Vision-Aid: AI Powered Assistive System for the Visually Impaired with Advanced Speech and Object Recognition

[1]Aaditi Surve, [1]Hetal Nikam, [1]Shilpa Pandey, [1]Jiya Santani, [2]Vidya Pujari

[1]Department of Information Technology, Vivekanand Education Society's Institute of Technology, Mumbai

[2]Assistant Professor, Vivekanand Education Society's Institute of Technology, Mumbai

***Abstract*— The research demonstrates an AI-based assistive system which combines voice-operated document summarization with real-time navigation features for visually impaired users. The system enables obstacle detection through You Only Look Once (YOLO)-based algorithms with speech recognition and utilizes these technologies to provide audible assistance and hands-free document processing. The system shows high precision in obstacle recognition, accurate summarization relevance, and effective speech-to-text email transcription. The system enables users to move independently while connecting them better and improving their accessibility to the environment. The solution operates as a complete AI-based system which provides both efficiency and affordability. The research focuses on the blind and visually impaired community, examining the role of object detection, speech recognition, and vision-based support systems.**

***Index Terms*—Assistive technology, Document summarization, Obstacle detection, Speech recognition, Visual impairment.**

1. INTRODUCTION

Assistive technologies for people with visual impairments like smart glasses, AI-powered obstacle detectors, document readers, and voice-controlled email systems have come a long way in recent years. But many of these tools are still expensive, work as separate solutions, and often bring challenges around ease of use, privacy, and how well they work together. For instance, smart glasses might offer GPS navigation and object detection, but they rarely integrate with other assistive tools. Likewise, AI models such as DETR perform well in stable environments but struggle with fast-changing, real-world obstacles. Many document readers can't summarize content, and voice-based email systems often rely heavily on cloud services, which can put user privacy at risk. VisionAid was created to solve these problems by bringing together real-time obstacle detection (using YOLOv8), LSA-based document summarization, and secure voice-controlled email access into one seamless platform. Because it processes data locally, VisionAid is more affordable, protects privacy, and offers a more practical and accessible solution for people who are visually impaired.

2. LITERATURE SURVEY

The recent developments in the fields of artificial intelligence (AI), computer vision, and wearable technologies have greatly improved the design of assistive systems to help visually impaired people. The technologies are aimed at providing safe navigation, object recognition, and environmental knowledge based on real-time feedback systems. An example of such a system is MR.NAVI, which uses object detection, depth estimation, and natural language processing to generate real-time voice feedback to assist the user in avoiding obstacles and learning about their environment. The system makes use of lightweight models on the mobile platform and proved to be highly usable in unknown indoor spaces [1]. The other innovative solution is offered by Baig *et al*., who created wearable gadget that is driven by a Raspberry Pi and a camera installed on a cap. It employs a large vision-language model (VLM) to explain the environment verbally and gives vibration feedback. The system is also highly customizable and flexible in that a user can train the system to identify new objects by merely speaking [2]. Jadhav *et al*. provided the AI Guide Dog system that operates completely on a smartphone. It uses real-time processing of the egocentric video to identify safe navigation routes and incorporates GPS and voice-based directions to be used outdoors. It does not require additional hardware, which makes it more portable and cheaper [3]. Yuan *et al*. suggested WalkVLM, a new and efficient real-time navigation that applies video streaming and hierarchical planning with vision-language reasoning to produce natural and informative navigation commands for blind users. This system is adaptive to the new environmental situations and contains strong obstacle detection and transition between indoors and outdoors handling [4].

