DATA SCIENCE & AI LAB (BSCSS3001)

MILESTONE - 1

GROUP NO. 2

PRASHASTI SARRAF **(21f1001153)**
TANUJA NAIR **(21f1000660)**
BALASURYA K **(22f3002744)**
KARAN PATIL **(22f2001061)**
JIVRAJ SINGH SHEKHAWAT **(22f3002542)**

IITM BS Degree Program
Indian Institute of Technology,
Madras, Chennai,
Tamil Nadu, India, 600036

Vision Assist: Real-Time Navigation Support for the Visually Impaired

1. Problem Definition

1.1 The Challenge

Visually impaired individuals face persistent difficulties in navigating dynamic and unstructured environments such as urban streets, public transport areas, and crowded indoor spaces. Obstacles include pedestrians, vehicles, stairs, escalators, uneven pavements, and street furniture. Traditional aids (white canes, guide dogs) provide limited tactile feedback and cannot reliably warn about overhanging obstacles, fast-moving objects, or distant hazards.

Existing mobile apps like **Seeing AI** and **Be My Eyes** rely heavily on cloud-based processing, introducing latency and connectivity dependence, making them less usable in real-time navigation contexts. Smart canes or wearables (OrCam, Envision Glasses) are expensive and hardware-dependent, limiting accessibility.

1.2 Proposed Solution

We propose a **deep learning–powered, real-time navigational aid** that performs **object detection and segmentation** on a live video feed and provides **context-aware, distance-aware audio navigation cues**.

Core Pipeline: Live video input → YOLOv8-seg (fine-tuned) → object masks & bounding boxes → concise natural language audio cues via a lightweight TTS engine.

Deployment: Web/mobile demo (Hugging Face Spaces + Gradio) for portability, accessibility, and ease of use.

Optimization: Quantization (FP16/INT8) and pruning will ensure ~15–20 FPS with <1s latency, enabling near real-time usability.

This solution aims to deliver affordable, scalable, and user-friendly navigation support without specialized hardware.

2. Project Objectives

- **Accessibility:** Build a cost-effective assistive tool usable on smartphones or laptops.

- **Real-Time Object Detection:** Implement YOLOv8-seg lightweight variant capable of identifying navigation-critical obstacles (e.g., pedestrians, vehicles, traffic lights, benches, barriers, stray animals). SSD and Faster R-CNN can also be tested to compare accuracy and speed trade-offs.

- **On-Device Processing:** Ensure the model runs with low latency on edge devices or simulated environments, reducing cloud reliance.

- **Context-Aware Audio Output:** Provide natural-language navigation cues (e.g., “Person ahead, slightly left, 2m”) via a simple template-based or lightweight generative TTS pipeline.

- **Prototype Deployment:** Deliver a live web demo (Hugging Face Space) demonstrating the full pipeline.

- **Extensibility:** Create an open-source modular framework (ONNX-exported, Gradio-based) extendable to indoor navigation, AR overlays, or landmark recognition.

- **User Testing:** Conduct pilot evaluations with 3–5 visually impaired users to refine usability and audio feedback strategies.

3. Implementation Feasibility (Model & Approach)

- **Model Choice:** YOLOv8-seg (lightweight variant) is the primary model due to its balance of speed, segmentation accuracy, and mobile suitability. Feature Pyramid Networks (FPN) improve small-object detection. Pixel-accurate masks allow estimation of obstacle shape/size.

- **Other Candidates:** SSD and Faster R-CNN will be tested to evaluate trade-offs in speed and accuracy, but YOLOv8-seg remains the baseline.

- **Customization & Optimization:**

    - Transfer learning using COCO-pretrained weights.
    - Dataset-specific augmentations: motion blur, rotation, low-light conditions, occlusion.
    - Quantization (FP16/INT8) and pruning for deployment efficiency.
    - Training estimated ~30–40 GPU hours on Colab/Kaggle Pro.
- **Deployment:** Hugging Face Spaces + Gradio for browser/mobile accessibility.

