Research on a Computer Vision-Based Guide Glasses System for the Visually Impaired

Enci Wang [1, *, #], Xinhui Cai [2, #], Sixuan Chen [3, #]

[1] Department of Computer Science and Engineering, Anhui University of Science and Technology, Anhui, China, 232001

[2] Department of Computer Science and Engineering, Tianjin University of Technology, Tianjin, China, 300382

[3] Department of Biomedical Engineering, Guangzhou Medical University, GuangDong, China, 511495

* Corresponding Author Email: 2022304321@aust.edu.cn

[#]These authors contributed equally.

Abstract. As the global population of visually impaired individuals continues to grow, traditional assistive tools struggle to meet the demands of safe navigation. This paper presents the design of an intelligent guide glasses system based on the deep learning model YOLOv5, integrating technologies such as ultrasonic ranging, computer vision, and global positioning to enable obstacle detection, environmental sensing, and navigation. By constructing a specialized dataset, Blind Vision-YOLO, and training the model, the system demonstrates impressive real-time performance and high precision in target detection. Experimental results reveal that the system achieves real-time target detection at 28.01 FPS with a mean average precision (mAP) of 74.1%, accurately identifying potential obstacles for visually impaired individuals during daily travel and providing timely voice feedback. The smart guide glasses designed in this study offer excellent portability and practicality, providing a safer and more convenient travel experience for the visually impaired.

Keywords: guide glasses, computer vision, YOLOv5, target detection, deep learning.

1. Introduction

According to the World Health Organization, by 2023, at least 2.2 billion people worldwide will be visually impaired or blind, with at least 1 billion suffering from visual impairment [1]. China, being the country with the highest number of blind people, currently has over 17 million visually impaired individuals, and the number of new blind cases is expected to increase by 450,000 every year [2]. As the global population of visually impaired people continues to grow, they face escalating challenges in terms of mobility and safety. In China, while "blind alleys" are provided on sidewalks to assist the visually impaired, insufficient supervision has led some of these alleys to become hazardous obstacles rather than aids. Furthermore, relying solely on blind alleys does not guarantee travel safety.

Traditional assistive tools, such as guide dogs, have notable drawbacks, including a limited number and high cost. A survey by the Association for the Blind indicates that there are only around 400 active guide dogs in China, a number that is far from sufficient to meet the increasing needs of the visually impaired population.

With the rapid advancements in computer vision and artificial intelligence in recent years, deep learning-based computer vision technologies have matured. This has led to the development of smart guide glasses for the blind, offering them a safer, more convenient, and intelligent way to travel. However, many existing studies on guide glasses rely on fixed datasets (such as the COCO dataset) for training. These datasets typically contain only common object categories and environments, which do not reflect the complex, dynamic real-world scenarios that visually impaired individuals face. Existing deep learning models often struggle to generalize to these diverse environments, causing the object recognition systems in guide glasses to perform poorly under specific conditions.

Currently, most guide glasses for the blind function as standalone devices, without integration with other assistive technologies. Although some guide glasses can connect with smartphones, smartwatches, or other smart devices to provide additional functions like voice control and real-time navigation, these systems often suffer from weak cross-device synergy. The lack of full integration and interoperability limits the comprehensiveness and flexibility of the solution.

To address these challenges, this project proposes a smart guide glasses system that can collect a variety of road information in real time and accurately detect obstacles ahead. The system integrates image classification, ultrasonic ranging, and other technologies with the YOLOv5 [3] object detection algorithm, which is capable of identifying obstacles that the blind may encounter during their daily travels, such as pedestrians, battery-powered vehicles, traffic signals, and more. Experimental results show that, using the YOLOv5 algorithm, the system achieves a mean average precision (mAP) of 74.1% in target detection and processes images in real time at a rate of 28.01 frames per second (FPS), ensuring sufficient real-time responsiveness. The system also provides timely obstacle alerts via voice feedback to help blind individuals avoid collisions and navigate safely on blind roads.

Moreover, the system integrates a Global Positioning System (GPS), allowing real-time location tracking of the blind user and providing their family members with location tracking services to further enhance travel safety. The ESP32 microprocessor, mounted on the glasses frame, is responsible for processing all the data and coordinating communication with other sensors and devices. This design ensures high-precision object detection while maintaining light weight and portability, meeting the daily mobility needs of visually impaired individuals.

