	88.4	85.8
4	88.1	93	90.1	85.9	85.3	87.2
5	87	90.6	90.3	85.7	85.8	86.1
6	89	91.9	91.4	85.5	85.6	86.2
7	87.1	93	94	86.4	87.8	85.2
8	87.3	93.6	90.1	86.1	86.9	85.4
9	87.4	93.7	92.7	85.7	87.7	86.7
10	88.6	91.5	91.2	87.6	88.2	87.5

Tab. 6. Detection rates under 300-lux condition at speed 0.2 m/s

Time (s)	Person	Red Light	Green Light	Stop Sign	Speed Indicator 60	Speed Indicator 20
1	81.8	87.1	89.6	86.5	82.6	82.4
2	82.7	90.1	87.7	87.5	83.1	82.2
3	83.2	88	88.5	87.2	82.6	82.9
4	83.1	87.6	89.9	85.9	82.2	82
5	83.2	90.3	87.5	85.7	83.1	82.3
6	82.8	86.8	88.1	85.5	82.4	82.9
7	82.8	89.4	90.9	86.4	82.3	83.5
8	83.4	86.6	89.5	86.1	82.7	83.3
9	84	88.4	89.1	85.7	83.9	83.7
10	84	89.3	86.5	87.6	83.7	82.6

From the obtained results, some summaries are drawn up as follows:
- To achieve the path planning, a lane detection method based on binary images is proposed to provide exact trajectories.
- The results indicate that environmental conditions such as light intensity could impact the object recognition of the robot. As the light intensity increased, the recognition ability of the robot model improved. This finding is consistent with previous studies in the same field.
- Moreover, it is noticed that the precision of recognition on multiple targets is better when the robot model moves at a slower speed of 0.2 m/s instead of 0.5 m/s. This implies that in practical scenarios, it may be necessary to slow down the robot to enhance the detection accuracy.
- In the case of complicated traffic situations, it is necessary to design rules of precedence to give appropriate motion behaviors on the mobile robot. Hence, the safety of both the robot and the target could be ensured. This is proven clear in Case Study 3.

4. CONCLUSION

In this paper, a detection framework for mobile robot navigation based on computer vision and deep learning is proposed. This study aims to automatically navigate the mobile robot in complex traffic situations by intentionally combining several algorithms and the YOLOv5 model. The experimental environment is configured with the appropriate conditions to evaluate the functionality of the proposed approach. In conclusion, this study contributes to the understanding of the factors that affect the performance of an object recognition algorithm. Moreover, the proposed framework not only assures obstacle avoidance, but also stabilizes the robot in its desired position and orientation despite slippage. These findings have important implications for the development of robotics and automation technologies in various fields, including manufacturing, transportation, and healthcare. Future research can focus on exploring the impact of other environmental factors, such as temperature, humidity, and noise for mobile robots in real-time applications.

Conflicts of Interest

The authors have no conflicts of interest to declare.

REFERENCES

Ajeil, F. H., Ibraheem, I. K., Azar, A. T., & Humaidi, A. J. (2020). Autonomous navigation and obstacle avoidance of an omnidirectional mobile robot using swarm optimization and sensors deployment. *International Journal of Advanced Robotic Systems, 17(3)*, 1729881420929498. https://doi.org/10.1177/1729881420929498.

Al Khatib, E. I., Jaradat, M. A. K., & Abdel-Hafez, M. F. (2020). Low-cost reduced navigation system for mobile robot in indoor/outdoor environments. *IEEE Access, 8*, 25014-25026. https://doi.org/10.1109/ACCESS.2020.2971169

Amari, S. I. (1993). Backpropagation and stochastic gradient descent method. *Neurocomputing, 5*(4-5), 185-196. https://doi.org/10.1016/0925-2312(93)90006-O

Bui, T. L., Nguyen, T. H., & Nguyen, X. T. (2023). A Controller for Delta Parallel Robot Based on Hedge Algebras Method. *Journal of Robotics*, *2023*, 2271030. https://doi.org/10.1155/2023/2271030

