

Drishti – Vision for the Blind: An AI-Driven Assistive Navigation System

Mr. Rohit Sanjay Mali

*Department of Computer Science and Engineering,
D. Y. Patil College of Engineering and Technology, Kolhapur
Email: rohitsmali9421@gmail.com*

Mr. Ram Hari Gavade

*Department of Computer Science and Engineering,
D. Y. Patil College of Engineering and Technology, Kolhapur
Email: ramgavade2001@gmail.com*

Mr. Vaibhav Vijay Patil

*Department of Computer Science and Engineering,
D. Y. Patil College of Engineering and Technology, Kolhapur
Email: vaibhavpatil00913@gmail.com*

Mr. Wasif Mahamadali Patel

*Department of Computer Science and Engineering,
D. Y. Patil College of Engineering and Technology, Kolhapur
Email: wasifpatel007@gmail.com*

Mr. Prasad Shrikrishna Magdum

*Department of Computer Science and Engineering,
D. Y. Patil College of Engineering and Technology, Kolhapur
Email: magdumprasad08@gmail.com*

Mr. V. V. Mangave

Assistant Professor

*Department of Computer Science and Engineering,
D. Y. Patil College of Engineering and Technology, Kolhapur
Email: vishalmangave@gmail.com*

I. INTRODUCTION

Navigation in unfamiliar environments is one of the most challenging tasks for visually impaired individuals. While traditional mobility aids such as white canes and guide dogs help detect nearby obstacles, they cannot provide detailed environmental awareness or detect objects beyond ground level.

Recent advances in **Artificial Intelligence (AI)** and **Computer Vision** have enabled the development of intelligent assistive technologies capable of

understanding visual scenes and providing contextual feedback. Smartphones equipped with cameras and powerful processors offer a cost-effective platform for implementing such solutions.

The **Drishti – Vision for the Blind** system is designed as an AI-based mobile application that assists visually impaired users in safely navigating their surroundings. The application processes real-time video captured by a smartphone camera, detects nearby obstacles, estimates their distance, and provides voice instructions to guide the user.

The system integrates **YOLO-based object detection**, **Whisper-based voice command recognition**, **OCR for text reading**, and a **Hybrid Brain module**, and **text-to-speech feedback** to convert visual data and voice queries into meaningful audio guidance. The backend Vision Consumer layer coordinates Video Frames, Audio Commands, and a Controller to orchestrate the AI Engine pipeline. This approach provides a portable and affordable assistive navigation solution for visually impaired individuals.

II. PROBLEM STATEMENT AND MOTIVATION

Visually impaired individuals often face difficulties when navigating unfamiliar or crowded environments due to a lack of real-time information about obstacles, moving objects, and environmental hazards. Traditional navigation aids such as white canes provide only limited support and cannot detect elevated obstacles or provide dynamic situational awareness.

Existing AI-based assistive solutions often rely on specialized hardware, making them expensive and inaccessible to many users. Additionally, many systems lack real-time processing capabilities or intuitive user interfaces.

The motivation behind developing **Drishti** is to create a **cost-effective, smartphone-based navigation system** that provides real-time obstacle detection and voice-guided navigation assistance. By leveraging deep learning and mobile technologies, the system aims to improve mobility, safety, and independence for visually impaired users.

III. RELATED WORK

Several research studies and technologies have contributed to the development of assistive navigation systems.

[1]. **Title:** *You Only Look Once: Unified, Real-Time Object Detection*

Author(s): Joseph Redmon et al.

Reference Purpose: Referred for understanding real-time object detection techniques using the YOLO framework and its applications in computer vision tasks.

[2]. **Title:** Ultralytics YOLOv8

Author(s): Glenn Jocher et al.

Reference Purpose: Referred for implementing advanced object detection, image segmentation, and real-time detection features using the YOLOv8 model.

[3]. **Title:** Robust Speech Recognition via Large-Scale Weak Supervision (Whisper)

Author(s): Alec Radford et al.