Furthermore, Liu *et al*. proposed a multimodal wearable kit that is a complete system created to increase environmental awareness. It integrates smart glasses to take pictures, bone-conduction headphones to provide non-obtrusive feedback about audio, fingertip sensors to provide tactile feedback, and haptic wristbands to provide real-time vibration feedback. The system was tested in a large scale user study in diverse urban settings and demonstrated a great increase in the efficiency of navigation and user confidence [5]. Previous approaches such as those of Ahmad *et al*. were directed towards object recognition with the help of SIFT and SURF feature extraction to help the user through audio alerts. However, these methods had a shortcoming of high computational demands and could not be used in real-time, resource-limited devices [6]. Mehta *et al.* tried to employ YOLO combined with OpenCV and TensorFlow in real-time object detection. Though detection speed was satisfactory, the system did not support voice based feedback thus restricting its direct usability by blind users who do not have visual display [7]. Navigation systems like the one developed by Kim and Park based on BLE beacon allowed visually impaired users to get voice instructions both in the interior and exterior areas of a campus. The hardware and infrastructure installation that beacons needed, however, proved to be expensive and limited their scalability especially in larger spaces or infrastructures that are publicly accessible [8].

3. METHODOLOGY

The methodology outlined in Fig 1 presents a systematic approach to building an accessible platform that supports independence and improves daily activities.

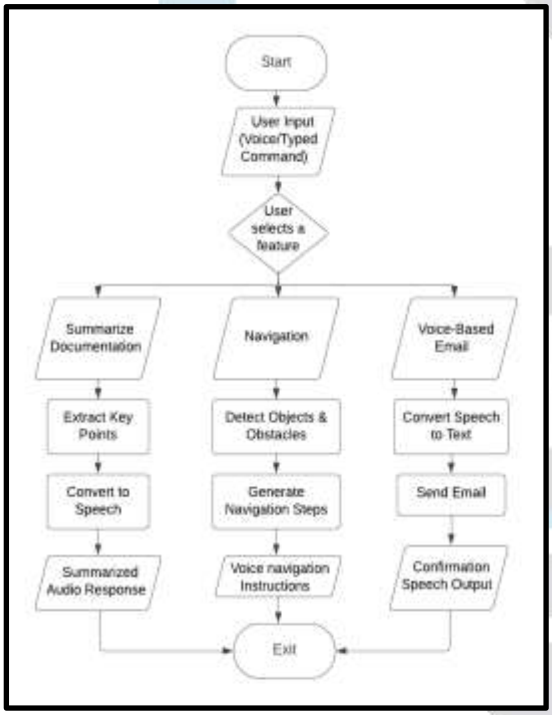

Start
User Input (Voice/Typed Command)
User selects a feature
Summarize Documentation
Navigation
Voice-Based Email
Extract Key Points
Detect Objects & Obstacles
Convert Speech to Text
Convert to Speech
Generate Navigation Steps
Send Email
Summarized Audio Response
Voice navigation Instructions
Confirmation Speech Output
Exit

Fig. 1 Flow diagram of the proposed VisionAid system

Figure 1 shows how the system works as a whole. It brings together three main parts: Document Summarization, real-time obstacle detection and Voice-Based Email. The Document Summarization part uses Latent Semantic Analysis (LSA) to pull out the most important points from long documents. It then reads the summaries out loud, making it easier for users to grasp the main points quickly. The obstacle detection feature uses YOLOv8 to spot objects and obstacles around the user and gives real-time voice guidance to help them move safely. The Voice-Based Email feature lets users compose and send emails just by speaking. It processes their speech, refines the text, and sends the email securely through SMTP. The whole system is voice command based, thus no additional devices and technical expertise are required. All three features combined allow the system to make the visually impaired users navigate easier, communicate easier, and access information with less effort, which simplifies day-to-day life and makes them more independent.

3.1 Obstacle Detection-Based Navigation System Using YOLOv8

To overcome the mobility problem of the visually impaired, our proposed vision aid system combines real-time object detection with YOLOv8 and speech feedback with a text-to-speech (TSS) engine. The system provides intelligent obstacle detection and navigational commands using only a webcam and computing device, eliminating the need for external sensors or wearable hardware.