Cao, J., Zhao, D., Tian, Ch., Jin, T., & Song, F. (2023). Adopting improved Adam optimizer to train dendritic neuron model for water quality prediction. *Mathematical Biosciences and Engineering, 20*(5), 9489-9510. https://doi.org/10.3934/mbe.2023417

Chen, Ch. H., Lin, Ch. J., Jeng, S. Y., Lin, H. Y., & Yu, C. Y. (2021). Using ultrasonic sensors and a knowledge-based neural fuzzy controller for mobile robot navigation control. *Electronics, 10*(4), 466. https://doi.org/10.3390/electronics10040466

Cho, J. M., Park, S. Y., & Chien, S. I. (2020). Hole-filling of realsense depth images using a color edge map. *IEEE Access, 8,* 53901-53914. https://doi.org/10.1109/ACCESS.2020.2981378

Elfwing, S., Uchibe, E., & Doya, K. (2018). Sigmoid-weighted linear units for neural network function approximation in reinforcement learning. *Neural Networks, 107*, 3-11. https://doi.org/10.1016/j.neunet.2017.12.012

Geng, H., Liu, H., Wang, B., & Sun, F. (2018). Reinforcement extreme learning machine for mobile robot navigation. In Cao, J., Cambria, E., Lendasse, A., Miche, Y., & Vong, Ch. M. *Proceedings of ELM-2016*, (pp. 61-73). Springer. https://doi.org/10.1007/978-3-319-57421-9_6

Gharajeh, M. S., & Jond, H. B. (2020). Hybrid global positioning system-adaptive neuro-fuzzy inference system based autonomous mobile robot navigation. *Robotics and Autonomous Systems, 134*, 103669. https://doi.org/10.1016/j.robot.2020.103669

Hichri, B., Gallala, A., Giovannini, F., & Kedziora, S. (2022). Mobile robots path planning and mobile multirobots control: A review. *Robotica*, *40*(12), 4257-4270. https://doi.org/10.1017/S0263574722000893

Hu, M., Wei, Y., Li, M., Yao, H., Deng, W., Tong, M., & Liu, Q. (2022). Bimodal learning engagement recognition from videos in the classroom. *Sensors, 22*(16), 5932. https://doi.org/10.3390/s22165932

Iqbal, J., Xu, R., Sun, S., & Li, C. (2020). Simulation of an autonomous mobile robot for LiDAR-based in-field phenotyping and navigation. *Robotics, 9*(2), 46. https://doi.org/10.3390/robotics9020046

Kingma, D. P., & Ba, J. (2014). *Adam: A method for stochastic optimization.* arXiv. https://doi.org/10.48550/arXiv.1412.6980

Lagaza, K. P., Kashyap, A. K., & Pandey, A. (2020). Spider monkey optimization algorithm based collision-free navigation and path optimization for a mobile robot in the static environment. In Biswal, B. B., Sarkar, B. K., & Mahanta, P. *Advances in Mechanical Engineering: Select Proceedings of ICRIDME 2018* (pp. 1459-1473). Springer. https://doi.org/10.1007/978-981-15-0124-1_128

Lin, H., & Yang, J. (2022). Ensemble cross-stage partial attention network for image classification. *IET Image Processing, 16*(1), 102-112. https://doi.org/10.1049/ipr2.12335

Liu, B., Xiao, X., & Stone, P. (2021). A lifelong learning approach to mobile robot navigation. *IEEE Robotics and Automation Letters, 6*(2), 1090-1096. https://doi.org/10.1109/LRA.2021.3056373

Liu, Y., Li, Z., Zhang, T., & Zhao, S. (2020). Brain–robot interface-based navigation control of a mobile robot in corridor environments. *IEEE Transactions on Systems, Man, and Cybernetics: Systems, 50*(8), 3047-3058. https://doi.org/10.1109/TSMC.2018.2833857