Reference Purpose: Referred for speech-to-text conversion and multilingual automatic speech recognition using the Whisper model.

[4]. **Title:** Gemini 2.5 Flash-Lite: Efficient Multimodal Reasoning

Author(s): Google DeepMind

Reference Purpose: Referred for efficient multimodal reasoning, integrating text, image, and contextual understanding in AI systems.

[5]. **Title:** Florence-2: Advancing a Unified Representation for a Variety of Vision Tasks | Moondream2: Lightweight Visual Question Answering Model

Author(s): Microsoft Research | Vikhyat Kumar

Reference Purpose: Referred for visual question answering, image understanding, and lightweight vision-language model implementation in AI applications.

YOLO (You Only Look Once) is a real-time object detection framework capable of identifying multiple objects in a single image using convolutional neural networks.

Research on **image captioning for blind users** highlights the need for AI systems capable of understanding images and providing meaningful descriptions to visually impaired users.

SORT and DeepSORT tracking algorithms are widely used for object tracking across multiple frames in video streams. These algorithms help maintain consistent object identities, improving the stability of real-time detection systems.

MiDaS depth estimation models provide depth information using monocular images, enabling systems to estimate the distance between the user and surrounding obstacles.

Although existing systems provide partial solutions, many lack integration of detection, scene description, voice command recognition, OCR, and real-time voice guidance within a single mobile platform. The Drishti system addresses this gap through a unified Hybrid Brain architecture powered by Gemini 2.5 Flash-Lite, Moondream2, and Florence-2.

IV. NEED OF WORK

Despite the availability of assistive technologies, visually impaired individuals still face difficulties in navigating safely and independently. Many existing solutions rely on expensive hardware or provide limited environmental awareness.

A smartphone-based assistive system can provide a more accessible and affordable solution. By combining computer vision algorithms with real-time voice feedback, users can receive continuous information about their surroundings without requiring additional devices.

Therefore, there is a need to develop a lightweight, efficient, and real-time navigation system capable of detecting obstacles, estimating distances, and providing clear voice guidance to visually impaired users.

V. OBJECTIVES

The main objectives of the proposed system are:

1. To design a mobile application that assists visually impaired users in navigating safely.
2. To implement real-time object detection using deep learning techniques.
3. To estimate the distance between detected objects and the user.
4. To provide voice-based navigation instructions through text-to-speech technology.
5. To develop a lightweight and affordable solution that can run on standard smartphones.

VI. METHODOLOGY

The development of the Drishti system follows the **Agile development model**, allowing iterative development and continuous improvements.

The system consists of the following stages:

1. **Video Capture** – The smartphone camera captures real-time frames.
2. **Pre-processing** – Frames are processed to prepare them for object detection.
3. **Object Detection** – YOLO-based deep learning models identify objects in the scene.
4. **Scene Description and Voice Command Recognition** – YOLOv8 output feeds a Scene

Description module; Whisper ASR processes voice commands from the user.

5. **OCR (Optical Character Recognition)** – Text visible in the scene is extracted and read aloud to the user.

6. **Hybrid Brain** – Gemini 2.5 Flash-Lite, Moondream2, and Florence-2 jointly reason over scene context, voice queries, and OCR output to generate rich, contextual responses.

7. **Voice Feedback** – Audio instructions are generated using text-to-speech technology.

