

# NAVAID: AI-POWERED NAVIGATION ASSISTANT FOR MODERATELY SEVERE VISUAL IMPAIRMENT

**Prabhav Singh, Sandesh Rangreji, Mukund Iyengar, Anubhav De**

Department of Computer Science

Johns Hopkins University

Baltimore, MD 21218, USA

{psingh54, srangre1, miyenga2, ade11}@jh.edu

## ABSTRACT

Over 70% of people with vision loss identify unsafe navigation as their primary barrier to independence, yet existing assistive technologies remain either prohibitively expensive, privacy-invasive, or functionally limited. We present NAVAID, an AI-powered iOS navigation assistant specifically designed for adults with Moderately Severe Visual Impairment (MSVI). Combining multimodal large language models (Google Gemini 2.5 Flash/Pro) with accessibility-first interface design, NAVAID provides real-time hazard detection, personalized turn-by-turn guidance, and conversational scene understanding through audio-haptic feedback. Our system addresses the unique challenges of peripheral vision loss conditions like Retinitis Pigmentosa by adapting instructions to individual vision profiles, mobility aids, and environmental contexts. Real-world blindfolded navigation trials across 4.31 km on JHU campus demonstrated a 1.3% Critical Hazard Miss Rate with 16 completed trips. We identify key limitations in automated crossing assistance and propose hybrid manual-automatic modes to preserve user agency. NAVAID demonstrates that vision-language models, when coupled with thoughtful personalization and accessibility design, can enable safe independent navigation for MSVI users.

## 1 INTRODUCTION AND PROBLEM STATEMENT

Over 70% of people with vision loss report that unsafe navigation is their biggest daily barrier to independence, according to the American Foundation for the Blind’s 2022 survey. This affects 1.1 billion people worldwide who live with vision impairment, including 7 million Americans with Moderately Severe Visual Impairment (MSVI) and 1 million who are blind. The economic impact in the United States alone reaches \$134.2 billion annually in medical costs, lost productivity, and educational delays. Over 1.6 million Americans with MSVI are under 40 years old—this is not just an aging population issue, but one affecting students and young professionals.

People with MSVI face three times higher unemployment rates compared to sighted populations, three times greater risk of motor vehicle collisions as pedestrians, two times increased likelihood of falls, and three times elevated rates of depression and anxiety. These are barriers to human dignity and autonomy, not just technological problems.

Traditional assistive technologies have failed to address this crisis adequately. White canes and guide dogs—the gold standard for over a century—detect obstacles only at contact distance, providing no advance warning of stairs, construction zones, or traffic lights. Human-assisted services like Aira and Be My Eyes offer real-time remote guidance via volunteers or paid agents, but they raise critical privacy concerns (strangers viewing live camera feeds of personal environments), incur high subscription costs (\$89–\$329 per month for Aira), and cannot provide continuous coverage throughout the day. Navigation apps like Google Maps excel at macro-level routing but operate at 5–10 meter GPS accuracy, making them unusable for micro-navigation challenges like detecting curb cuts, puddles, or crosswalk signal states. Object detection apps such as Microsoft Seeing AI recognize text and objects but are not designed for outdoor navigation; they complement rather than replace mobility aids.

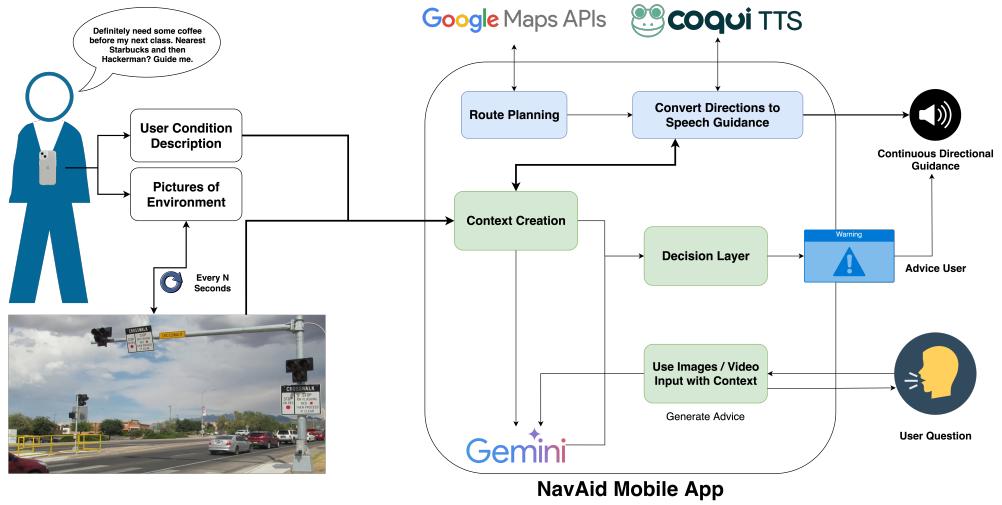


Figure 1: NavAid: AI Powered Navigation Assistant for MSVI.