4. Data Sourcing and Governance

- **Base Dataset:** MS COCO (~118k training, ~5k validation, 80 classes with segmentation masks & bounding boxes).
- **Extended Dataset:** Annotated Creative Commons YouTube frames to capture diverse real-world scenes.
- **Supplementary Datasets:** Experimentation with additional open-source datasets such as Pascal VOC, ILSVRC (ImageNet), Objects365, Open Images Dataset (OID) to improve coverage of navigation-critical objects and urban scenarios.
- **Custom Dataset:** ~2k–3k images covering navigation-specific obstacles (stairs, escalators, tactile tiles, local urban barriers).
- **Augmentations:** Rotation, lighting changes, blur, occlusion, and perspective shifts to simulate real-world variability.
- **Governance:** Licensing-compliant and carefully annotated datasets, with short documentation covering sourcing, annotation quality, and compliance.

5. User Experience Design

- **Audio Guidance:** Short, directional, distance-aware cues (e.g., “Person ahead, slightly right, 2m”).

- **Latency Target:** <1s per frame.
- **Pilot Study:** 3–5 visually impaired users tested in indoor/outdoor/stairs scenarios.
- **Feedback Loop:** Iterative refinement of audio cues based on pilot study recordings and structured feedback.

6. Literature Review

6.1 Existing Assistive Solutions

- **Seeing AI (Microsoft):** Scene description; cloud-dependent → high latency.
- **Be My Eyes:** Human volunteers provide assistance → not autonomous.
- **Aipoly Vision:** On-device classification only; lacks obstacle detection/avoidance.
- **Smart Canes / Wearables (OrCam, Envision Glasses):** Expensive and hardware-dependent.

Drawback: Most solutions are costly or lack real-time, context-aware obstacle avoidance.

6.2 Baseline Models & Benchmarks

- **SSD:** Good at small objects; slower in real-time deployment.
- **Faster R-CNN:** High accuracy; computationally heavy.
- **YOLOv5/v8:** Strong speed-accuracy balance; YOLOv8 adds segmentation + FPN.
- **DeepLabV3+:** Excellent segmentation but not real-time.

Metrics: mAP, FPS, Precision/Recall.

6.3 Generative AI Baseline (Optional)

Rule-based template (object + distance + direction) compared against lightweight generative approaches for audio narration; full LLM use avoided to minimize latency.

7. Gaps and Opportunities

Gaps:

- **Latency Gap:** Cloud-based solutions introduce delays.
- **Context Gap:** Existing models identify objects but do not provide actionable navigation cues.
- **Dataset Gap:** COCO lacks navigation-specific classes (stairs, escalators, tactile tiles).
- **Scalability Gap:** Closed-source systems are non-extensible.

Opportunities:

- Real-time, edge-based object segmentation with context-aware speech.
- Fine-tune with locally relevant datasets → robust in dynamic urban environments (e.g., Chennai).
- Open-source, extensible framework deployable across platforms.
- Enable future extensions (indoor navigation, AR-based overlays, landmark recognition).

8. Expected Contributions & Impact

- **Technical:** Feasibility of real-time segmentation + navigation guidance on mobile/web.
- **Societal:** Affordable, scalable alternative to smart canes/wearables.

- **Novelty:** YOLOv8-seg integrated with context-aware TTS into a modular, extensible pipeline.

9. References & Resources

- **Datasets:** MS COCO, Open Images, etc
- **Literature:**
    - *A Deep Learning Based Model to Assist Blind People in Their Navigation* (ResearchGate)
    - *Assistive Deep Learning Solutions* (ScienceDirect)
- **Apps:** Seeing AI, Be My Eyes, Aipoly Vision
- **Video Demos:** YouTube explainers on YOLO-based assistive navigation
  - A Device for Blind And Visually Impaired People
  - How Deep Learning Is Helping Blind People with Karthik Kannan, Founder o…
  - Smartglasses Use ChatGPT To Help The Blind And Visually Impaired | 5G Pla…
  - AI based visual assistance system for the visually impaired