

VAVI – Voice Assistant for Visually Impaired

Final Project Report

Berkay Kaan Karaca

Ceyda Kuşçuğlu

Kivanç Terzioglu

1. Introduction

Visually impaired individuals face significant challenges in independent mobility and environmental awareness. Traditional assistive tools such as white canes or guide dogs provide limited contextual information and require extensive training. Recent advances in computer vision, mobile computing, and artificial intelligence enable new assistive technologies that can enhance perception through auditory feedback.

VAVI (Voice Assistant for Visually Impaired) is a senior design project developed within the scope of CMPE 491/492 at TED University. The aim of the project is to design and implement a prototype system that detects people and obstacles using a smartphone camera and provides real-time auditory feedback to assist visually impaired users in navigating their environment safely.

This final report consolidates and summarizes the outcomes of the previous project documents, including the High-Level Design Report, Detailed Design Reports, Multidisciplinary Engineering Analysis Report, and Test Plan Report. It presents the overall system architecture, design decisions, implementation approach, testing strategy, challenges, and future work.

2. Project Objectives and Scope

The primary objective of VAVI is to improve the daily mobility and safety of visually impaired users by providing real-time awareness of nearby people and obstacles. The system is designed as a prototype with a focus on feasibility, usability, and extensibility rather than full-scale deployment.

The main goals of the project are:

- To detect people and obstacles accurately using a deep learning-based object detection model.
- To deliver low-latency auditory feedback that indicates the position of detected objects.
- To use commonly available hardware, primarily a smartphone, without requiring additional wearable devices.

- To design a modular and extensible software architecture that supports future features such as mapping, offline operation, and enhanced privacy controls.

The scope of the project includes real-time image capture, network-based image transmission, server-side object detection, and audio feedback generation. Features such as persistent data storage, full mobile application development, and encrypted communication are considered future enhancements and are not fully implemented in the current prototype.

3. System Overview

VAVI follows a client–server architecture. The client side is the user’s smartphone, which captures images from the environment and plays audio feedback. The server side is a local computer that processes incoming images using a YOLO-based object detection model.

In the current prototype, images are transmitted over a local Wi-Fi network to minimize latency. The server analyzes each frame, identifies people and obstacles, determines their relative positions, and generates corresponding auditory feedback such as directional speech or beeping sounds. The feedback is then sent back to the smartphone and played through the device’s speaker.

The system is intentionally designed without persistent storage. All processing is done in real time, and no images, audio, or user data are saved, supporting a privacy-by-design approach.

4. High-Level Architecture

The high-level architecture of VAVI has evolved from traditional sensor-based navigation approaches toward a more robust and deterministic **visual-based localization and navigation pipeline**. The system is designed to minimize dependency on unstable indoor signals and cumulative sensor errors while maintaining real-time responsiveness and user safety.

At a high level, the VAVI system consists of the following main subsystems:

1. **Client Subsystem (Smartphone)**
 - Captures a photograph of the surrounding indoor environment upon user request.
 - Sends the captured image to the AI processing module.
 - Receives navigation and obstacle-related audio feedback.
 - Plays directional and descriptive audio cues to guide the user.
2. **AI-Based Visual Localization Subsystem**
 - Processes the captured image using a deep learning–based visual feature extraction model.
 - Matches extracted features against a pre-built indoor visual map.
 - Estimates the user’s current location as a discrete node within the indoor graph representation.
3. **Navigation and Path Planning Subsystem**
 - Uses the identified starting node and the user-selected destination.

- Computes the optimal and safest route using graph-based search algorithms such as A*.
 - Produces an ordered sequence of navigation instructions.
- 4. Perception and Obstacle Detection Subsystem**
- Continuously analyzes camera input using a YOLO-based object detection model.
 - Detects nearby people and obstacles in real time.
 - Generates immediate auditory alerts that take priority over navigation guidance.
- 5. Audio Feedback and Interaction Subsystem**
- Converts navigation steps and detection results into intuitive auditory feedback.
 - Uses directional cues and distance-based modulation to convey spatial information.

This architecture intentionally avoids continuous localization updates. Instead, it relies on **single-shot visual localization** followed by deterministic path planning. This design reduces system complexity, eliminates sensor drift, and improves reliability in safety-critical indoor environments.