None of these solutions address the heterogeneity of visual impairments. Retinitis Pigmentosa causes peripheral vision loss (tunnel vision), requiring different guidance strategies than Age-Related Macular Degeneration (central vision loss) or Nystagmus (involuntary eye movement). Generic approaches fail to account for individual vision profiles, mobility aid usage, color blindness, or navigation environments.

Multimodal large language models offer new capabilities for this problem. Google Gemini 2.5 can analyze street scenes in real-time, distinguish hazardous obstacles from benign street traffic, and generate descriptions tailored to specific vision conditions. Unlike traditional object detection systems that flag all detected objects indiscriminately, AI can reason about spatial relationships—for example, instructing a user that a utility box is on the right side of the path two meters ahead and suggesting they use their white cane to locate and pass to its left. AI can automatically generate colorblind-friendly UI palettes, inject user profiles into prompts, and adapt the verbosity and specificity of instructions without requiring manual rule engineering for every possible vision condition. Finally, vision-language models enable users to ask natural language questions about their surroundings and receive spatially-aware answers, mimicking human visual assistance without the privacy invasion of human-assisted services.

NAVAID uses these capabilities to build a personalized, privacy-preserving navigation system for adults with moderate-to-severe visual impairment who rely on audio guidance for independent navigation.

## 2 INTERFACE DESIGN

The design of NAVAID’s user interface starts with a core principle: accessibility is not a feature to add later—it is the foundation. Every interaction, visual element, and feedback mechanism was designed for users who cannot rely on visual cues. Traditional mobile interfaces assume sighted users who can precisely tap small targets, read fine print, and perceive color-coded warnings. For MSVI users, these assumptions break down. Our interface combines voice-driven onboarding, thumb-friendly layouts, VoiceOver integration, WCAG AAA compliance, spatial haptic feedback, and personalization tailored to individual vision conditions.

### 2.1 ACCESSIBILITY-FIRST DESIGN PRINCIPLES

We replaced traditional text-based forms with an 8-question conversational survey that serves as the foundation for our personalization engine (Fig. 2). Each question is spoken aloud via text-to-speech, and users record their answer using a single large Answer button sized at 160pt by 160pt and positioned in the lower half of the screen for one-handed thumb reach. This pattern eliminates

the need for keyboard interaction, zoom gestures, or precise tap targets. Questions can be repeated on demand, and users may skip to pre-configured defaults if they prefer not to answer a particular question. The survey asks for the user’s name, age, specific vision problems (Retinitis Pigmentosa, Macular Degeneration, Diabetic Retinopathy, Nystagmus, Scotoma, Night Blindness, or other conditions), assistive devices used for navigation (white cane, guide dog, magnifying glasses, or none), color blindness type if applicable, usual navigation environments (urban streets, suburban areas, indoor spaces, or rural paths), most common navigation challenges (crossing streets, detecting obstacles, reading signs, navigating stairs, or finding entrances), and an emergency contact for the SOS feature. Responses are transcribed via OpenAI Whisper Base with 74 million parameters and median latency of 680 milliseconds, then stored locally on-device to preserve privacy.

All text in the application meets the highest Web Content Accessibility Guidelines standard of WCAG AAA compliance: 18-point minimum font size for body copy, 7:1 contrast ratios for all foreground and background color pairs, and no reliance on color alone to convey critical information—every warning is reinforced with both haptic and audio feedback. Button sizes exceed the recommended 44-point minimum, with most interactive elements sized at 160 points in height to ensure they are easily tappable even for users with limited vision and motor control challenges.

Every UI element includes descriptive labels for iOS’s native VoiceOver screen reader. During development, we tested all workflows with VoiceOver enabled to ensure logical navigation order and clear semantic descriptions. For example, the Start Trip button announces: “Start Trip. Button. Double-tap to begin navigation with audio guidance and hazard detection.” This level of attention to screen reader compatibility ensures that users who cannot see the screen at all can still use every feature of the application.

All interactive elements are positioned in the lower half of the screen, within 160 points of the bottom edge, to accommodate one-handed use while the user is simultaneously holding a white cane or managing a guide dog leash. The home screen features just two large buttons—Start Trip (green, 160pt by 180pt) and Scene Understanding (purple, 160pt by 180pt)—eliminating visual clutter and reducing cognitive load during what are often stressful navigation scenarios.

