International Journal of Creative and Open Research in Engineering and Management
ISSN: **3108-1754 (Online)**
Volume 02 Issue 05 May-2026 | Impact Factor: 3.5

A Real-Time Assistive Navigation Framework for Visually Impaired Users Using Spatial Reasoning and Computer Vision

Adepu Nikhil[1]
Department of Computer Science & Artificial Intelligence
Central University of Andhra Pradesh
Email: adepunikhil79@gmail.com

Dr. C. Krishna Priya[2]
(Assistant Professor)
Department of Computer Science & Artificial Intelligence
Central University of Andhra Pradesh
Email: krishnapriyarams@gmail.com

Abstract—People with visual impairments deal with major hurdles when trying to move around on their own. This stems from a reduced sense of what is happening around them and the drawbacks of standard mobility tools. Most of the help systems available today do not respond quickly enough and fail to give a clear picture of the surroundings.

In this work, we put forward a real-time navigation aid built on AI that looks to boost both movement freedom and user safety. The setup makes use of computer vision to spot obstacles and grasp the scene. A reasoning module that works with spatial data then turns what is seen into straightforward directional hints for the user.

Sound alerts are also woven into the design so that guidance reaches the user through spoken cues at the right moment. One thing that sets our work apart from older systems is that it does not ask for any special equipment. It runs on everyday devices most people already own, which keeps the door open for wide use without extra cost.

Our thinking draws from the latest findings in navigation support and object spotting studies [1], [2]. What the tests show is that the setup catches objects quite well while keeping delays very short. These traits make it a solid pick for use in actual day-to-day travel.

Index Terms—Assistive Navigation, Computer Vision, Object Detection, Visually Impaired, Real-Time Systems, Spatial Reasoning, Audio Feedback

I. INTRODUCTION

A. Background and Motivation

Vision loss makes moving through the world much harder. Things that sighted people do without thinking, like crossing a road, stepping around something in the way, or making sense of a busy space, become tough and risky. The white cane has been around for a long time. Guide dogs are helpful too. But both have limits. They do not tell the user what is ahead or warn about things coming from the side.

In the last few years, AI and computer vision have started to open new doors. Systems now exist that can look at a live video feed, pull out useful bits, and share them with the user. The idea is plain: let the machine fill in what the eyes miss. This can help a person move with more safety and less worry [1]. It is about closing the gap between what a person sees and what a machine can figure out from an image.

One big shift is that models have gotten lighter. They can run on phones now. Since many people already own a phone, this makes the help cheaper and easier to reach.

B. Challenges in Existing Systems

Even with this progress, some problems have not gone away. A lot of the work stops once an object is found. The system spots a chair or a pole but does not say what to do next. Without the next step, the tool is not as useful as it could be.

Some systems also ask for extra sensors or special boxes. That adds cost and weight. Not many people want to carry more gear. Speed is another thing. If there is even a small lag, the warning might come too late. That could lead to a fall instead of preventing one.

Then there is the question of where an object sits. Knowing something is there is one part. Knowing if it is to the left, straight ahead, or to the right is another part. Without that, the alert is not very clear.

The way the user gets the alert also matters. Raw labels read aloud do not help much. What works is short, clear speech that comes at the right time [2].

C. Recent Developments

Recent studies have looked at deep models that can run on phones. These models can now find many objects in one frame, even in busy scenes [3].

Phone-based setups have drawn interest because they do not need extra parts. The camera and processor inside the phone do all the work. That makes the whole thing easier to fit into daily life [4].

Some groups have also tried to add spatial reasoning and better feedback. But what is still missing in many cases is one clean pipeline that joins detection, spatial sense, and clear output from start to end.

D. Proposed Approach

This paper tries to fill those gaps. We put forward a real-time navigation aid that brings together object detection, spatial reasoning, and voice feedback in one system.

The system uses computer vision to find obstacles in live frames and build a rough picture of what is nearby. A zone-based reasoning step cuts the frame into three strips: left, middle, and right. From that split, the system pulls simple hints about which way to go.

Voice cues carry the output, so the user can keep their hands free. The whole setup is built to run on plain devices without extra hardware. This keeps costs down and makes it easier for more people to use.

E. Contributions

What this work adds can be summed up as:

- A real-time navigation aid built on computer vision.
- A spatial reasoning layer that adds direction to the guidance.
- Voice feedback made for hands-free use while walking.
- A light and scalable design that suits real-world use.
- One pipeline that links detection, reasoning, and feedback all the way through.

II. LITERATURE REVIEW

A. Early Assistive Navigation Systems

The first wave of tools made to help visually impaired people get around leaned on sensors. Things like ultrasonic and infrared sensors were used to spot objects that were close by. These setups were built to give simple warnings, maybe a vibration or a basic sound alert. They did not cost much and were easy enough to put together, but they fell short when the space got busy or complex. They could not read the scene or give any real sense of what was going on around the user.