2. Related work

2.1. Guide glasses for the blind

HAFEEZ ALI A. et al [4] proposed an application implemented on Google Glass2 so that it acts as a visual assistant for BVIP. By designing an application that solves the usability problems faced by previous applications and also provides real-time response to complex problems such as scene recognition and object detection. The application has a very intuitive interactive interface that allows the user to interact with it using voice commands that trigger the camera to capture images and send them to the mobile application.

Mais R. Kadhim et al [5] proposed the use of Deep Learning Yolo algorithm for object detection, which is implemented through the OpenCV library and Python language to build a Region Proposal Network (RPN) for target detection. The algorithm is trained on a set of images from the COCO dataset, on the basis of which the algorithm has demonstrated its high accuracy in detecting objects at a very high rate.

LIN Huiqi et al [2] proposed based on hardware platform Raspberry Pi sends trigger signals to control each sensor to detect obstacle distance, through Yolov3 and deep network training model to detect objects in life.

Dan Yufang et al [6] proposed the use of Nvidia Jetson development board as its control core, through the YOLO-v3 model, is responsible for object detection of the captured video, and its device can detect obstacles with a wider range, helping blind people to detect dangers in advance and improve the safety of traveling.

2.2. Target detection algorithms

Target detection is a core task in computer vision, and in recent years, deep learning-based algorithms have become the dominant approach in this field due to their high accuracy and efficiency. These algorithms are typically classified into two categories: two-stage and single-stage target detection algorithms. Two-stage algorithms (e.g., R-CNN (Region-based Convolutional Neural Networks) [7], Fast R-CNN [8], Faster R-CNN [9]) excel in detection accuracy but suffer from slower inference times due to the need for candidate region generation followed by classification. In contrast, single-stage algorithms (e.g., YOLO [10], SSD (Single Shot MultiBox Detector) [11]) streamline the detection process, improving speed while still meeting real-time detection requirements, albeit with slightly reduced accuracy in more complex scenarios.

The YOLO family of algorithms is widely recognized for its excellent balance between speed and accuracy. YOLOv5, for example, uses a single network structure that simultaneously predicts both the location and category of targets, offering high real-time performance with low computational overhead. This makes it particularly suitable for applications that require fast processing. YOLOv8 [12] improves the detection capabilities for small and dense targets based on YOLOv5, but this comes at the cost of increased computational complexity. Meanwhile, YOLOv11 enhances the network architecture to handle more complex detection tasks but demands higher computational resources, which may impact real-time performance.

In comparison, SSD performs well with multi-scale targets but slightly lags behind in accuracy, especially in complex environments. While R-CNN and its variants provide high accuracy, their slow inference speed makes them unsuitable for real-time detection tasks. Considering both speed and accuracy, YOLOv5 is selected as the target detection algorithm for this study. It offers a robust balance between real-time performance and high detection accuracy, making it ideal for embedded systems and guide-eye applications where quick and reliable target recognition is critical.

3. Spectacle Architecture

3.1. Overall structure

Guide glasses for the blind are smart devices that assist visually impaired people by integrating multiple sensors and modules to provide navigation, obstacle detection, and environmental information. We use a master control unit, ESP32, as the core processing unit of the system to perform the connection with the server and receive data from various sensors. Computer vision is used to train and test the objects and ultrasonic range sensors are used for obstacle detection, the camera on the glasses captures the user's surroundings in real time and sends the results of the object recognition and the detected information in the form of audio data to the speakers and feeds back to the user informing them of the objects around them. The overall architecture of the guide glasses product is depicted in Figure 1:

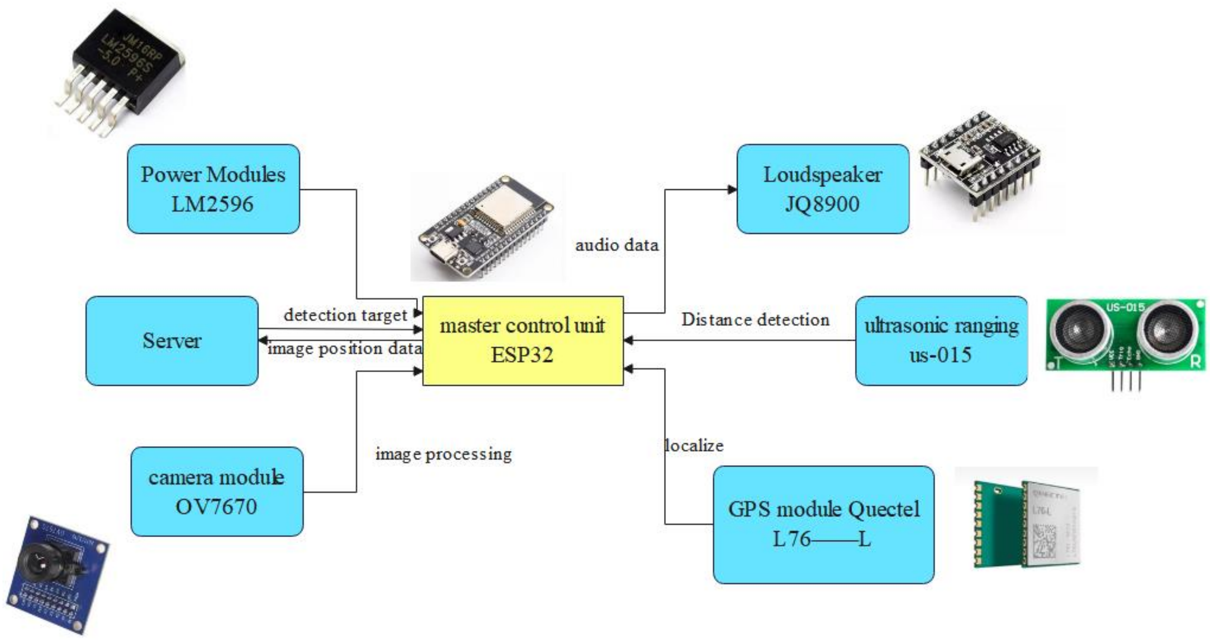


Figure 1. Overall architecture of the guide glasses

The power supply module LM2596, a step-down voltage regulator, is used to provide a stable voltage output, converting the input voltage to 3.3V or 5V for use by other modules. The system as a whole uses the master control unit ESP32 as the core processing unit, which is responsible for processing data from various sensors, executing the control logic, and communicating with other devices via Wi-Fi or Bluetooth. Obstacle avoidance is achieved through the ultrasonic sensor US-015, which measures distance by emitting ultrasonic waves and calculating the time delay of the reflected signal. The ESP32 processes the data collected by the ultrasonic sensor to measure the distance to the obstacle. The speaker module JQ8900 is used to play audio, such as alarm sounds and voice prompts. The GPS module Quectel L76-L provides global positioning services and communicates with the ESP32 via UART to monitor the device's position in real time. The camera module OV7670 captures images or videos in real time for visual identification or monitoring. It communicates with the ESP32 via SPI or I2C, sending image data and receiving control commands.

3.2. Target detection algorithms

3.2.1. YOLOv5 Architecture

The YOLOv5 network structure, as shown in Figure 2, consists of three main components: the backbone network, the neck network, and the head network. The backbone network is responsible for extracting low-level features from the image. YOLOv5 uses CSPDarknet53, an efficient and lightweight architecture, which reduces computational complexity while maintaining feature extraction accuracy. The neck network is designed to fuse the features extracted by the backbone and further integrate information across multiple scales using the Feature Pyramid Network (FPN) and Path Aggregation Network (PAN). This design enhances YOLOv5s ability to detect multi-scale targets, particularly small and dense objects, significantly improving detection performance. Finally, the head network performs target detection on the fused feature maps, outputting the bounding box coordinates and category information for the targets. YOLOv5s detection head is simplified and optimized to further improve inference speed and detection accuracy, particularly in real-time detection scenarios.

Additionally, YOLOv5 offers several versions of varying sizes, including YOLOv5n, YOLOv5s, YOLOv5m, YOLOv5l, and YOLOv5x, each providing different performance options depending on computational resource requirements. In this study, we selected YOLOv5n (the YOLOv5 Nano version), the lightest version in the YOLOv5 family. YOLOv5n has fewer model parameters and reduced computational demands, making it ideal for real-time target detection on devices with limited resources. It was chosen for its excellent inference speed with relatively minimal accuracy loss, making it particularly well-suited for embedded devices and applications with stringent real-time performance requirements.