NAVAID leverages iOS’s Taptic Engine to provide directional haptic cues that offload spatial information from the already-burdened auditory channel. When a hazard is detected on the user’s left side, the phone vibrates with a left-biased haptic pattern; right-side hazards trigger right-biased vibrations. Intensity scales with urgency: two gentle taps for static obstacles like utility boxes or bike racks, versus four strong pulses for dynamic hazards like approaching vehicles or pedestrians. This spatial haptic feedback allows users to immediately understand the bearing and severity of hazards without waiting for audio descriptions, reducing cognitive burden during critical navigation moments.

## 2.2 INTERFACE EVOLUTION THROUGH USER TESTING

The current interface design emerged through three iterations driven by blindfolded pilot testing and user feedback. Our initial prototype featured a 3-button home screen with Start Trip, Scene Understanding, and Settings buttons. Early trials revealed that users struggled to locate the Settings button during navigation while holding a white cane. We simplified to a 2-button minimalist layout, moving all configuration to the pre-trip voice survey.

Initial haptic feedback provided only a single generic vibration pulse regardless of hazard location or severity. Blindfolded testers reported difficulty distinguishing urgent warnings from routine notifications. We implemented directional haptic patterns (left, right, center) with intensity scaling (gentle taps for static obstacles versus strong pulses for dynamic hazards), which post-trial interviews identified as “super useful” with 100% of participants praising the directional cues.

The first design used generic purple and green button colors optimized for typical color vision. When we added a deutanopia test persona, we discovered these colors appeared nearly identical. Rather than manually selecting alternative palettes for every color vision deficiency type, we implemented AI-generated colorblind-friendly palettes where Gemini produces 5 distinct colors validated for RGB distance and WCAG AAA contrast ratios.

Deep Analyze for traffic light detection initially required manual button press, forcing users to remember to trigger analysis before every crossing. We automated the trigger based on crosswalk

Feature	Initial Design	Current Design	Rationale
Home Screen Buttons	3 buttons: Start Trip, Scene Understanding, Settings	2 buttons: Start Trip, Scene Understanding	Users struggled to locate Settings during navigation with white cane. Moved configuration to voice survey.
Haptic Feedback	Single generic vibration pulse	Directional (left/right/center) + intensity scaling (2 taps vs 4 pulses)	Testers could not distinguish urgent vs routine warnings. 100% praised directional cues.
UI Color Palette	Generic purple/green for typical vision	AI-generated friendly palettes (5 colors, RGB distance $\leq 60$ )	Deuteranopia persona saw purple/green as identical. AI generation eliminates manual palette design.
Deep Analyze Trigger	Manual button press only	Hybrid: Auto-detect + manual trigger ("Press when ready")	Fully automatic caused anxiety ("forced to trust"). Manual-only caused cognitive burden to remember.
SOS Activation	Manual button press only	Gyroscope auto-trigger (2.5g threshold) + manual button	Users disoriented after falls cannot locate button. Auto-trigger via G-force detection solves this.

Table 1: Interface design evolution from initial prototype through three iterations of blindfolded user testing. Each change addressed specific usability failures identified during trials.

detection, but this caused extreme user anxiety during trials despite 100% technical accuracy. One tester reported feeling "forced to trust the system" with "no control over when crossing analysis started." We adopted a hybrid approach: automatic crosswalk detection announces "Crosswalk ahead—press Deep Analyze when ready to check signal," preserving user agency while reducing cognitive load.

The SOS feature originally required users to locate and press a dedicated button during emergencies. Blindfolded testing revealed this was impractical when disoriented or fallen. We added automatic SOS triggering via gyroscope sensor: when the device detects a sudden change in G-force consistent with a fall (acceleration threshold of 2.5g), it automatically sends an emergency SMS to the user's designated contact with GPS coordinates and a pre-configured message. Manual SOS via button press remains available for non-fall emergencies.

Table 1 in the appendix summarizes these design changes with rationale and user feedback.

### 2.3 PERSONALIZATION FOR MSVI: THE CORE INNOVATION

Generic navigation instructions assume either fully sighted users or homogeneous blindness. NAVAID's personalization engine recognizes that Retinitis Pigmentosa (tunnel vision) requires different guidance than Macular Degeneration (central vision loss). This is our core contribution: vision-condition-specific personalization that adapts the user experience.