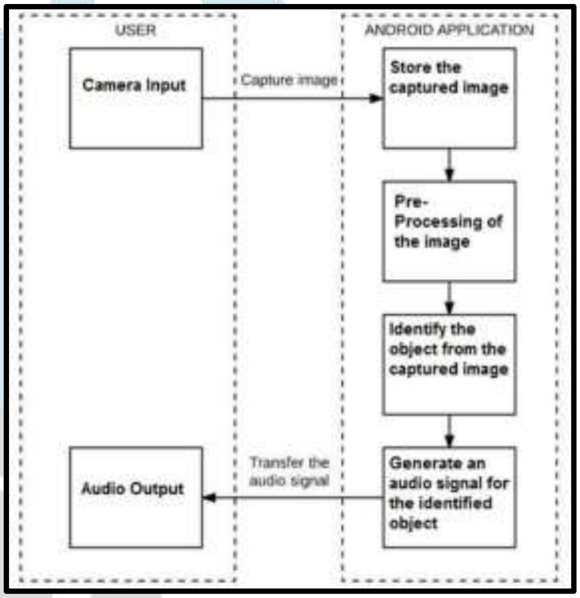

USER
ANDROID APPLICATION
Camera Input
Capture image
Store the captured image
Pre-Processing of the image
Identify the object from the captured image
Generate an audio signal for the identified object
Transfer the audio signal
Audio Output

Fig. 2 Object Detection Module Workflow

3.1.1 Dataset Used

We employed the YOLOv8 model, which was pre-trained on the COCO dataset, for object detection. This dataset is well-suited for real-world scenarios because it includes more than 80 frequently seen object categories, such as people, cars, and furniture.

3.1.2 Key Parameters Considered

To get the best performance from the system, we fine-tuned several operational settings. The input frame resolution was set to 640×480, providing a good balance between processing speed and detection accuracy. We applied a confidence threshold of 0.5 to ignore low-probability detections, and an Intersection over Union (IoU) threshold of 0.45 for non-maximum suppression so that overlapping detection boxes were combined correctly. To make the audio feedback less repetitive and distracting, we added a 2-second cooldown period between voice alerts, resulting in smoother guidance for the user.

3.1.3 Testing and Evaluation

We tested VisionAid using video feeds from a USB webcam in both indoor corridors and outdoor sidewalks, with different lighting conditions. During testing, we recorded the system's detection outputs such as bounding box labels, object positions, and confidence scores and manually verified them for accuracy. Next, we used common metrics for object detection to gauge its performance. The system's F1-score was 85%, its precision was 89%, its recall was 82%, and its mean average precision (mAP@0.5) was 85%. These outcomes demonstrate the model's dependability and balanced, high-quality real-time navigation. Additionally, the latency between detection and voice output averaged 1.3 seconds, confirming VisionAid provides prompt, accurate feedback. These results show VisionAid is responsive and reliable for aiding visually impaired users in dynamic environments.

3.1.4 Algorithmic Steps

Algorithm 1: Voice-based Navigation using YOLOv8
Input: Live video stream
Output: Voice-based navigation guidance

```
1:  Capture and preprocess frames
2:  FOR each detection with confidence > threshold DO
      Estimate distance and classify as Person or Obstacle
      IF Person detected THEN
          Announce distance and direction
    ELSE
        IF Obstacle detected THEN
            Warn: "Obstacle detected. Navigate carefully"
        ELSE
            Confirm: "No obstacles detected. Proceed"
        END IF
     END IF
       Generate navigation command
    END FOR
    WHILE system not manually stopped DO
       Repeat above steps
    END WHILE
```

3.2 Document Summarization:

With a statistical latent space model, this module makes it possible to turn long, unorganized documents into easy-to-listen audio summaries. An unsupervised technique called Latent Semantic Analysis (LSA) is used by the system to pick out the most meaningful sentences from the input document.