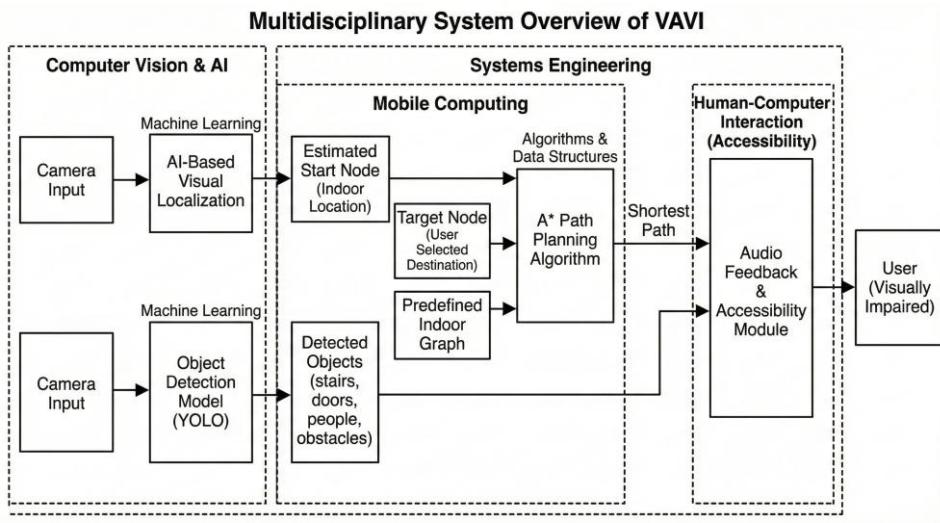
5. Detailed Design

5.1 Software Components

The software design decomposes the system into clearly defined components:

- Image Capture Module (Client)
- Network Communication Module (Client & Server)
- Object Detection Module (Server)
- Feedback Generation Module (Server)
- Audio Playback Module (Client)

The object detection module is implemented using the YOLO algorithm due to its real-time performance and high detection accuracy. Python is used as the primary programming language because of its strong ecosystem for machine learning, image processing, and rapid prototyping.



5.2 Data Flow

1. The user activates the system.
2. The smartphone captures an image frame.
3. The image is transmitted to the server.
4. The server processes the image and detects objects.
5. The server generates feedback based on object position.
6. The feedback is sent back to the smartphone.
7. The smartphone plays the audio feedback to the user.

This event-driven flow ensures minimal delay between perception and feedback, which is critical for user safety.

6. Navigation and Indoor Guidance Module

In addition to obstacle detection, VAVI is designed to support indoor navigation and route guidance for visually impaired users. Although the primary prototype focuses on perception and real-time feedback, a navigation subsystem has been designed and partially implemented as described in the High-Level and Detailed Design Reports.

6.1 Navigation Objective

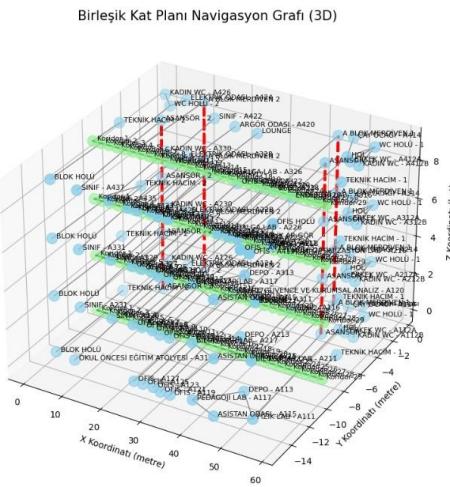
The goal of the navigation module is to guide users safely from their current indoor location to a selected destination (such as a classroom, elevator, or exit) using auditory instructions. Since GPS signals are unreliable indoors, the system relies on indoor maps and graph-based path planning rather than traditional outdoor navigation methods.

6.2 Indoor Map Representation

Indoor environments are modeled as a **graph-based structure** derived from authorized building floor plans:

- **Nodes** represent key locations such as corridor intersections, rooms, staircases, and elevators.
 - **Edges** represent walkable paths between nodes and store distance and traversal cost information.

This representation enables efficient computation of optimal paths while allowing the system to adapt to accessibility constraints (e.g., avoiding stairs if necessary).



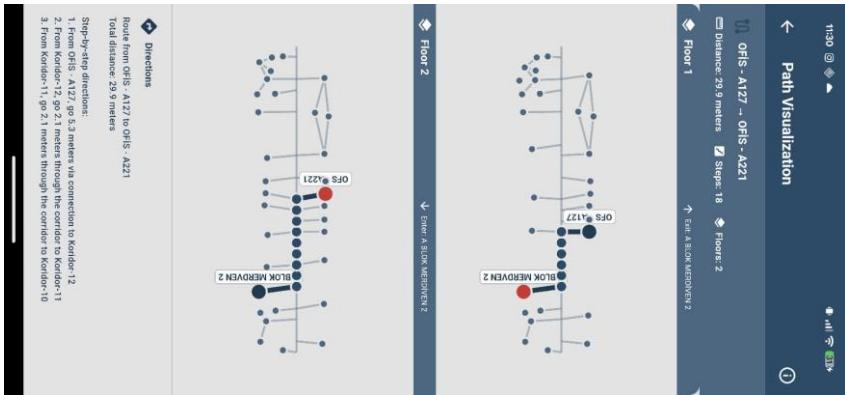
6.3 Visual-Based Localization Strategy

Instead of relying on continuous sensor-based localization such as GPS, Wi-Fi, or dead reckoning, VAVI adopts a **visual localization approach** to determine the user's current indoor position.