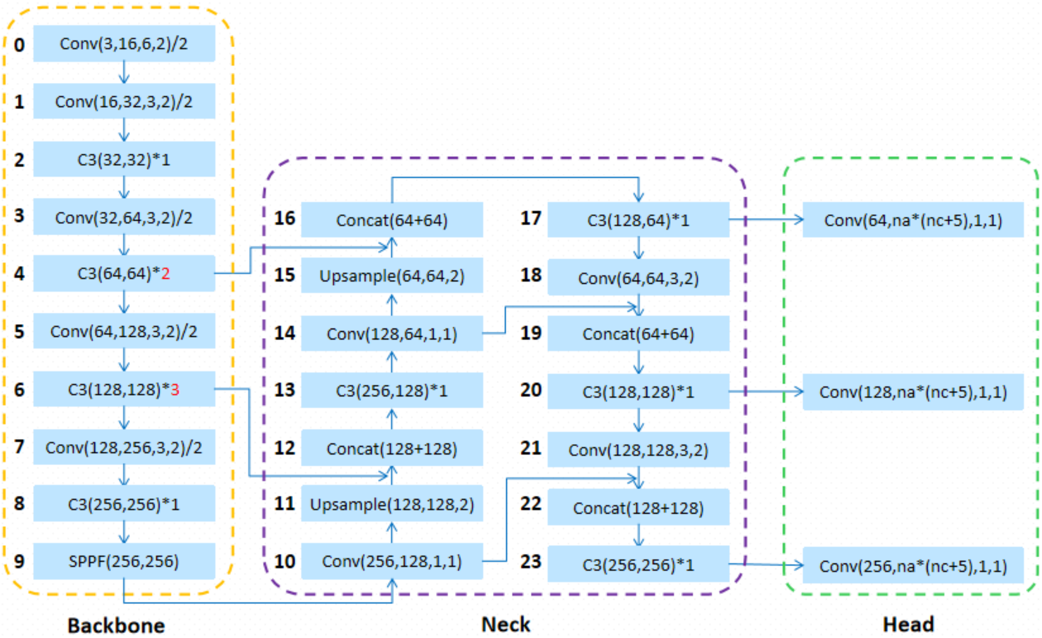


Figure 2. YOLOv5 network architecture

3.2.2. Establishment of data sets

To meet the needs of visual assistive systems for the blind, this study created the BlindVision-YOLO Dataset, specifically designed for the target detection task in guide glasses for the visually impaired. The goal of this dataset is to cover objects that blind individuals may encounter while walking in urban environments, providing high-quality labeled data to train the target detection model and support the development of an efficient, real-time visual assistive system. The dataset was sourced from multiple platforms, including Kaggle, Roboflow, and Flying Paddle, to ensure a broad and diverse range of data. The research team collected extensive image data from these platforms and filtered it according to the specific needs of the guide glasses system. Ultimately, 12 object categories were selected for inclusion in the dataset: blind alleys, pedestrians, greenbelts, motor vehicles, dogs, trash cans, fences, walls, bus stops, traffic signals, battery-powered vehicles, and bicycles. Each category contains 100 images. All of these object types are potential obstacles that visually impaired individuals may encounter in daily travel, making their detection crucial for the guide glasses system.

To ensure the accuracy and high quality of the dataset annotations, all images were manually labeled by the research team using the LabelImg tool. Each target in the images was labeled with bounding boxes and associated with category information. The labeling process adhered to a standardized procedure to ensure precise labeling and accurate bounding boxes for every target. The final dataset consists of 1,200 images and 2,253 instances, covering 12 categories. This ensures that the training process incorporates a diverse set of data with varying challenges. The detailed image data is summarized in Table.1. The dataset's size and the quality of its annotations make it an ideal resource for training the YOLOv5 target detection model, which can significantly enhance the performance of the system in real-world applications.