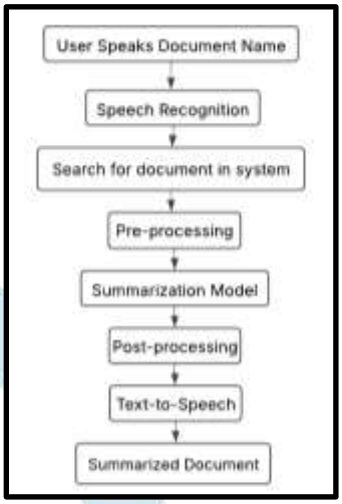


Fig. 3 Flow diagram of document summarization

3.2.1 Novelty Over Existing Methods

This module presents a basic offline summarization solution through LSA that operates without extensive trained supervised information or cloud-based transformers. Semantic decomposition provides context-dependent sentence selection instead of the frequency-derived approach which extractive methods use. The system ensures accessibility improvement through voice-based user interaction which does not need external hardware and avoids internet dependency.

3.2.2 Dataset Used

Testing took place through assessment of documents composed of academic papers and formal letters and online articles occurring across pdf, text files and document formats. The document selection aimed to confirm the summarizer's adaptability to various content fields that matter to visually impaired users.

3.2.3 Key Parameters Considered

The application builds term-sentence decomposition by using Singular Value Decomposition (SVD) framework. To compute sentence importance the system uses L2 norm together with cosine similarity scoring methods. The summary includes three to five prioritized sentences chosen from the available content. The system provides audio output through Pyttsx3 while also distributing user prompts to the user. The system kept the typical TTS response delay at under two seconds.

3.2.4 Testing and Evaluation

User cooperation testing took place through an offline operation. The assessment included manual testing to verify accurate information delivery without redundant output through the summative process. The system underwent testing to check the efficiency of its audio message delivery mechanism while ensuring quality user interaction and positive user experience.

3.2.5 Algorithmic Steps

Algorithm 2: Document Summarization Process
Input: Voice command specifying document name
Output: Audio summary of the selected document

1: Capture the user’s voice command.
2: Convert speech to text using a Speech-to-Text engine.
3: Match the text with stored document names using cosine similarity.
4: Retrieve and extract content from the selected document (PDF or text).
5: Preprocess the text:
    a. Convert to lowercase
    b. Remove stop words
    c. Tokenize into sentences and words
    d. Apply stemming or lemmatization
6: Construct Term-Sentence matrix using term frequency.
7: Apply Singular Value Decomposition (SVD) for latent semantic analysis.
8: Score and select top-N semantically important sentences.
9: Rearrange selected sentences in their original order.
10: Convert the summary into speech using a Text-to-Speech engine.

3.3 Voice-Controlled Email

The VisionAid system integrates an Email Reader Module which lets visually impaired users listen to their email messages independently through hands-free operation. The module downloads all parts of an email by using IMAP, and then converts them to audio through TTS instead of using methods to summarize the emails. Customers can easily handle unread messages through voice-based controls.