In this approach, the user captures a single photograph of the surrounding environment using the smartphone camera. This image is processed by an artificial intelligence model that extracts visual features from the scene and compares them against a pre-built visual representation of the indoor environment. By matching the captured image with known locations, the system estimates the user's current position as a corresponding node within the indoor navigation graph.

This design choice was made after observing that traditional indoor localization techniques suffer from instability, sensor drift, and environmental variability. Visual localization provides a more robust and deterministic method for initial positioning, especially in structured indoor spaces such as university buildings.

Once the user's location is identified, continuous re-localization is not required. The system proceeds with path planning and guidance based on the detected starting node, reducing computational complexity and minimizing cumulative localization errors.



6.4 Path Planning Algorithm

Once the user's starting node and target destination are determined, the system computes the shortest and safest path using classical graph search algorithms such as **Dijkstra** or **A***. Edge weights can be adjusted to prioritize safety over distance, which is critical for visually impaired users.

6.5 Navigation Guidance and Feedback

Navigation instructions are delivered through step-by-step auditory guidance:

- Directional commands (e.g., “turn left”, “go straight”).
- Distance-based updates when approaching the next node.
- Alerts when reaching important landmarks or destination points.

The navigation module operates concurrently with the object detection module, ensuring that obstacle alerts always take priority over route instructions for user safety.

6.6 Integration with Obstacle Detection

Navigation and object detection are designed as parallel subsystems. While navigation provides global route guidance, the perception module handles immediate hazards. Detected obstacles can temporarily override navigation instructions, ensuring real-time responsiveness and safety.

This layered guidance strategy combines **global path planning** with **local obstacle awareness**, resulting in a robust and user-centered navigation experience.

7. Multidisciplinary Engineering Considerations

The VAVI project integrates concepts from multiple engineering disciplines:

- **Computer Engineering:** software architecture, networking, and system integration.
- **Artificial Intelligence:** deep learning-based object detection.
- **Human–Computer Interaction:** auditory feedback design and usability for visually impaired users.

- **Ethics and Privacy:** real-time processing without data storage to protect user privacy.

Design decisions were made with accessibility and safety as primary concerns. Auditory feedback was chosen as the main interaction modality to avoid overloading users and to align with the needs of visually impaired individuals.

7. Testing and Evaluation

A structured test plan was developed to validate system functionality and performance. The tests focused on:

- Correct detection of people and obstacles.
- Accuracy of directional feedback.
- System responsiveness and latency.
- Stability of communication between client and server.

Testing was conducted in controlled indoor environments. The results showed that the system can successfully detect common obstacles and provide timely feedback. However, performance may degrade under poor lighting conditions or unstable network connectivity.

8. Limitations and Challenges

Several limitations were identified during development:

- Dependency on continuous Wi-Fi connectivity between client and server.
- Lack of encryption and authentication in the current communication model.
- Limited robustness in cases of camera obstruction or server overload.
- Absence of a fully developed mobile application interface.

These limitations are primarily due to time and scope constraints typical of a senior design project and are addressed as future work.

9. Future Work

Future improvements planned for VAVI include:

- Native mobile application development for Android and iOS.
- On-device (edge) object detection to eliminate server dependency.
- Secure communication with encryption and authentication.
- Offline operation and fallback mechanisms.
- Integration of mapping and navigation features for indoor environments.
- Enhanced audio feedback personalization.

These enhancements aim to transform the prototype into a more robust and deployable assistive system.

10. Conclusion

VAVI demonstrates the feasibility of using modern computer vision and mobile technologies to assist visually impaired individuals in real time. The project successfully delivers a working prototype that detects people and obstacles and provides intuitive auditory feedback.

Through a modular architecture, multidisciplinary design approach, and user-centric focus, VAVI lays a strong foundation for future development. While the current system has limitations, the project achieves its primary objectives and provides valuable insights into the design of accessible AI-driven assistive technologies.

References

1. Bruegge, B., & Dutoit, A. H. *Object-Oriented Software Engineering: Using UML, Patterns, and Java*.
2. Redmon, J., & Farhadi, A. (2018). YOLOv3: An Incremental Improvement.
3. Python Official Documentation.
4. PyTorch / TensorFlow Documentation.
5. Android Developers Documentation.