Table 1. Detailed data on the dataset

Category Number	Category Name	Number of Instances	Number of Pictures
0	blind pathway	127	100
1	pedestrian	2646	100
2	greenbelt	114	100
3	motor vehicle	104	100
4	dog	107	100
5	trash can	105	100
6	fence	253	100
7	wall	117	100
8	bus stop	101	100
9	traffic lights	247	100
10	electric bicycle	142	100
11	bicycle	136	100
total		4189	1200

3.2.3. Linkage of YOLOv5 with the Large Model

The large model of Tongyi Qianqian is integrated with an assistive dataset for the blind, named BlindVision-YOLO, which was independently constructed by our research team. This dataset includes 12 categories of images, such as blind alleys, pedestrians, green belts, and more. These images have been accurately annotated using the LabelImg tool, with each target object assigned corresponding bounding boxes and category labels. The dataset encompasses a wide variety of environmental conditions and lighting variations, ensuring the robustness and accuracy of the model across different scenarios. The deep learning framework PyTorch is used to build and train the generalized Tongyi Qianqian model. The model is trained for target detection based on the YOLOv5 architecture. To ensure effective generalization and evaluation, the dataset is divided into a training set and a validation set in an 8:2 ratio. The batch size is set to 8, with 100 training epochs, and the input image size is 640 × 640. During training, the model continuously adjusts its parameters using the gradient descent method to optimize network performance.

At the end of training, the model's performance is evaluated based on several metrics, including mean average precision (mAP), precision, recall, and frames per second (FPS), to comprehensively assess both the detection accuracy and speed. Special attention is given to the real-time capability of the model, ensuring that it responds quickly during inference and provides accurate target detection results, which is critical for the application in guide glasses for the blind.

4. Experimentation

4.1. Data set

Dataset: This paper utilizes the BlindVision-YOLO dataset, a custom dataset created by the research team. It was constructed by collecting image data from multiple sources and carefully screening it to meet the specific needs of guide glasses for the visually impaired. The dataset consists of 1,200 images, covering 12 categories: blind alleys, pedestrians, greenbelts, motor vehicles, dogs, garbage cans, fences, walls, bus stops, traffic signals, battery-powered vehicles, and bicycles. The images were captured under varying lighting, color, and environmental conditions, representing common objects encountered in daily life, ensuring diversity and practical applicability. Each category contains 100 images. The dataset is split into a training set and a validation set in an 80:20 ratio, where 80 images from each category are randomly selected for the training set, and the remaining 20 images are used for validation. The training set is used for model fitting and neural network parameter optimization, while the validation set serves for hyperparameter tuning and preliminary model evaluation. In training the YOLOv5n algorithm, the batch size is set to 8, with 100 epochs and an input image size of 640×640 pixels.

Experimental Environment: The experimental environment consists of a Windows 11 operating system, an Intel Core i7-12700H CPU, an NVIDIA GeForce RTX 3050 Ti GPU, 16 GB of RAM, and the PyTorch 2.4.1 deep learning framework, with CUDA 12.4 and Python 3.8.

Evaluation Metrics: The performance of the network in this paper is evaluated using several metrics, including mean Average Precision (mAP), Precision (P), and Recall (R). Additionally, Frames per Second (FPS) is used to assess the model's processing speed. The formulas for Precision (P), Recall (R), mAP, and FPS are provided in Eq. (1), Eq. (2), Eq. (3), and Eq. (4), respectively.

$$mAP = \frac{1}{n}\sum_{i=1}^{n}\int_{n}^{1} Pdr(i) \tag{1}$$

$$P = \frac{TP}{TP + FP} \tag{2}$$

$$R = \frac{TP}{TP + FN} \tag{3}$$

$$FPS = \frac{1}{processing\ time} \tag{4}$$

In Eq. n denotes the number of classes in the dataset. TP refers to the number of samples correctly predicted as positive classes, FP represents the number of samples incorrectly predicted as positive (i.e., false positives), and FN denotes the number of samples incorrectly predicted as negative (i.e., missed positives).