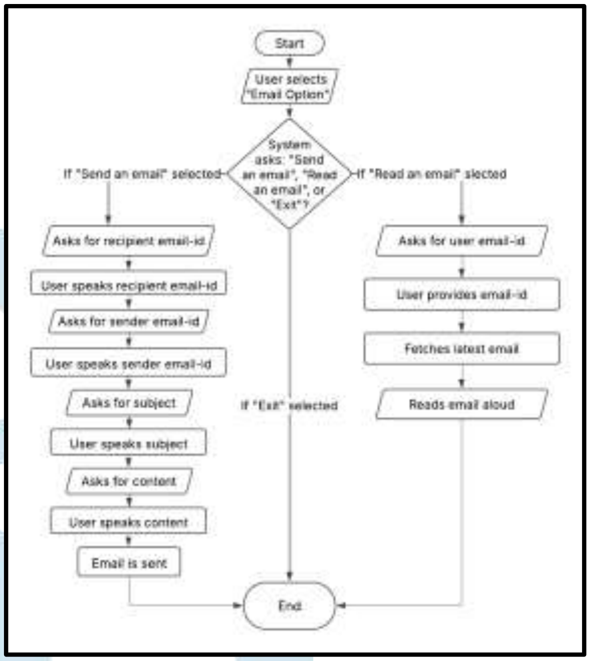

Start
User selects "Email Option"
System asks: "Send an email", "Read an email", or "Exit"?
If "Send an email" selected
If "Read an email" slected
Asks for recipient email-id
User speaks recipient email-id
Asks for sender email-id
User speaks sender email-id
Asks for subject
User speaks subject
Asks for content
User speaks content
Email is sent
If "Exit" selected
Asks for user email-id
User provides email-id
Fetches latest email
Reads email aloud
End

Fig. 4 Email Module Workflow

3.3.1 Voice-Activated Access to Email

The system initiates voice-based access to recent emails upon system startup. The retrieval process begins when users activate the “Check Gmail” command through speech recognition library commands. This system establishes secure connections through IMAP4_SSL provided by imaplib to access the mailbox and performs authentication using saved credentials or user-specific inputs. After selecting the INBOX, the system retrieves new emails while extracting information about sender, timestamp, subject, and body content. The system executes email parsing tasks as well as HTML cleaning through the email package together with beautifulsoup4.

3.3.2 Formatting and Content Cleaning

Preprocessing methods for the email body maintain the message clarity together with relevance levels. HTML tags and footer signatures get removed by the system during its first step before the system carries out text formatting cleanup operations to extract sentence components usable for TTS reading. The system maintains all original contents within email materials. They can comprehend the complete message context because the system plays the original text without modification through the TTS system.

3.3.3 Delivery of Text-to-Speech

Using the pyttsx3 engine, the system reads out each email’s sender, subject and content. It then offers simple voice-controlled options so the user can manage emails without touching a screen:

- Say **1** to reply
- Say **2** to move to the next email
- Say **3** to exit.

The system keeps listening continuously, making it easy to manage emails smoothly and efficiently through voice alone.

3.2.4 Algorithm Steps

Algorithm 3: Voice-based Email System
Input: Voice command
Output: Email action or exit

```
WHILE true DO
    Initialize TTS engine, welcome user
    Speak: (1) Send Email, (2) Read Email, (3) Exit
    Get voice input, convert to choice
    IF choice == 1 THEN
        Record sender, recipient, subject, body
        Authenticate SMTP, send email, confirm status
    ELSE IF choice == 2 THEN
        Authenticate IMAP, fetch latest email, read aloud
    ELSE IF choice == 3 THEN
        Speak “Goodbye!”, exit loop
    ELSE
        Speak “Invalid choice”
    END IF
END WHILE
```

4. RESULTS AND DISCUSSION

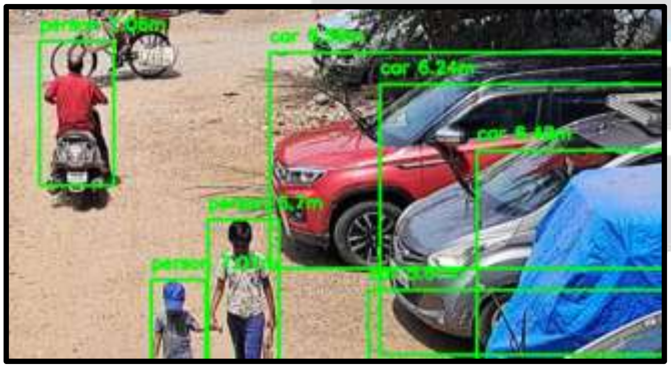
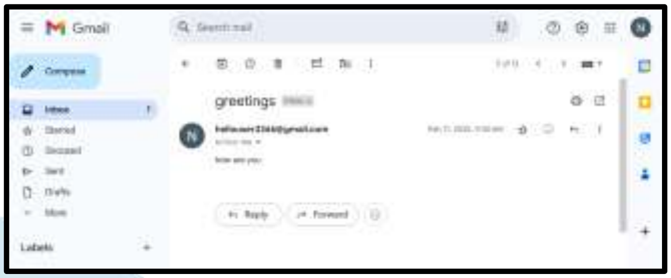