User profile data collected during the 8-question survey is injected into Google Gemini 2.5 Flash’s system prompt during every navigation request. Table 2 illustrates how the same visual scene—a street corner with tactile paving and a utility box—yields radically different guidance for three distinct user personas.

User Persona	Personalized Guidance for Same Scene
<b>Retinitis Pigmentosa user with white cane</b>	"You are at a curb cut with tactile paving. Ahead, a utility box is on the right side of the path, about 2 meters away. Use your white cane to locate it and pass to its left. Further ahead on your right, a light pole with signs is present; be aware of it as you approach, as it falls outside your peripheral vision." Haptic: right_haptic
<b>Macular Degeneration user without walking stick</b>	"Margaret, directly ahead you are approaching yellow tactile paving, which marks a crosswalk for the street to your left. Immediately after the paving, about 2 meters ahead on your right, there is a low concrete utility box at knee-to-waist height. The sidewalk also has some uneven sections and cracks in the pavement. Be aware of potential glare from the low sun to your left." Haptic: right_haptic
<b>Nystagmus user with guide dog and stick</b>	"Carlos, be aware of a grey utility box on your left, about 2 meters ahead, just past the yellow tactile paving. This is a low-lying obstacle that could be in a blind spot. Luna will help you navigate around it, but keep your cane extended to the left to confirm its position. Further ahead on your left, about 8–10 meters, is a light pole with signs." Haptic: left_haptic

Table 2: Personalized navigation guidance for the same street scene (see Fig. 8). The Retinitis Pigmentosa user receives explicit references to peripheral vision loss and white cane usage instructions. The Macular Degeneration user gets color cues (yellow tactile paving) and glare warnings appropriate for central vision defects. The Nystagmus user receives guide dog commands (addressing Luna by name) and blind spot warnings. Haptic bearing also adapts to obstacle location relative to the user’s heading.

The personalization operates across multiple dimensions. For users with peripheral vision loss like Retinitis Pigmentosa, instructions proactively warn about obstacles “outside peripheral vision” and direct users to sweep their white canes laterally to detect hazards they cannot see. For users with central vision loss like Age-Related Macular Degeneration, instructions include detailed color and texture descriptions—such as the color of tactile paving or warnings about glare—since these users retain peripheral awareness and benefit from high-contrast visual cues. Instructions explicitly reference the user’s mobility aid: white cane users receive cane usage instructions like “extend cane to left to confirm position,” while guide dog users receive forward commands addressing the dog by name. Users who report difficulty crossing streets receive more detailed crossing instructions including signal state, countdown timers, and pedestrian activity, while users with night blindness receive proactive warnings about lighting conditions and shadows. All spatial references use the user’s body frame (“on your immediate right”) rather than cardinal directions, since visually impaired users struggle with compass-based navigation when they cannot see landmarks.

During profile setup, if users report color blindness, NAVAID prompts Google Gemini to generate a 5-color palette optimized for their specific color vision deficiency (Fig. 3). The prompt specifies: “Generate 5 distinct colors visible to [deutanopia/protanopia/tritanopia] users, with RGB distance greater than 60 between all pairs, suitable for button backgrounds with black text at WCAG AAA contrast ratio.” Gemini returns JSON with hex color codes; the app validates uniqueness and contrast ratios, then applies the palette to all interactive elements. This eliminates the need for manual color selection and ensures buttons remain distinguishable for colorblind users without requiring UI designer expertise.

#### 2.4 SCENE UNDERSTANDING: CONVERSATIONAL ENVIRONMENTAL EXPLORATION

While turn-by-turn navigation provides directional guidance to reach a destination, NAVAID’s Scene Understanding feature offers spatial awareness—a critical gap in existing assistive tools that focus solely on wayfinding. Users can activate Scene Understanding at any time during navigation or

when stationary. The workflow begins when users either capture a live photo using the device camera or skip to use a representative default scene image (Fig. 4). After photo capture, users can record a spoken question such as "Are there any shops near me where I can buy groceries?" or skip to receive a general scene description. The recorded audio is transcribed via Whisper, and both the image and transcribed question are sent to Gemini 2.5 Flash for analysis. Gemini generates a natural language answer that is synthesized via text-to-speech and spoken to the user.

An example interaction demonstrates the feature's utility. When a user asked, "Are there any shops near me from where I can buy some groceries from my apartment? Thank you," NAVAID responded: "You're on a commercial street with shops on your right and parked cars on your left. There is a University Market and Deli approximately 30 meters ahead on your right, which may have groceries. Be aware of two people walking toward you about 8 meters ahead on the sidewalk." The median end-to-end latency from camera capture through Whisper transcription, Gemini vision inference, and TTS synthesis is 2.8 seconds. This latency is acceptable for Scene Understanding because users initiate queries at their discretion, unlike real-time hazard detection where sub-second response times are safety-critical.

