

# Assistive Vision

---

Assistive Vision is an iOS application designed to empower visually impaired individuals through real-time object detection and audio feedback. The project leverages a custom YOLOv11m model—heavily influenced and inspired by the Ultralytics YOLO11 architecture—to provide fast, accurate, and context-aware detection, all executed locally on the device to ensure user privacy.

## Project Overview

- **Objective:**

Enhance the independence of visually impaired users by recognizing objects in real time and delivering natural-sounding audio descriptions via Apple's Text-to-Speech (TTS) API.

- **Key Features:**

- Real-time object detection using the iPhone's built-in camera.
- On-device processing with a custom YOLOv11m model influenced by Ultralytics YOLO11.
- Integration with iOS accessibility tools such as VoiceOver and Siri Shortcuts.

## Environment Configuration

### Prerequisites

- macOS 13.0 or later
- Xcode 15.0 or later
- iOS 16.0 or later
- iPhone with A12 Bionic chip or later (for optimal performance)
- Apple Developer Account

### Development Environment Setup

1. Install Xcode from the Mac App Store
2. Install CocoaPods: `sudo gem install cocoapods`
3. Clone the repository
4. Run `pod install` in the project directory
5. Open the `.xcworkspace` file in Xcode
6. Configure your Apple Developer account in Xcode preferences

# Dependencies

## Core Dependencies

- Swift 5.9
- iOS 16.0+
- CoreML 5.0+
- AVFoundation
- Vision
- Speech

## Third-Party Libraries

- CocoaPods (Dependency Manager)
- YOLOv11m (Custom model)
- SwiftUI
- Combine
- Swift-testing
- Swift-syntax
- generative-ai-swift
- Get
- SwiftSpeech

## Development Tools

- Xcode 15.0+
- CocoaPods 1.12.0+
- Git
- macOS 13.0+

# Technologies

- **Hardware:**
  - **iPhone:** Utilizes the built-in camera, microphone, speakers, and Apple's Neural Engine for on-device processing.
  - **Audio Devices:** Supports both built-in speakers and Bluetooth-connected headphones or hearing aids.
  - **Battery Optimization:** Implements Battery Saver mode to adjust frame capture rates and reduce power consumption.

- **Software:**

- **Swift & SwiftUI:** For developing the application and designing an accessible user interface.
- **CoreML:** For running the custom YOLOv11m model locally, ensuring fast and private object detection.
- **AVFoundation:** Manages live camera feed processing and synchronizes audio output.
- **Text-to-Speech (TTS) API:** Converts detection results into clear, natural-sounding audio descriptions.
- **iOS Accessibility Features:** Leverages VoiceOver, Siri Shortcuts, and Dynamic Type to ensure ease of use for visually impaired users.

## Credentials and Configuration

### Required Credentials

- Apple Developer Account credentials
- Bundle Identifier: com.assistivevision.app
- Team ID: [Your Team ID]
- Provisioning Profile: AssistiveVision\_Development

### API Keys and Configuration

- Gemini API key required (obtain from <https://ai.google.dev/gemini-api/docs/api-key>)
- Store the API key in Environment.swift file
- No other external API keys required (all other processing is done locally)
- CoreML model is bundled with the app
- Text-to-Speech uses native iOS APIs

## Team Members

- **Venkat Yenduri** – Team Leader, Communications Lead, Software Developer
- **Pierre Tawfik** – Scribe, Maintenance Lead, Software Developer Lead
- **Sukrut Nadigotti** – Documentation Lead, Presentation Lead, Software Developer
- **Sharefa Alshaary** – Quality Assurance Lead, Software Developer

## License

Assistive Vision is distributed under the [AGPL-3.0 License](#). This open-source license is designed to ensure that all modifications and derivatives remain free and available to the community. Key points

include:

- **Copyleft Requirement:** Any derivative work must also be distributed under the AGPL-3.0 License.
- **Source Availability:** Modifications and extensions to the project must be made available to the public.
- **Academic and Collaborative Use:** The license promotes sharing and collaboration, making it ideal for academic projects and community-driven development.