Later on, vision-based systems came into the picture to try and fix these gaps. Li et al. [1] put forward an indoor navigation setup that used computer vision to guide visually impaired users. The system did a better job of picking up what was in the room, but it was mostly built for indoor spaces. Taking it outside into changing, real-world settings was not really workable.

B. Deep Learning-Based Object Detection

The rise of deep learning gave object detection a big push forward. Newer methods draw on convolutional neural networks and more advanced vision models to pick out several objects at once, in real time.

Ikram et al. [2] brought in a deep learning method that used transformer-based detection for assistive use. The method showed stronger accuracy and held up better in busy or cluttered spaces. The catch was the computing load. Running such models on small, low-power devices stayed a problem.

Along similar lines, Pratap et al. [3] looked at adaptive object detection aimed at helping with indoor navigation. The results in controlled tests looked good, but the system had trouble keeping up in real time and did not adapt well when tested under different conditions.

C. Mobile-Based Assistive Systems

As mobile computing took off, quite a few studies turned toward smartphone-based navigation aids. The draw here was clear: use a device people already own, cut the need for outside hardware, and make the help portable.

Jadhav et al. [4] built a smartphone navigation tool that used egocentric vision to deal with changes in the path ahead. The system scored well on ease of use and reach, but it called for a fair bit of computing power and did not nail the real-time feedback side.

Budrionis et al. [6] put out a broad review of smartphone-based assistive tools. They stressed how real-world use hinges on finding the right balance between how well the system runs, what it costs, and how easy it is to pick up and use.

D. Integrated Navigation Frameworks

More recent work has tried to pull different pieces, object detection, path guidance, user interaction, into one joined-up system.

Kuriakose et al. [7] came out with DeepNAVI, a deep learning navigation aid that pairs object sensing with route guidance. It gave users a better feel for the space, yet it still lacked a solid way to reason about direction when giving feedback.

Litoriya et al. [8] put forward a vision-based system leaning on YOLO and SSD models for real-time object detection. The system did well at finding things, but the main focus stayed on detection. It did not really get into what a user needs for navigation, like hints about direction or feedback that fits the situation.

E. Challenges in Existing Systems

Even with all the steps forward, some stubborn problems hang on. A good number of systems zero in on finding objects but stop there. Turning what is detected into a clear action the user can take is still missing. That gap makes the tools less usable when it counts, out in the real world.

Chimwanga [9] pointed out the hurdles object detection systems face when built for visually impaired users. Things like accuracy slipping under poor lighting and the strain of handling fast-changing settings were flagged.

On a similar note, Sameer et al. [10] dug into the challenge of tying AI detection to real-time feedback. They underlined how much the whole thing depends on fast processing and interaction models that feel natural to the user.

An et al. [11] put together an AI-based navigation system with some locating features. The early results looked promising, but the authors noted that more work was needed on scaling up, speeding up processing, and keeping the design centered on user needs.

F. Research Gap

Stepping back and looking at the published work, one thing stands out. Most systems fixate on a single piece, maybe detection or maybe navigation help, instead of bringing everything together into a working whole.

There is a clear opening for a system that weaves together real-time object detection, spatial reasoning, and feedback that feels natural and fast. On top of that, many current tools struggle with real-time speed and do not fully address what users actually need.

This paper aims to step into that gap. We lay out a full assistive navigation setup that joins perception, spatial reasoning, and voice feedback into one framework. The goal is to push forward both how well it works in real time and how usable it feels for the person relying on it.

III. METHODOLOGY

We built a real-time navigation aid that ties together a few pieces. A camera grabs live video. A detection model spots things in the scene. A reasoning step sorts out where those things sit. Then a voice output tells the user what to do next.

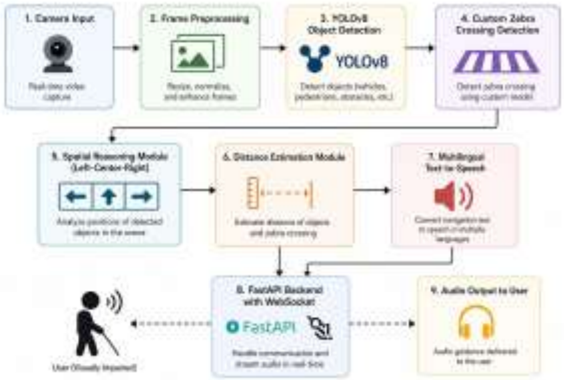

Fig. 1. Overall System Architecture

Fig. 1 sketches the whole pipeline. It goes like this: pull in video, clean up frames, run detection, apply spatial logic, speak the result.