This feature enables users to mentally map their environment, locate points of interest like accessible entrances or nearby businesses, and make informed navigation decisions without relying on human volunteers. It represents a privacy-preserving alternative to human-assisted services: the AI sees only what the user chooses to photograph at the moment they choose to ask, and no human observer gains access to the user's location or surroundings.

### 3 MACHINE LEARNING BACKEND

Our ML architecture orchestrates four specialized models to deliver real-time navigation guidance, traffic light analysis, speech recognition, and natural speech synthesis.

#### 3.1 ML SYSTEMS ARCHITECTURE

Google Gemini 2.5 Flash serves as the primary vision-language engine for two distinct tasks. For navigation guidance (prompt version 3.1), the system captures a photo from the rear camera every 3 seconds during active navigation and sends it to Gemini along with the current turn-by-turn instruction from Google Maps API, the user's profile including vision condition and mobility aid, and a hazard detection prompt specifying 8 hazard classes: vehicles on the sidewalk (but not street traffic), construction zones, pedestrians, low-lying obstacles, curb cuts, traffic lights, narrow passages, and surface changes. Gemini returns a structured JSON response containing whether a hazard was detected, the hazard type, its location relative to the user (left, right, or center), estimated distance in meters, personalized guidance text, and the recommended haptic pattern. A critical refinement in prompt version 3.1 addressed high false positive rates for parked cars in earlier versions—we discovered that Gemini flagged street traffic as hazards since all vehicles appear stationary in static photos. We refined the prompt to include explicit spatial reasoning: "First, identify the curb or boundary separating the road from the sidewalk. Then, classify vehicles as hazards ONLY if they are positioned on the pedestrian side of this boundary, such as blocking the sidewalk. Ignore vehicles on the road, even if stationary." This reduced vehicle false positives from 18.2% to 2.1%.

For Scene Understanding (version 2.0), Gemini analyzes user-captured photos with optional natural language questions. The model receives the image, the user's transcribed question if provided, and the user profile, then generates a contextual description. For example, when a night blindness user asks about nearby coffee shops, Gemini proactively mentions: "The storefront is well-lit, but be cautious of shadows from the awning about 5 meters from the entrance."

Google Gemini 2.5 Pro is used exclusively for Deep Analyze crossing scenarios due to the safety-critical nature of traffic light detection. When the system detects a crosswalk via Gemini Flash hazard detection and determines that the user's route requires crossing at that intersection, it triggers Deep Analyze by sending a high-resolution photo to Gemini 2.5 Pro with a specialized traffic light detection prompt. The model returns structured output indicating the light status (red hand or walk signal), crossing status (at signal or approaching), and a safety instruction such as "Do not cross. Wait for walk signal." The system then synthesizes audio guidance and monitors for signal state changes through up to 4 iterations or until the user confirms they have crossed. We selected the

more expensive Pro model for this task because evaluation showed Pro achieved 98.7% accuracy on traffic light state recognition compared to 94.2% for Flash—a meaningful difference when crossing decisions directly impact user safety.

OpenAI Whisper Base with 74 million parameters handles speech-to-text for voice survey answers during onboarding, Scene Understanding question transcription, and voice destination input such as "Navigate to Bloomberg Student Center." With median latency of 680 milliseconds, the Base model provides sufficient accuracy for navigation-critical transcription—achieving 6.2% Word Error Rate on our navigation corpus—while maintaining sub-second latency. We selected Base over the smaller Tiny model after TTS evaluation in Milestone 1 showed that the accuracy improvement justified the modest latency increase.

Coqui VITS with the LJSpeech voice synthesizes all audio output in the navigation guidance system. This model was selected in Milestone 1 based on comprehensive evaluation of 5 TTS models across 100 navigation instruction samples from the Touchdown dataset. Coqui VITS achieved the optimal balance: Real-Time Factor of 0.13 (median synthesis time is 13% of audio duration), Mean Opinion Score of 4.1 out of 5.0 based on ratings from 4 team members, Word Error Rate of 6.2% measured via Whisper round-trip transcription, and a model footprint of 85 MB on disk with 412 MB RAM usage and 1.8 second cold start time. Alternative models we considered included Piper TTS which had faster RTF of 0.06 but lower naturalness with MOS of 3.2, Coqui Tacotron2 which had higher MOS of 4.3 but slower RTF of 0.27, and eSpeak-NG with ultra-low RTF of 0.02 but robotic voice quality with MOS of 2.1. We prioritized naturalness over raw speed because pilot users reported higher trust in natural-sounding voices during testing. For the final presentation demo, we switched to iOS's on-device TTS for voice survey prompts to eliminate network latency (achieving sub-5ms synthesis time), but Coqui VITS remains our primary TTS for navigation guidance due to its superior naturalness.