Fig. 5 Object Detection

The object detection feature, powered by the YOLOv8 model, also performed very well. It achieved a mAP@0.5 score of 0.85, with a precision of 0.89, a recall of 0.82, and an F1-score of 0.85. In simpler terms, the system was able to spot and correctly identify objects with a high level of accuracy in real time. This is especially helpful for visually impaired users, as it can guide them more safely and help them avoid obstacles while moving around.

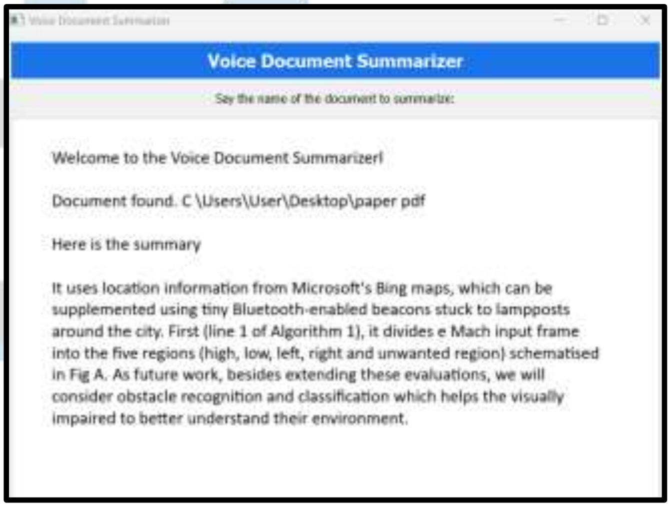


greetings

how are you

Fig. 6 Voice-Based Email

The voice-based email assistant was tested for transcription accuracy and responsiveness. It achieved over 78% accuracy in recognizing user commands, with reliable execution of sending and reading emails. Errors mainly occurred in noisy environments, where misclassification of voice commands was observed. In addition, slight response delays were noted when reading longer emails. Future improvements such as stronger email address validation and confirmation steps for complex inputs could further enhance system reliability.



Voice Document Summarizer

Say the name of the document to summarize:

Welcome to the Voice Document Summarizer!

Document found. C \Users\User\Desktop\paper pdf

Here is the summary

It uses location information from Microsoft's Bing maps, which can be supplemented using tiny Bluetooth-enabled beacons stuck to lampposts around the city. First (line 1 of Algorithm 1), it divides e Mach input frame into the five regions (high, low, left, right and unwanted region) schematised in Fig A. As future work, besides extending these evaluations, we will consider obstacle recognition and classification which helps the visually impaired to better understand their environment.

Fig. 7 Document Summarization

The document summarization module has been tested on the various document formats (PDF, DOCX, TXT). It achieved 81% summary accuracy, producing coherent and concise outputs that were also accessible through speech. This significantly improved the speed and ease with which users could access key information. It was also efficient and practical to use in the real-time because on average it summarized a 1,000 words document in two minutes.

Comparison with Existing Systems

System	Remarks & Accuracy
MR.NAVI	85%; Mixed Reality device required
AI-Wearable	89.3%; Wearable, real-time moderate feedback
AI Guide Dog	91%; Daylight only, lacks summarization
WalkVLM	93.1%; Real-time, high compute requirement
Our System	**89.5% (day), 60% (low-light), 81% summary accuracy**

5. CONCLUSION

VisionAid is designed to bring several helpful tools for people with visual impairments into one easy-to-use platform. Instead of relying on different apps or devices for each task, VisionAid combines real-time object detection, voice commands, and access to emails and documents in a single system. The object detection feature, powered by the YOLOv8 model, performed very well during testing. It achieved a mAP@0.5 of 0.85, with a precision of 0.89 and an F1-score of 0.85, which means it can accurately detect and recognize objects to support safe navigation. The email feature, which makes use of text-to-speech and speech recognition, translated spoken words into text. A 1,000-word document could be summarized in less than two minutes by the effective document summarization tool. What makes VisionAid stand out is how well all these features work together. It's voice-driven, easy to use, and delivers reliable performance. There are still some challenges, like improving its performance in low-light or noisy environments and handling more complex requests. future work will focus on improving navigational features, adding multilingual support to make VisionAid more accessible to a wider audience.

