

AI-Powered Smart Blind Stick for Real-Time Scene Narration

Ms. Pooja¹, Ms. Meghana N², Ms. Nandini³, Ms. Rajeshri⁴, Prof. Deepthi⁵

^{1,2,3,4,5} *Department of Computer Science and Engineering, Shree Devi Institute of Technology, Mangaluru*

Abstract—People with severe visual impairment depend largely on the white cane, which senses an obstacle only after touching it and cannot say what the obstacle is. This paper presents an AI-powered smart blind stick that describes the surroundings to the user by voice in real time. The prototype combines a USB webcam, OpenCV and the YOLO11 object-detection model to identify objects and to decide whether each one lies on the left, in the center or on the right of the user. An HC-SR04 ultrasonic sensor, read through an Arduino Uno, measures the distance of the obstacle and passes it to a Python program over a USB serial link. When the obstacle is within 100 cm, the program joins the object name, distance and direction into a spoken sentence, and when it is 50 cm or closer the Arduino also sounds a buzzer. Prototype tests showed that persons, bottles, suitcases, dogs, cars and a teddy bear were announced with the correct side and distance. By pairing camera-based recognition with sensor-measured distance at low hardware cost, the proposed design gives the user a single, easily understood message instead of a bare alarm.

Index Terms—Smart blind stick, Artificial Intelligence, YOLO11, Object Detection, OpenCV, Ultrasonic Sensor, Arduino Uno, Voice Guidance, Assistive Technology.

I. INTRODUCTION

Moving safely through streets, corridors and public places is a daily difficulty for visually impaired people. The white cane is the most common aid because it is inexpensive and needs no power, but it gives information only through contact. The user discovers an obstacle when the tip hits it and cannot tell whether it is a wall, a parked vehicle, a chair or a person walking nearby.

Electronic canes tried to widen this reach by adding ultrasonic sensors and buzzers, so that the user is warned before touching an obstacle [5], [6]. Such devices, however, raise the same kind of alarm for

every object. Their usefulness is limited because the user still does not know what is ahead or on which side it lies.

Progress in deep learning has made real-time object recognition practical. Detectors of the YOLO family have been combined with speech output in several assistive systems [1], [3], [7], [8], so that detected objects can be read aloud. Many of these systems, though, estimate distance from the size of the detected box, a value that changes with the camera, the viewing angle and the object itself, and some demand considerable computing resources [2], [4].

A practical aid therefore needs to answer three questions together: what is the object, where is it, and how far away is it. This work proposes an AI-powered smart blind stick in which a webcam with YOLO11 supplies the first two answers, an HC-SR04 ultrasonic sensor supplies the third, and a buzzer and a voice engine deliver the result to the user.

The proposed system follows a Capture–Detect–Measure–Decide–Narrate framework. Its main contributions are:

- a low-cost architecture that fuses YOLO11 detection with ultrasonic distance measurement;
- a simple left / center / right rule based on the position of the detected object in the frame;
- a two-level alert scheme using voice narration within 100 cm and an added buzzer warning at 50 cm or less; and
- a working prototype whose test outputs show real-time narration of object, direction and distance.

II. LITERATURE REVIEW

Research on mobility aids for the visually impaired has moved from simple distance sensing towards camera-based recognition and speech output. The studies

reviewed below form the technical background for the proposed smart blind stick.

- [1] Kothapalli et al. developed an obstacle-detection and audio-navigation system based on YOLO. It detects obstacles, estimates how far they are and gives spoken guidance in real time. The authors note that the quality of the result depends on the camera and the model in use.
- [2] Kela et al. proposed a blind-assistance system that joins YOLOv3 with optical character recognition, so that both objects and printed text are converted to speech. The combination broadens the information given to the user but needs more computation.
- [3] Singh et al. trained a YOLOv8 detector on a custom dataset and linked it with voice commands. Detection is fast, while the approach requires dataset preparation, training effort and adequate hardware.
- [4] Ikram et al. studied assistive object detection with a transformer-based DETR model and compared it with other deep-learning architectures. Detection quality is high, but the models need large training data and substantial computing power.
- [5] Illakiya and Loganathan built a smart blind stick from an Arduino UNO, an ultrasonic sensor, a buzzer and a GSM module. It is cheap and warns of nearby obstacles, although it needs several hardware modules and network connectivity.
- [6] Ulfa et al. presented S-Cane, a cane that uses an ultrasonic sensor and an Arduino to detect obstacles. It is simple and low in cost, but it cannot tell the user what kind of object has been found.
- [7] More and Sangamkar used YOLOv8 together with distance estimation and text-to-speech to assist blind users. The accuracy of the estimated distance can vary with the conditions of use.
- [8] Santhosh et al. applied YOLO-based detection and audio feedback to identify objects for visually impaired users. Real-time operation is achieved, but processing capacity and environmental effects limit performance.
- [9] Ahmad et al. designed a smart cane with several sensors and a model-based state-feedback controller. Awareness of the surroundings improves, but the cost and complexity of the system are higher.