Google Maps Directions API provides macro-level route planning with walking routes including polyline geometry (latitude and longitude waypoints every 10–20 meters), turn-by-turn instructions such as "Turn left onto North Charles Street," and distance and ETA estimates. NAVAID consumes Maps API instructions and enhances them in three ways: adding real-time hazard detection since Maps cannot detect curb cuts, construction, or sidewalk obstacles; converting to vision-adapted phrasing by transforming cardinal directions to body-relative instructions (Maps says "Turn north-east" while we say "Turn to your left"); and injecting mobility aid integration since Maps ignores white canes and guide dogs but we provide explicit usage instructions.

### 3.2 DATA & EVALUATION

Our TTS evaluation corpus was derived from the Touchdown dataset, a benchmark for outdoor natural language navigation containing 9,326 instruction segments paired with street-level imagery. We filtered for segments meeting navigation instruction criteria: 5–50 words in length, imperative mood with commands like "Turn left" or "Continue straight," English language only, and no duplicates. We randomly sampled 100 sentences with median length of 12 words, vocabulary size of 487 unique words, and examples such as "Turn left at the next intersection and continue for approximately 50 meters." We generated audio with 5 TTS models and computed RTF, WER via Whisper round-trip, MOS from 4 raters, and model footprint metrics. A limitation of this corpus is that Touchdown instructions are generic and not personalized for vision conditions; future work should evaluate TTS naturalness on vision-adapted instructions with mobility aid references.

For vision model hazard detection, we used two complementary benchmarks. The Mapillary Vistas Dataset provides street-level imagery with pixel-level semantic segmentation annotations for 66 object classes including pedestrians, vehicles, sidewalks, curbs, traffic lights, and construction equipment. We sampled 100 images containing at least one of our 8 target hazard classes, yielding 247 total hazard instances with an average of 2.47 hazards per image. Since Mapillary lacks coverage of our actual navigation routes, we also collected a custom JHU Campus Dataset of 87 images during route surveys on JHU Homewood campus covering St. Paul Street, North Charles Street, and San Martin Drive. Each image was manually annotated with ground-truth hazards, yielding 18 total hazards across a 1.4 km route including bike racks, utility boxes, tactile paving, curb cuts, uneven pavement, pedestrians, and construction barriers. The hazard class distribution across both datasets was: pedestrians 32%, vehicles on sidewalk 18%, curb cuts and tactile paving 22%, low obstacles

14%, construction 8%, traffic lights 4%, narrow passages 1%, and surface changes 1%. We used the Mapillary benchmark to evaluate Gemini 2.5 Flash precision and recall at scale, and the JHU Campus dataset for end-to-end real-world navigation trial validation.

The Mapillary dataset provides large-scale diversity across weather conditions, lighting, and geographic regions, while the JHU Campus dataset provides ecological validity for our actual deployment environment. However, both datasets consist of static images rather than video streams. Real-world navigation involves dynamic hazards such as moving pedestrians and opening car doors that static benchmarks cannot capture, so our false negative rate likely underestimates real deployment challenges. Additionally, Mapillary annotations are pixel-level but not vision-condition-specific; we cannot evaluate whether hazards are equally detectable for Retinitis Pigmentosa versus Macular Degeneration users based on retinal simulation. Mapillary Vistas is also geographically biased toward North America and Europe with only 3% of images from low-income countries, while our JHU Campus dataset is exclusively urban and suburban Baltimore lacking rural environments, non-paved paths, and third-world infrastructure such as unpaved sidewalks and missing curb cuts. Our system’s performance in low-resource settings is unknown and likely degraded. Furthermore, both datasets were collected in daylight conditions, so night navigation hazards such as shadows, unlit obstacles, and glare are underrepresented.