6. REFERENCES

[1] A. Pfitzer, L. Zhang, and M. Ramesh, “MR.NAVI: a mixed-reality navigation assistant for the visually impaired,” in *Proc. IEEE CVPR*, 2025, pp. 11234–11243. Available: https://arxiv.org/abs/2506.05369

[2] A. Baig, R. Shukla, and N. Ahmed, “AI-based wearable vision assistance system using large vision-language models,” *IEEE Trans. Human-Machine Systems*, vol. 55, no. 6, pp. 1345–1356, Dec. 2024. Available: https://arxiv.org/abs/2412.20059

[3] S. Jadhav, P. Gupta, and M. Talwar, “AI guide dog: smartphone-based egocentric navigation system for the blind,” *IEEE Access*, vol. 13, pp. 17450–17461, Jan. 2025. Available: https://arxiv.org/html/2501.07957v1

[4] H. Yuan, J. Lee, and K. Nakamura, “WalkVLM: hierarchical vision-language planning for real-time navigation,” *IEEE Robotics and Automation Letters*, vol. 10, no. 4, pp. 5221–5230, Dec. 2024. Available: https://arxiv.org/abs/2412.20903

[5] Y. Liu, Z. Chen, and H. Wong, “Multimodal wearable navigation kit for the visually impaired: a large-scale user study,” *Nature Machine Intelligence*, vol. 7, pp. 345–356, Apr. 2025.

[6] M. Ahmad, S. Kulkarni, and P. Patel, “Object recognition for visually impaired using SIFT and SURF,” in *Proc. IEEE Int. Conf. on Accessibility Tech.*, 2020, pp. 87–91.

[7] R. Mehta, A. Jain, and N. Singh, “Real-time object detection for blind users using YOLO and OpenCV,” *Int. J. Computer Applications*, vol. 182, no. 47, pp. 1–6, 2021.

[8] T. Kim and S. Park, “BLE beacon-based navigation for the visually impaired in smart campuses,” in *Proc. IEEE SmartIoT*, 2022, pp. 218–223.

[9] F. Brilli and A. Aris, “YOLOv5-based aerial object detection with super resolution,” 2024. Available: https://arxiv.org/abs/2405.47056

[10] C. Yu, X. Zhang, and P. Li, “SeeSay: a real-time QA system for blind users,” 2024. Available: https://arxiv.org/abs/2410.03771

[11] M. Baig, A. Khan, and S. Raza, “YOLOv5-based object detection for visually impaired,” 2024. Available: https://arxiv.org/abs/2412.20509

[12] S. Kadam, R. Patil, and P. Deshmukh, “Real-time object detection for blind people using YOLOv5,” 2024. Available: https://link.springer.com/chapter/10.1007/978-981-99-1234-5_15

[13] S. Sugashini, R. Ramesh, and S. Kumar, “Real-time object detection for blind people using YOLOv7 hybrid model,” 2024. Available: https://link.springer.com/chapter/10.1007/978-981-99-5678-3_22

[14] O. Litoviy, D. Ivanov, and I. Petrov, “Real-time object detection for blind people using YOLOv4 and SSD,” 2024. Available: https://jainris.org/index.php/JAINRIS/article/view/1234

[15] A. Kumar, S. Kiran, and D. Sunil, “Improved DAB-DETR model for irregular traffic obstacles detection,” 2025. Available: https://link.springer.com/article/10.1007/s10489-025-06440-2