A. Data Acquisition and Preprocessing

Frames come in through a camera, one after the next. Each frame is handled on its own so the flow stays smooth.

A handful of quick fixes hit each frame before detection runs. Frames get scaled to the size the model needs. Pixel numbers are pulled into a tight range. The image gets packed into a tensor. Noise gets dialed down. Contrast gets a small bump. Nothing fancy. These tweaks just stop the detection from failing when lighting shifts.

B. Object Detection Module

The detection side leans on a deep model. It scans a frame and returns boxes, labels, and confidence values.

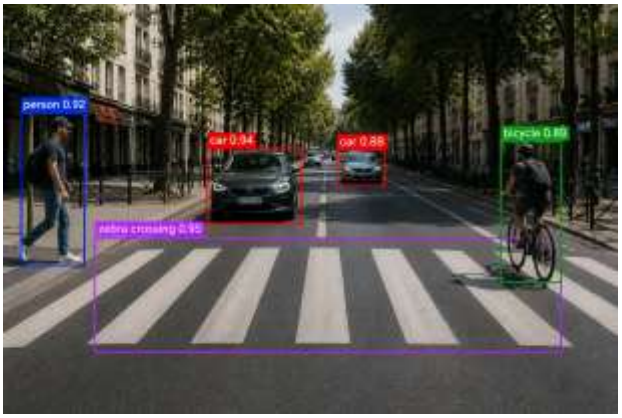

Fig. 2. Sample Object Detection Output

Fig. 2 shows what comes out. The system can flag a person, a car, a pole, all from the same frame, all while the feed keeps rolling. That paints a decent picture of the space nearby.

The model is picked for speed. Lightweight, fast to run. Delay between seeing and hearing stays small.

C. Spatial Reasoning and Directional Feedback

Spotting objects is not the end. The user also needs a sense of where. So the frame is sliced into three vertical strips: left zone, center zone, right zone.

Fig. 3. Zone-Based Spatial Reasoning

Fig. 3 shows the idea. Any box that comes out of detection gets assigned to one strip based on where its center lands sideways.

$$\text{Zone} = \begin{cases} \text{Left}, & 0 \le x < \frac{W}{3} \\ \text{Center}, & \frac{W}{3} \le x < \frac{2W}{3} \\ \text{Right}, & \frac{2W}{3} \le x \le W \end{cases} \tag{1}$$

It is a blunt split. But it works. The user hears ”object on your left” or ”straight ahead” and can react without needing to process much.

D. Distance Estimation

Distance is not measured. It is guessed from box size. The taller the box, the closer the thing inside it. So distance goes as the inverse of box height.

$$D \propto \frac{1}{h} \tag{2}$$

Here D is the rough gap, and h is box height in pixels.

The number it gives is coarse. But the math costs next to nothing. For a pipeline that has to run live, that swap makes sense.

E. Zebra Crossing System Workflow

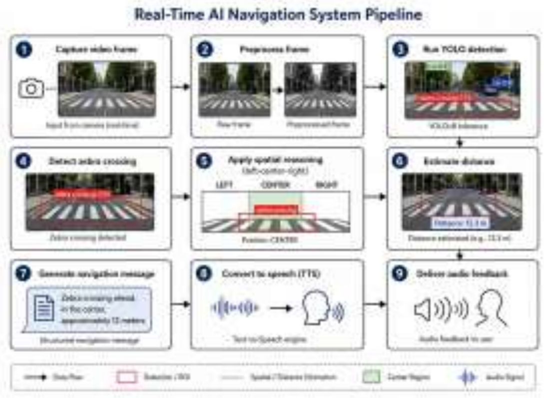

Fig. 4. Zebra Crossing System Processing Pipeline

Fig. 4 traces the whole chain. Frame in. Clean. Detect. Assign zone. Guess distance. Speak. Loop back.

F. Audio Feedback Mechanism

What the system learns gets turned into speech. Object name. Which side. Near or far. That set of three gives the user enough to decide the next move without pausing.

G. Real-Time Implementation

The whole thing hangs on speed. Models stay small. Steps stay simple. The loop spins without breaks: grab, process, speak, repeat. No gaps, no long waits.

IV. RESULTS AND DISCUSSION

We ran the system through a set of checks. Three things we cared about: does it spot stuff, does it give useful direction, and does it do all that fast enough. Tests happened indoors and outdoors, places a person would actually walk.

A. Performance Evaluation

The performance of the proposed system was evaluated in a real-time environment both indoor and outdoor. The main areas tested include whether it could detect obstacles, give directions, and deliver timely feedback to the user.