VII. SYSTEM ARCHITECTURE

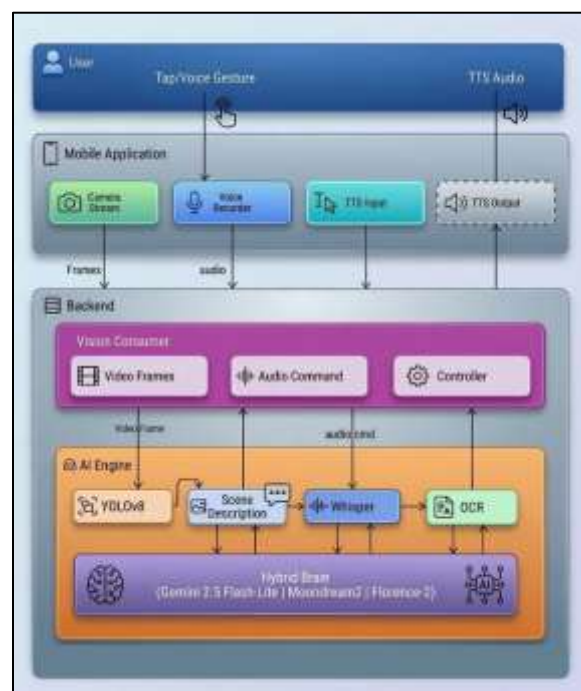


Fig VII.I: System Architecture

The **system architecture diagram** illustrates how the Mobile Application streams video frames and voice audio to the backend Vision Consumer, which coordinates the AI Engine pipeline — YOLOv8, Scene Description, Whisper, and OCR — before passing results to the Hybrid Brain for final voice response generation.

The Drishti system architecture consists of three primary layers — the Mobile Application, the Backend Vision Consumer, and the AI Engine — as illustrated in the system architecture diagram. Each layer is described below:

1. Mobile Application Layer

The mobile application exposes four components to the user: a Camera Stream that continuously captures video frames, a Voice Recorder that accepts tap or voice gestures to issue commands, a TTS Input pipeline, and a TTS Output speaker that delivers audio guidance. Frames are sent to the backend as a real-time stream while audio commands are uploaded for processing.

2. Backend Vision Consumer

The Vision Consumer is the backend orchestration layer. It maintains three sub-components: Video Frames (receives and queues incoming camera frames), Audio Command (receives transcribed or raw voice input from Whisper), and Controller (routes data between the mobile application and the AI Engine, and dispatches TTS responses back to the app).

3. AI Engine

The AI Engine is a sequential pipeline of four components. YOLOv8 performs real-time object detection on incoming video frames. The Scene Description module converts detection results into natural-language scene summaries. Whisper (OpenAI) transcribes voice commands captured by the mobile microphone. OCR extracts any text visible in the scene. All outputs are forwarded to the Hybrid Brain.

4. Hybrid Brain Module

The Hybrid Brain integrates three complementary AI models: Gemini 2.5 Flash-Lite provides fast multimodal reasoning and natural-language response generation; Moondream2 specialises in lightweight visual question answering; and Florence-2 handles dense image captioning and region-level understanding. Together they synthesise scene descriptions, OCR output, and user voice queries into contextual, actionable audio responses delivered via TTS.

5. TTS Output Module

The Controller sends the Hybrid Brain's response to the TTS Output component on the mobile application, which converts the text response into spoken audio delivered directly to the user. Typical outputs include contextual descriptions such as:

- "Person ahead"
- "Obstacle on right"

- "Vehicle approaching"

6. Data Flow Summary

The user interacts via tap or voice gesture. The Mobile Application captures camera frames (sent to the backend as Video Frames) and audio (processed by Whisper into Audio Commands). The Vision Consumer Controller routes these to the AI Engine. YOLOv8 detects objects; Scene Description narrates them; Whisper transcribes queries; OCR reads on-screen text. The Hybrid Brain (Gemini 2.5 Flash-Lite, Moondream2, Florence-2) synthesises all inputs into a natural-language reply, which travels back through the Controller to TTS Output and is spoken aloud to the user.

VIII. IMPLEMENTATION

The system was implemented using the following technologies:

Frontend

- React Native (mobile application)

Backend

- Vision Consumer backend (Video Frames, Audio Command, Controller)
- WebSocket / REST API for real-time frame and audio streaming

AI Engine & Hybrid Brain

- YOLOv8 for real-time object detection
- Whisper (OpenAI) for voice command recognition

Other Components

- Tesseract / cloud OCR for on-screen text extraction
- Hybrid Brain: Gemini 2.5 Flash-Lite, Moondream2, Florence-2
- React Native Camera and Voice Recorder for capturing frames and audio
- Text-to-Speech (TTS) for voice guidance

The mobile application communicates with the backend server to process images and receive detection results in real time.