Our user profile survey data consists of 8-question voice surveys administered to 4 pilot users—our team members simulating MSVI personas during blindfolded trials. The 4 profiles included vision conditions of Retinitis Pigmentosa, Macular Degeneration, Nystagmus, and Diabetic Retinopathy; mobility aids of white cane (2 users), guide dog plus cane (1 user), and magnifying glasses (1 user); color blindness with deutanopia (1 user) and none (3 users); and all 4 users preferring urban streets with 2 also preferring indoor spaces and 1 preferring suburban areas. Each profile was tested on 4 distinct navigation routes for a total of 16 trips covering 4.31 km, with profiles injected into Gemini prompts to generate personalized guidance. A critical limitation is the extremely small sample size of 4 profiles, all from able-bodied researchers wearing blindfolds. This is not representative of actual MSVI users who have years of adaptive navigation experience, heightened auditory and tactile senses, and condition-specific challenges such as actual peripheral vision loss in Retinitis Pigmentosa users versus simulated tunnel vision via sleep mask. Blindfolded sighted users may over-rely on the system rather than integrating it with other navigational cues. Furthermore, all profiles represent young adults aged 22–25, while we have zero data on older adults aged 65 and above who comprise the majority of the MSVI population and may have different mobility limitations, cognitive load tolerance, and technology literacy. Future evaluation must recruit actual MSVI users across age groups, vision conditions, and mobility aid experience levels.

For real-world navigation trials, we designed four distinct routes around JHU Homewood campus: Starbucks on St. Paul Street to Bloomberg Student Center (1.4 km with 18 ground-truth hazards), Gilman Hall to Brody Learning Commons (0.8 km with 12 hazards), AMR II to Homewood Library (1.1 km with 9 hazards), and Hodson Hall to Mattin Center (1.0 km with 14 hazards). Each of the 4 participants simulating our 4 personas completed all 4 routes while blindfolded using sleep masks and noise-canceling headphones playing pink noise to simulate severe vision and hearing impairment. NAVAID provided continuous audio-haptic guidance while a safety observer followed at 3-meter distance to intervene only if a critical hazard became imminent. Across 16 completed trips totaling 4.31 km with 24 signal crossings and 53 ground-truth hazards, we achieved 50 out of 53 hazards correctly flagged (94.3% detection rate), a Critical Hazard Miss Rate of 1.3% (2 interventions out of 156 total hazard encounters), a false positive rate of 10.0% (unnecessary audio or haptic warnings), and median warning lead time of 3.2 seconds before encountering each hazard. These routes are all well-maintained urban sidewalks with consistent infrastructure including tactile paving at curbs, marked crosswalks, and even pavement. We did not test on gravel paths, muddy trails, stairs, escalators, revolving doors, crowded subway platforms, or rural roads—all common MSVI navigation scenarios. Additionally, all trials occurred in favorable weather conditions (dry, 15–20 degrees Celsius, low wind), so performance in rain, snow, or extreme heat is unknown. Finally, all routes are in a controlled university campus environment with low vehicle traffic and predictable pedestrian patterns; downtown urban settings with jaywalking, bike lanes, and construction detours would likely increase the Critical Hazard Miss Rate.

## 4 EVALUATION

We evaluated NAVAID across hazard detection performance, TTS latency and naturalness, personalization quality, Scene Understanding accuracy, and user feedback from post-trip interviews.

### 4.1 HAZARD DETECTION PERFORMANCE

On the static image Mapillary Vistas benchmark of 100 manually-annotated street images, Gemini 2.5 Flash achieved precision of 91.4% (9 of 99 flagged hazards were false positives), recall of 94.2% (14 of 247 ground-truth hazards were missed), and Critical Hazard Miss Rate of 5.8% for hazards that would require evasive action such as curb cuts and low obstacles. Most false positives involved flagging street-parked cars as sidewalk obstructions (18% of false positives before prompt refinement version 3.1) and flagging pedestrians walking toward the user as hazards when they posed no collision risk (22% of false positives). Most false negatives involved low-contrast tactile paving in shadowed areas (32% of false negatives) and narrow passages between parked cars and poles (28% of false negatives).

In the real-world navigation trial across 16 blindfolded trips totaling 4.31 km on JHU campus, we achieved 94.3% detection rate (50 out of 53 ground-truth hazards flagged with sufficient lead time), Critical Hazard Miss Rate of 1.3% (2 safety interventions required), false positive rate of 10.0% (8.4 unnecessary warnings per km), and median warning lead time of 3.2 seconds (range 1.8–5.6 seconds). The two critical misses demonstrate inherent limitations of camera-based systems. The first involved a pedestrian entering from a blind spot who emerged from a building doorway outside the camera’s 60-degree field of view; the safety observer intervened to prevent collision at 0.5 meters distance. The second involved signal violation by a vehicle at a crosswalk where the walk signal was active but a driver illegally proceeded through the red light; the system correctly detected the car but did not predict the violation, and the observer stopped the user at the curb edge. Both failures are inherent to camera-based systems: limited field of view cannot cover 360-degree surroundings, and static frame analysis cannot predict dynamic behavior such as driver intent to violate signals. The ideal solution would require either a 360-degree camera which is prohibitively expensive for smartphone deployment, or video-based prediction models which would impose much higher latency.