4.2. Comparative Experiments of Target Detection Algorithms

Table 2. mAP results for each category under different target detection algorithms

Backbone / Class	YOLOv5	YOLOv8	YOLOv11
Blind Pathway	63.2	70.2	59.8
Pedestrian	86.6	86.8	87.0
Greenbelt	87.6	92.0	97.8
Motor Vehicle	74.3	64.4	80.1
Dog	99.5	99.3	99.5
Trash Can	88.2	87.6	94.9
Fence	39.8	37.7	49.7
Wall	72.5	72.1	79.3
Bus Stop	88.9	91.6	94.6
Traffic Lights	36.8	38.0	36.5
Electric Bicycle	79.3	91.9	85.9
Bicycle	72.2	80.1	71.1

Table 3. Comparative experimental results of target detection algorithms

YOLO	Precision	Recall	mAP50	FPS
YOLOv5	77.9	69.2	74.1	28.01
YOLOv8	76.9	73.6	76.0	10.76
YOLOv11	83.8	71.6	78.0	12.80

To verify the superior performance of YOLOv5 in the target detection task for guide glasses for the blind, we selected three YOLO algorithms for comparative experiments: YOLOv5, YOLOv8, and YOLOv11. A single-variable control method was employed to ensure the accuracy and fairness of the experiments. Each algorithm was trained independently using the same experimental parameters, and the results are presented in Table 2 and Table 3.

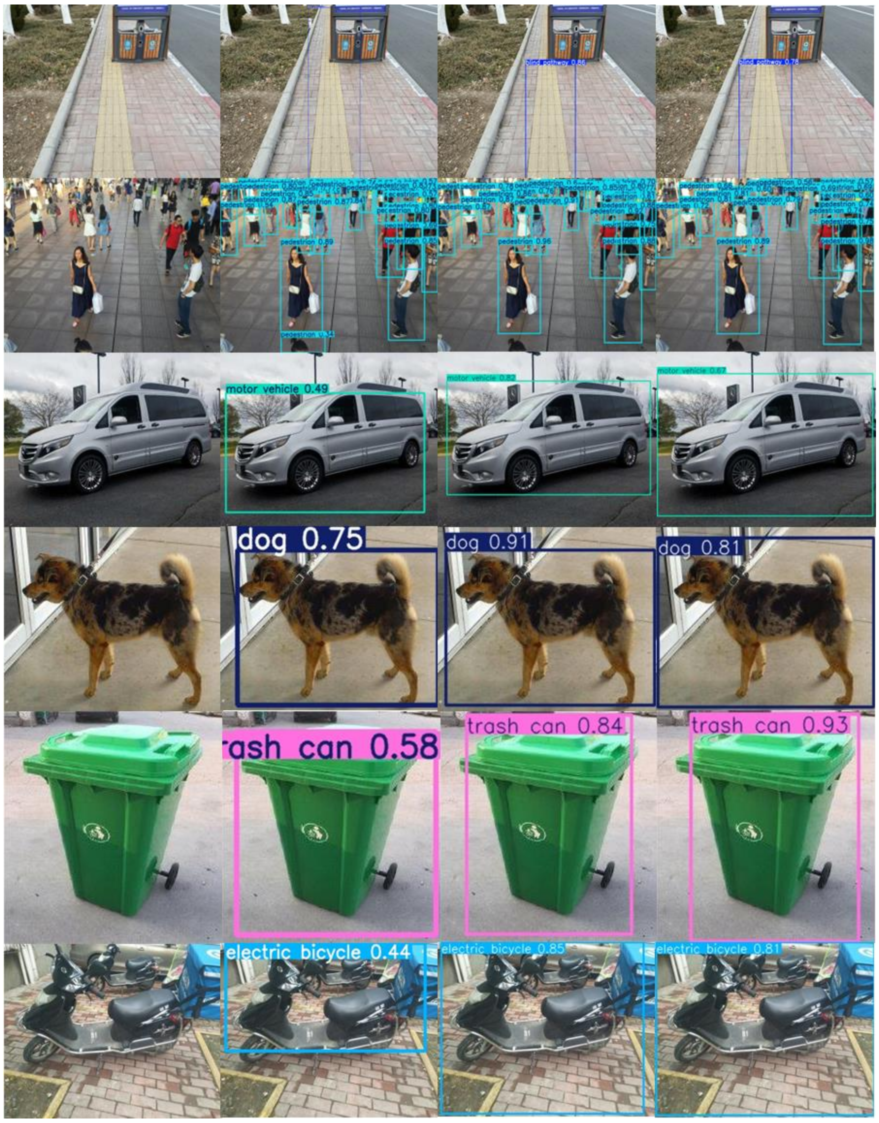
As shown in the experimental results in Tables.2 and 3, YOLOv5 demonstrates stability in overall performance, particularly excelling in inference speed. With an inference speed of 28.01 FPS, YOLOv5 significantly outperforms YOLOv8 and YOLOv11, highlighting its suitability for applications requiring high real-time responsiveness—especially in the target detection task for guide glasses for the blind. This task demands an extremely fast response time, and YOLOv5 meets this need by enabling real-time obstacle detection, thereby enhancing the safety and convenience of blind individuals. Despite YOLOv8's improved accuracy, its FPS is only 10.76, which is much lower than YOLOv5s; while YOLOv11 achieves a slightly higher FPS of 12.80, it still lags behind YOLOv5.