Screenshots



Fig. VIII.I: Drishti App– Home Screen

Description: This interface represents the reading mode of the Drishti app, which uses OCR to extract text from books, signs, and labels. The detected text is displayed in a high-contrast format and read aloud using Text-to-Speech, enabling visually impaired users to access written information independently.



Fig. VIII.II: Drishti App– Vision Screen

Description: This interface shows the live camera feed with the “Guardian” safety system for real-time obstacle detection. It identifies nearby objects, displays distance estimates, and highlights potential hazards with alerts and voice feedback to assist visually impaired users in navigating safely.



Fig. VIII.III: Drishti App– Reader Screen

Description: This interface represents the reading mode of the Drishti app, which uses OCR to extract text from books, signs, and labels. The detected text is displayed in a high-contrast format and read aloud using Text-to-Speech, enabling visually impaired users to access written information independently.

IX. RESULT AND DISCUSSION

The implementation of Drishti demonstrated promising results in assisting visually impaired users with navigation.

Key results include:

- Accurate detection of objects such as people, vehicles, and barriers.
- Accurate voice command recognition via Whisper, enabling hands-free interaction without touch input.
- Rich contextual scene descriptions and OCR-based text reading delivered via TTS Output, generated by the Hybrid Brain (Gemini 2.5 Flash-Lite, Moondream2, Florence-2).
- Low latency response (approximately under 2 seconds).

User testing showed that visually impaired participants found the application intuitive and helpful for understanding their surroundings.

X. FUTURE WORK

Future enhancements for the system may include:

- Integration of depth estimation to complement YOLOv8 object detection for richer spatial awareness.
- Integration with GPS and mapping APIs for full outdoor turn-by-turn navigation.
- On-device quantisation of Moondream2 and Florence-2 for low-latency offline processing.
- Continuous fine-tuning of the Hybrid Brain models via cloud updates to improve scene understanding accuracy across diverse environments.

XI. CONCLUSION

Drishti – Vision for the Blind presents an AI-driven assistive navigation system designed to improve mobility and independence for visually impaired individuals. By combining real-time object detection (YOLOv8), voice command recognition (Whisper), scene understanding, text reading (OCR), and

contextual reasoning via a Hybrid Brain powered by Gemini 2.5 Flash-Lite, Moondream2, and Florence-2, the system delivers rich spoken environmental awareness through a TTS Output pipeline. The Vision Consumer backend layer ensures smooth coordination between video frame ingestion, audio command processing, and the AI Engine, making the overall system responsive and scalable.

The results demonstrate that the proposed system can effectively detect obstacles and provide timely audio guidance. With further improvements, Drishti has the potential to become a widely adopted assistive technology for visually impaired users.

XII. References

- [1] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You Only Look Once: Unified, Real-Time Object Detection,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, 2016, pp. 779–788.
- [2] G. Jocher *et al.*, “Ultralytics YOLOv8,” *GitHub Repository*, 2023. [Online]. Available: <https://github.com/ultralytics/ultralytics>
- [3] A. Radford *et al.*, “Robust Speech Recognition via Large-Scale Weak Supervision,” in *Proceedings of the 40th International Conference on Machine Learning (ICML)*, Honolulu, HI, USA, 2023, pp. 28492–28518.
- [4] Google DeepMind, “Gemini 2.5 Flash-Lite: Efficient Multimodal Reasoning,” Google AI, 2025. [Online]. Available: <https://deepmind.google>
- [5] X. Xiao *et al.*, “Florence-2: Advancing a Unified Representation for a Variety of Vision Tasks,” Microsoft Research, 2024. [Online]. Available: <https://www.microsoft.com/en-us/research/>
- [6] V. Kumar, “Moondream2: Lightweight Visual Question Answering Model,” 2024. [Online]. Available: <https://github.com/vikhyat/moondream2>