- [10] Narayani et al. integrated a camera module with ultrasonic sensing in a smart cane. Using two kinds of sensing helps detection, although the camera result depends on lighting.

Research Gap

The reviewed work shows that ultrasonic canes are affordable but anonymous [5], [6], that camera-based systems identify objects but often depend on lighting and on estimated distance [1], [7], [10], and that multi-sensor designs raise cost and complexity [9]. From this review, the following gaps were identified:

- Ultrasonic-only canes warn of an obstacle but do not tell the user what the obstacle is [5], [6].
- Several YOLO-based aids estimate distance visually, which varies with camera and conditions [1], [7].
- Systems that use heavier models or extra modules need more computing power, hardware and cost [2], [4], [9].
- Few studies give one spoken sentence containing the object name, its side and a sensor-measured distance, with a graded buzzer warning.

The proposed AI-powered smart blind stick addresses these gaps by combining YOLO11 detection, left / center / right direction estimation, HC-SR04 distance measurement, buzzer warning and voice narration in one low-cost framework.

Table I. Comparison with Existing Approaches

Approach	Object type	Real distance	Cost
Ultrasonic cane [5],[6]	No	Yes	Low
YOLO + speech [1],[3],[7],[8]	Yes	Estimated	Medium
Camera + ultrasonic [10]	Partial	Yes	Medium
Multi-sensor [9]	No	Yes	High
Proposed system	Yes	Yes	Low

III. METHODOLOGY

The proposed smart blind stick combines camera-based recognition, ultrasonic ranging, embedded control and voice output so that the user is told what is ahead, on which side and how far. The prototype follows a closed processing loop in which the surroundings are repeatedly captured, analysed and

reported. The methodology consists of image acquisition, object detection, direction estimation, distance measurement, alert decision and voice narration. Fig. 1 shows the overall flow.

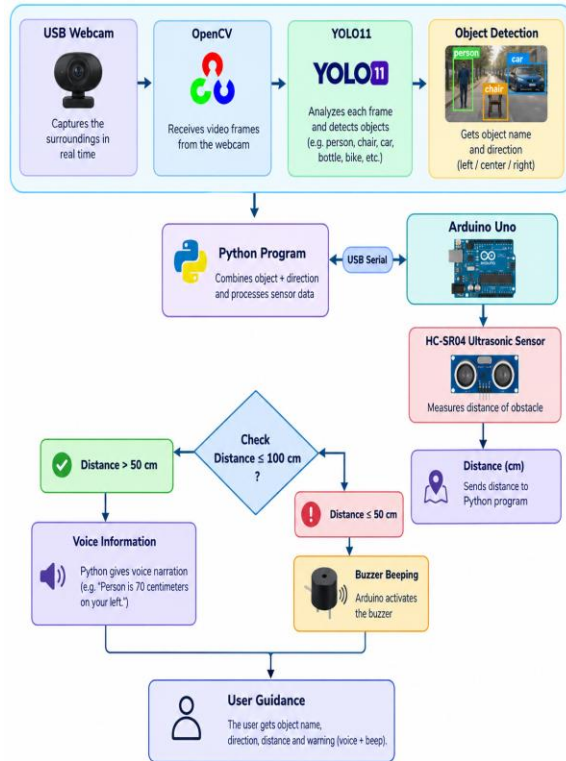


Fig. 1: System Architecture and Flow Diagram

A. Prototype Architecture

The system has two input paths that meet in a Python program. In the vision path, a USB webcam supplies frames that OpenCV passes to the YOLO11 model. In the distance path, an HC-SR04 ultrasonic sensor is read by an Arduino Uno, which forwards the distance to the Python program through a USB serial link. The program decides what to announce, and the Arduino drives the buzzer.