Furthermore, YOLOv5 performs well in terms of both precision and recall. It achieves a precision rate of 77.9% and a recall rate of 69.2%, indicating a more balanced detection capability. Although YOLOv8 achieves a higher recall rate of 73.6%, its precision rate is slightly lower at 76.9%, allowing YOLOv5 to maintain an edge in accuracy. YOLOv11, with a precision rate of 83.8%, outperforms YOLOv5 in precision but has a recall rate of 71.6%, which is slightly lower than both YOLOv5 and YOLOv8, making YOLOv11 more inclined towards reducing false positives.

In the mAP50 metric, YOLOv11 slightly surpasses both YOLOv8 and YOLOv5 with a score of 78.0%, showcasing its advantage in precision. However, when considering the balance between inference speed and overall performance, YOLOv5 remains the most optimal choice.

In conclusion, despite the improvements in accuracy seen with YOLOv8 and YOLOv11, YOLOv5s superior inference speed makes it the most suitable for the blind guide glasses target detection task. The comparative experiments confirm that YOLOv5 offers faster inference while maintaining a high level of detection accuracy, making it an ideal solution for real-world applications aimed at improving the safety and convenience of travel for the blind.

4.3. Visualization of test results



(a) Input_imge (b) YOLOv5 (c) YOLOv8 (d) YOLOv11

Figure 3. Comparative visualization of the results of the three target detection algorithms

To assess the detection performance of YOLOv5 across different scenarios, six distinct environmental conditions were randomly selected from the BlindVision-YOLO test set for experimentation. The visualization of detection results for YOLOv5, YOLOv8, and YOLOv11 is shown in Figure 3. As seen in the results, YOLOv5 demonstrates a remarkable ability to capture detailed target information. For instance, in the first scenario—blind pathway detection—YOLOv5 identifies more complete and specific targets. In the second scenario, pedestrian detection, YOLOv5 detects 39 pedestrians, whereas YOLOv8 and YOLOv11 detect only 29 and 19 pedestrians, respectively. This highlights YOLOv5s superior performance compared to both YOLOv8 and YOLOv11. When considering the results from the first two scenarios, it is evident that YOLOv5 excels in detecting smaller and more distant targets.

Although YOLOv5s overall accuracy in the subsequent four scenarios is slightly lower than that of YOLOv8 and YOLOv11, its detection frames are relatively compact, allowing it to localize targets with greater precision. This suggests that YOLOv5 is particularly effective for target detection in guide glasses for the blind, especially when detecting dense and small targets. It provides more detailed and accurate detection results, making it the most advantageous choice for this application scenario.

4.4. Multimodal System

This project also combines YOLOv5 and Tongyi Qianqian to form a multimodal system, which combines computer vision and natural language processing technologies to optimize the target detection and intelligent interaction functions in the guide glasses for the blind. Among them, YOLOv5 is responsible for real-time environmental target detection and localization, while Tongyi Qianwen utilizes its powerful natural language comprehension and generation capabilities to provide users with real-time, intelligent voice feedback, thus enhancing the assistive capabilities and interactive experience of the guide system.

4.4.1. Synergy between Target Detection and Speech Feedback

In this system, YOLOv5 is responsible for real-time detection of obstacles in the surrounding environment, such as blind alleys, pedestrians and traffic signals. Through its efficient single-stage detection framework, YOLOv5 is able to accurately acquire the location, category, and size of the target and deliver the information to the system. Based on these detection results, Tongyi Qianqian generates structured natural language descriptions and informs the user via voice output. For example, when a pedestrian is detected, the system can send out the voice prompt “There is a pedestrian in front of you, about 3 meters away, please pay attention to avoid”. This kind of interaction enables blind users to perceive the surrounding environment in real time through voice, improve spatial cognition and react in time.The combination of YOLOv5s efficient target detection and Tongyiqian's voice feedback provides low-latency and efficient environment perception, which can help blind people navigate better and improve their independence and safety in their life.