Luo, S., Yu, J., Xi, Y., & Liao, X. (2022). Aircraft target detection in remote sensing images based on improved YOLOv5. *IEEE Access, 10*, 5184-5192. https://doi.org/10.1109/ACCESS.2022.3140876

Marroquín, A., Garcia, G., Fabregas, E., Aranda-Escolástico, E., & Farias, G. (2023). Mobile robot navigation based on embedded computer vision. *Mathematics, 11*(11), 2561. https://doi.org/10.3390/math11112561

Mourgias-Alexandris, G., Tsakyridis, A., Passalis, N., Tefas, A., Vyrsokinos, K., & Pleros, N. (2019). An all-optical neuron with sigmoid activation function. *Optics Express, 27*(7), 9620-9630. https://doi.org/10.1364/OE.27.009620

Nguyen, A. T., & Vu, C. T. (2022). Obstacle Avoidance for Autonomous Mobile Robots Based on Mapping Method. In Long, B. T., Ishizaki, K., Toan, N. D., Parinov, I. A. & Kim, Y.-H. *Proceedings of the International Conference on Advanced Mechanical Engineering, Automation, and Sustainable Development 2021 (AMAS2021)* (pp. 810-816). Springer. https://doi.org/10.1007/978-3-030-99666-6_118

Ran, T., Yuan, L., & Zhang, J. B. (2021). Scene perception based visual navigation of mobile robot in indoor environment. *ISA transactions, 109*, 389-400. https://doi.org/10.1016/j.isatra.2020.10.023

Ren, C., Jiang, H., Li, C., Sun, W., & Ma, S. (2022). Koopman-Operator-Based Robust Data-Driven Control for Wheeled Mobile Robots. *IEEE/ASME Transactions on Mechatronics, 28* (1), 461-472. https://doi.org/10.1109/TMECH.2022.3203518

Shang, W., Zhu, H., Pan, Y., Li, X., & Zhang, D. (2021). A distributed model predictive control for multiple mobile robots with the model uncertainty. Discrete Dynamics in Nature and Society, *2021*, 9923496. https://doi.org/10.1155/2021/9923496

Singh, P., Agrawal, P., Nandanwar, A., Behera, L., Verma, N. K., Nahavandi, S., & Jamshidi, M. (2021). Multivariable event-triggered generalized super-twisting controller for safe navigation of nonholonomic mobile robot. *IEEE Systems Journal, 15*(1), 454-465. https://doi.org/10.1109/JSYST.2020.2985730

Xiao, X., Liu, B., Warnell, G., & Stone, P. (2022). Motion planning and control for mobile robot navigation using machine learning: a survey. *Autonomous Robots, 46*(5), 569-597. https://doi.org/10.1007/s10514-022-10039-8

Wang, D., Hu, Y., & Ma, T. (2020). Mobile robot navigation with the combination of supervised learning in cerebellum and reward-based learning in basal ganglia. *Cognitive Systems Research, 59*, 1-14. https://doi.org/10.1016/j.cogsys.2019.09.006

Wang, H., Zhang, F., & Wang, L. (2020). Fruit classification model based on improved Darknet53 convolutional neural network. *2020 International Conference on Intelligent Transportation, Big Data & Smart City (ICITBS)* (pp. 881-884). IEEE. https://doi.org/10.1109/ICITBS49701.2020.00194

Wang, X., Mizukami, Y., Tada, M., & Matsuno, F. (2021). Navigation of a mobile robot in a dynamic environment using a point cloud map. *Artificial Life and Robotics, 26*, 10-20. https://doi.org/10.1007/s10015-020-00617-3

Zhou, L., Rao, X., Li, Y., Zuo, X., Qiao, B., & Lin, Y. (2022). A lightweight object detection method in aerial images based on dense feature fusion path aggregation network. *ISPRS International Journal of Geo-Information, 11*(3), 189. https://doi.org/10.3390/ijgi11030189