B. Image Acquisition and Object Detection

The webcam continuously captures the scene in front of the user. OpenCV reads the video frames and hands each frame to YOLO11, which returns the class name, a confidence score and a bounding box for every detected object, such as a person, chair, car, bottle or bike. The detections are drawn on the live window for verification.

C. Direction Estimation

Let W be the frame width and x_c the horizontal center of a bounding box. The object is placed in one of three zones:

left if $x_c < W/3$; center if $W/3 \leq x_c \leq 2W/3$; right if $x_c > 2W/3$

This label is attached to the object name before the message is built.

D. Distance Measurement

The HC-SR04 sensor emits an ultrasonic pulse and the Arduino converts the echo time into a distance d in centimetres. The Arduino sends d to the Python program over the serial link, so the distance is a measured value and does not depend on the size of the object in the image.

E. Alert Decision Logic

Two thresholds are used. The detection range is 100 cm. The warning distance is 50 cm. The decision is: $d > 100$: display only; $50 < d \leq 100$: voice information; $d \leq 50$: voice warning + buzzer

A speech cooldown of 10 seconds prevents the same message from being repeated continuously.

F. Closed-Loop Process

The stick works as a continuous cycle. The webcam and the ultrasonic sensor sense the surroundings, the Python program processes the data, a decision is made, and the user receives voice or buzzer feedback. As the user reacts and moves, the next frame and distance reading begin the cycle again:

Webcam → OpenCV → YOLO11 → Direction + Distance → Decision → Voice / Buzzer → User → Repeat

This Capture–Detect–Measure–Decide–Narrate cycle is the core of the proposed system.

IV. IMPLEMENTATION

The prototype was built as a working system with hardware assembly, firmware for the Arduino, a Python program for detection and narration, and voice output. These parts are described below.

A. Hardware Implementation

The hardware used in the prototype is:

- USB webcam
- Arduino Uno board
- HC-SR04 ultrasonic sensor

- Buzzer
- Computer for processing, with audio output
- USB cable for serial communication

The ultrasonic sensor is connected to the Arduino, which also controls the buzzer. The Arduino and the computer exchange data over USB serial, and the webcam is connected directly to the computer.

Table II. Function Of Each Component

Component	Function
USB webcam	Captures live video of the surroundings
Arduino Uno	Reads the sensor, drives the buzzer, talks to Python
HC-SR04	Measures distance to the obstacle
Buzzer	Gives a warning beep at 50 cm or less
Computer	Runs detection, decision logic and voice output

B. Embedded Software

The Arduino sketch triggers the ultrasonic sensor, converts the echo time to centimetres and writes the value to the serial port. It also switches the buzzer on when the close-range condition is met. The basic cycle is:

Initialize → Trigger Sensor → Compute Distance → Send to Python → Control Buzzer → Repeat

C. Detection and Narration Program

The Python program was developed and run in Visual Studio Code. Its main steps are:

1. Open the USB webcam and load the YOLO11 model.
2. Open the serial connection to the Arduino.
3. Read a frame and run object detection.
4. Compute the direction of each object from its bounding box.
5. Read the latest distance value from the Arduino.
6. Compare the distance with the 100 cm and 50 cm thresholds.
7. Build the sentence from object, distance and direction.
8. Speak the sentence, respecting the speech cooldown.
9. Show the annotated frame and repeat until the user presses Q.

D. Voice Narration

A text-to-speech engine converts the message into audio. Within the detection range the message states the object, the distance and the side, for example

“person detected at 80 centimeters on your left”. At close range it begins with a warning, for example “Warning! car is very nearby. It is 32 centimeters ahead of you”. The speech engine is re-initialised for every utterance so that an earlier message does not delay a later one.

E. Software and Development Tools

Table III. Software Tools Used

Tool	Purpose
Python	Main program and decision logic
OpenCV	Webcam capture and frame display
YOLO11	Real-time object detection
Arduino sketch (C/C++)	Ultrasonic reading and buzzer control
Text-to-speech engine	Voice narration
Visual Studio Code	Program development

V. RESULTS AND DISCUSSION

The prototype was tested to check the working of its detection, distance, alert and voice parts. At start-up the program reports the webcam in use, a detection range of 100 cm, a warning distance of 50 cm and a speech cooldown of 10 seconds, and the live window shows each detected object with its bounding box and confidence value.