4.4.2. Error Detection and Semantic Compensation

Although YOLOv5 shows high detection accuracy in most scenarios, it may still face the problem of missed or misdetected targets in complex environments (e.g., when there is uneven lighting, a cluttered background, or many dynamic objects). In this case, Tongyiqian's natural language processing capability can be used as an effective complement to analyze the contextual information of the current environment and provide intelligent feedback. For example, when YOLOv5 fails to accurately recognize an obstacle, Tongyiqian can issue a reminder of “slippery road ahead, please walk carefully” based on the existing environmental knowledge, which can make up for the shortcomings of computer vision and improve the reliability of the system in complex environments. In addition, the system also supports voice interaction, users can query, “Is there a bus stop nearby?” Tongyiqian will provide real-time voice feedback based on YOLOv5s detection results, such as “There is a bus stop 100 meters to the left”. This multimodal interaction based on computer vision and natural language processing improves the intelligence and personalization of the system, helping the blind to better navigate and adapt to the environment.

5. Conclusion

In this study, a guide glasses system based on YOLOv5 has been developed with the goal of providing a smarter and safer travel experience for the visually impaired. This system combines computer vision and deep learning technologies, with the ESP32 serving as the core processing unit. It integrates hardware components such as ultrasonic sensors, GPS modules, speakers, and cameras to enable real-time obstacle detection, navigation, and environmental awareness. The system helps blind individuals recognize obstacles around them through visual recognition and ultrasonic feedback, providing timely voice prompts to ensure safe walking. The high inference speed and accuracy of YOLOv5 in this study allow it to excel in guide glasses applications, particularly in its ability to accurately and efficiently handle dynamic targets, achieving effective obstacle recognition in a variety of real-world environments.

References

[1] World Health Organization. (2023). World Health Statistics 2023: World Health Organization. https://iris.who.int/handle/10665/367912. License: CC BY - NC - SA 3.0 IGO.

[2] LIN Hui-Qi, ZHOU Yi-Tao, WENG Ming-Key, et al. Design and realization of AI intelligent guide glasses for the blind [J]. Information and Computer (Theoretical Edition), 2021, 33 (06): 171 - 173.

[3] Jocher, Glenn. "YOLOv5: A Family of Object Detection Architectures." GitHub, 2020, https://github.com/ultralytics/yolov5. Accessed 23 Oct. 2023.

[4] H. Ali A., S. U. Rao, S. Ranganath, T. S. Ashwin and G. R. M. Reddy, "A Google Glass Based Real-Time Scene Analysis for the Visually Impaired," in IEEE Access, vol. 9, pp. 166351 - 166369, 2021, doi: 10.1109/ACCESS.2021.3135024.

[5] Kadhim, Mais R. and Bushra K. Oleiwi. "Blind Assistive System based on Real Time Object Recognition using Machine learning." Engineering and Technology Journal (2022): n. pag.

[6] DAN Yufang, SHI Kaikai, LI Weiren, et al. Design and development of intelligent obstacle avoidance glasses for the blind [J]. Fujian Computer, 2021, 37 (04): 99 - 101.

[7] Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (pp. 580–587). Columbus, OH, USA: IEEE.

[8] GIRSHICK R. Fast r-cnn [C]. Proceedings of the IEEE international conference on computer vision, 2015: 1440 - 1448.

[9] REN S Q, HE K M, GIRSHICK R, et al. Faster R-CNN: towards real-time object detection with region proposal networks [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 39 (6): 1137 - 1149.

[10] REDMON J, DIVVALA S, GIRSHICK R, et al. You only look once: Unified, real-time object detection [C]. Proceedings of the IEEE conference on computer vision and pattern recognition, 2016: 779 - 788.

[11] LIU W, ANGUELOV D, ERHAN D, et al. Ssd: Single shot multibox detector [C]. Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11 – 14, 2016, Proceedings, Part I 14. Springer International Publishing, 2016: 21 - 37.

[12] Jocher, G., Chaurasia, A., & Qiu, J. (2023). Ultralytics YOLO (Version 8.0.0) [Computer software]. https://github.com/ultralytics/ultralytics.