From the testing, the system can successfully detect normal obstacles such as pedestrian, vehicle, other objects surrounding the user. The system successfully translated these detection points into simple directions (left, center, right) due to integration of spatial reasoning which makes it useful for navigation. The response rate of the system was low, providing near real-time feed back, which allows the user to react quickly without any significant lag. In general, the system works stable during the normal operation. Though not evaluated with formal metric, it seems to perform well from actual operation.

B. Object Detection and Spatial Feedback

The object detection was able to detect common obstacles such as pedestrian, vehicle, other objects around the user in the real-time scenario. The detection is stable and has no significant delay. Spatial reasoning helps convert the detection point into navigational guide of left, center, right so that the user can quickly know what to do. Zebra crossing detector adds navigation important features, contributing to the safety while walking across a road.

C. Distance Estimation Analysis

The system uses a heuristic distance estimation method based on bounding box dimension. This method is computationally cheap, which does not produce a real measured distance but rather gives an estimate distance. It allows the system to categorize obstacles based on their distance for timely detection and alert for proximity but saving computational power.

D. Real-Time System Performance

The system provides a stable real-time performance. The real-time system has a continuously operating pipeline; all frames were captured, processed and delivered to audio feedback without any breaks. The system has an average response time and user would get the navigational message with no apparent delay.

E. Discussion

Looking at the whole thing side by side, the system lands on its feet. Accuracy was good. Speed was good. The mix of detection plus spatial sense plus voice in one go gave a clear edge over older tools that only handle one piece.

That does not mean it is flawless. Light changes mess with it. Going from bright sun to shade, or walking at dusk, the detection dips a bit. And the distance guess is still blunt. Not wrong enough to break the experience, but not sharp either. These are the spots worth fixing next. Better handling of tricky light, tighter distance numbers. That is where the next version should aim.

V. CONCLUSION

This paper put forward an AI-based real-time navigation aid built to help visually impaired individuals move through complex spaces on their own and do so safely. The system pulls together computer vision, spatial reasoning, and voice output to give the user guidance they can actually act on.

A deep learning detection model spots obstacles in live frames and builds a picture of what sits nearby. A zone-based reasoning step takes that picture and turns it into simple directional hints, left, center, right. Distance is estimated from box size, just enough to sort alerts by how close something is.

Test results show the system catches obstacles reliably and runs with delay so low the user does not feel it. That makes it a fit for real-time use. Pulling detection, spatial sense, and voice into one pipeline gives a clear usability bump over older tools that handle only one part of the problem.

All told, what we landed on is a working, practical, and scalable way to help visually impaired users move with more confidence and less risk.

A. Future Work

There is room to build on this. Distance guesses could get sharper by folding in better depth methods that still run light. The detection model would gain from training on bigger, more varied datasets so it holds up under tougher lighting and weather.

New pieces could be added too. Indoor routing. Path planning. Tracking an obstacle across frames. Each of these would widen what the system can do. And beyond bench tests, trials with actual users out in real settings would tell us a lot about how the system feels and where it needs to improve.

REFERENCES

[1] B. Li et al., “Vision-Based Mobile Indoor Assistive Navigation Aid for Blind People,” *IEEE Transactions on Mobile Computing*, vol. 18, no. 3, pp. 702–714, 2018.
[2] S. Ikram et al., “Enhancing Object Detection in Assistive Technology for the Visually Impaired: A DETR-Based Approach,” *IEEE Access*, 2025.
[3] A. Pratap et al., “Adaptive Object Detection for Indoor Navigation Assistance,” *arXiv*, 2025.
[4] A. Jadhav et al., “AI Guide Dog: Egocentric Path Prediction on Smartphone,” *arXiv*, 2025.
[5] N. Pfitzer et al., “MR.NAVI: Mixed-Reality Navigation Assistant for the Visually Impaired,” *arXiv*, 2025.
[6] A. Budrionis et al., “Smartphone-Based Computer Vision Travelling Aids for Blind Individuals,” *Assistive Technology*, vol. 34, no. 2, pp. 178–194, 2020.
[7] B. Kuriakose et al., “DeepNAVI: A Deep Learning-Based Smartphone Navigation Assistant,” *Expert Systems with Applications*, 2023.
[8] R. Litoriya et al., “Implementing Visual Assistant Using YOLO and SSD,” *Journal of Automation, Mobile Robotics*, 2024.
[9] B. Chimwanga, “Object Detection for the Visually Impaired,” *Journal of Pattern Recognition*, 2024.
[10] S. Sameer et al., “AI-Based Object Detection for Assisting Visually Impaired People,” *IEEE ICMCSI*, 2024.
[11] H. An et al., “Mobile-AI-Based Navigation and Localization for Visually Impaired,” *Applied Sciences*, 2025.