Fig. 2 and Fig. 3 show sample outputs and Table IV lists messages recorded in the terminal during testing. In each case the spoken message names the object, gives the side and states the distance, and the wording changes from information to warning once the distance falls to 50 cm or below. This goes beyond ultrasonic-only canes [5], [6], which cannot name the object, and avoids the box-size distance estimates used in [1], [7].

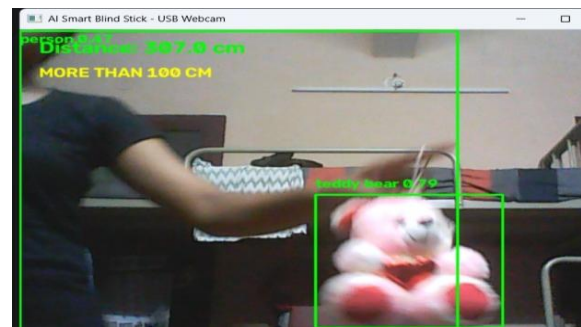


Fig. 2: Detection of a person and a teddy bear with voice output



Fig. 3: Detection of a bottle with a close-range warning

Table IV. Sample Outputs Observed During Testing

Object	Distance (cm)	Direction	Alert type
Person	80.3	Left	Voice
Person	83.2	Left	Voice
Suitcase	66.9	Center	Voice
Teddy bear	80.1	Center	Voice
Dog	65.3	Left	Voice
Dog	50.4	Left	Voice
Person	39	Center	Voice + warning
Car	32.2	Center	Voice + warning
Bottle	36.0	Center	Voice + warning

The camera supplies what the object is and where it lies, while the ultrasonic sensor supplies how far it is. Limiting speech to the 100 cm range and suppressing repeated messages keeps the user from being overloaded. The prototype uses a common webcam, an Arduino Uno and an HC-SR04 sensor, so its hardware cost is low.

The prototype also has limits. Detection quality depends on lighting and camera quality, the ultrasonic sensor gives one reading along its beam that may not always match the object seen by the camera, and processing is done on a computer. Quantitative values such as detection accuracy, distance error and response time have not been stated here because they must be obtained from systematic trials on the prototype rather than assumed.

VI. CONCLUSION

This paper presented an AI-powered smart blind stick that narrates the surroundings in real time using a USB webcam, OpenCV, YOLO11, an HC-SR04 ultrasonic sensor, an Arduino Uno, a buzzer and a voice engine. The prototype identifies common objects, places them on the left, center or right, measures their distance and speaks a message, adding a buzzer warning when the obstacle is very close.

Compared with a plain cane or an ultrasonic-only cane, the stick tells the user what the obstacle is, while keeping the cost low. Systematic testing is still needed to quantify accuracy, response time and reliability in different environments before wider use.

VII. FUTURE WORK

The prototype can be improved in several directions. The accuracy of object detection can be raised, and more types of obstacles and objects can be added to the recognised set. The voice instructions can be made clearer so that the user understands them at once, and obstacle detection in low-light conditions can be strengthened. The stick can be made lighter, more compact and more power-efficient, for example by moving the processing to an embedded board. Finally, the system should be tested in varied real-world environments and evaluated with measured trials, including trials with visually impaired users.

REFERENCES

- [1] D. Kothapalli *et al.*, "Real-time obstacle detection and audio navigation for the visually impaired using YOLO," 2025.
- [2] G. P. Kela *et al.*, "Vision to voice: An advanced blind assistance system integrating YOLOv3 and OCR technologies for enhanced mobility," 2025.
- [3] Singh *et al.*, "YOLOv8 based object detection with custom dataset and voice command integration," 2025.
- [4] S. Ikram *et al.*, "Enhancing object detection in assistive technology for the visually impaired: A DETR-based approach," 2025.
- [5] N. Illakiya and V. Loganathan, "Comprehensive assistive mobility system for

- the visually impaired: The smart blind stick with ultrasonic sensor solution,” 2024.
- [6] Ulfa *et al.*, “S-Cane: Ultrasonic sensor-based smart cane for the visually impaired,” 2023.
 - [7] P. More and S. Sangamkar, “Vision aid for blind people using YOLOv8,” 2024.
 - [8] L. Santhosh S. *et al.*, “Object detection and identification for visually impaired,” 2024.
 - [9] N. S. Ahmad *et al.*, “Multi-sensor obstacle detection system via model-based state-feedback control in smart cane design for the visually challenged,” 2018.
 - [10] T. Lavanya Narayani *et al.*, “Design of smart cane with integrated camera module for visually impaired people,” 2023.