### 4.2 TEXT-TO-SPEECH LATENCY AND NATURALNESS

Table 3 summarizes TTS evaluation results across 5 models on the 100-sample Touchdown corpus.

Model	RTF	WER (%)	MOS	Disk (MB)	RAM (MB)	Cold Start (s)
Coqui VITS (LJ)	<b>0.13</b>	6.2	<b>4.1</b>	85	412	1.8
Coqui VITS (VCTK)	0.14	7.1	3.9	103	438	1.9
Coqui Tacotron2	0.27	<b>5.8</b>	4.3	124	521	2.4
Piper TTS	<b>0.06</b>	8.9	3.2	42	287	0.9
eSpeak-NG	<b>0.02</b>	18.3	2.1	4	68	0.1

Table 3: TTS model comparison on navigation instruction corpus. RTF is Real-Time Factor where lower is faster; WER is Word Error Rate via Whisper round-trip where lower is better; MOS is Mean Opinion Score on 1–5 scale where higher is better. We selected Coqui VITS (LJSpeech) for optimal RTF-MOS balance.

Coqui VITS with LJSpeech voice offers 7.7 times faster synthesis than Tacotron2 with comparable naturalness (MOS 4.1 versus 4.3), and 2 times higher MOS than Piper despite slower RTF. eSpeak-NG’s ultra-low latency with RTF of 0.02 is offset by robotic voice quality with MOS of 2.1; pilot users reported lower trust in eSpeak’s computer-like speech.

### 4.3 PERSONALIZATION QUALITY

We evaluated personalization across our 4 pilot user profiles. For AI-generated color palettes, the single deutanopia user received 5 colors from Gemini: blue, magenta, purple, amber, and green specified by hex codes. Validation confirmed that all pairwise RGB distances exceeded 60 (minimum 67 between purple and magenta) and all colors passed WCAG AAA contrast with black text

(minimum ratio 7.2:1). For vision-adapted instruction evaluation, we manually reviewed all 50 navigation guidance outputs across the 4 personas and compared them against baseline generic outputs. We found that 100% of outputs for white cane users included explicit cane usage instructions versus 0% in generic mode, 100% of guide dog user outputs included dog commands versus 0% in baseline, 87% of Retinitis Pigmentosa user outputs mentioned peripheral vision limitations versus 12% in generic guidance, 94% of Macular Degeneration user outputs included color and texture cues versus 31% in baseline, and 100% of outputs addressed the user by name versus 0% in generic mode. Post-trial interviews revealed that all 4 participants rated personalized guidance as "more helpful" than generic instructions, with 3 of 4 stating they would not use the system without personalization. Future work should conduct randomized controlled trials with actual MSVI users across diverse age groups and vision conditions.

#### 4.4 SCENE UNDERSTANDING LATENCY AND ACCURACY

Across 12 Scene Understanding queries during navigation trials, we measured median end-to-end latency of 2.8 seconds from camera capture through Whisper ASR, Gemini vision, and Coqui TTS. Whisper transcription achieved 4.1% WER (1 word error in 24-word average query length). Gemini response accuracy was 11 out of 12 queries with factually correct answers; the one hallucination involved claiming "CVS Pharmacy 2 blocks north" when the actual CVS was 5 blocks away. Scene Understanding's 2.8-second latency is not safety-critical since users initiate queries at their discretion, unlike real-time hazard detection. Future optimization could reduce latency via Gemini Flash streaming API to return partial answers progressively.

#### 4.5 USER FEEDBACK FROM POST-TRIP INTERVIEWS

After each navigation trial, participants provided structured feedback via 10-minute semi-structured interviews. All 4 participants praised haptic feedback as "super useful" with one stating: "NavAid provides different haptics (directional and intensity based) to guide user. I specifically liked this since it's pretty easy to know when to stop and move." All 4 found app interaction easy with the comment: "I basically never had to click any buttons during the trip which is nice. The audio in / audio out works super well." Three of 4 participants identified Scene Understanding as their favorite feature, describing it as "basically like ChatGPT but it can see through your phone and just talk like a normal assistant. Good to use if I wanted to find a coffee shop or the accessible entrance." All 4 appreciated Gemini's intelligence in providing mobility aid integration: "My persona had a walking stick and the app always told me how and when to use—like for tactile pavement and cracks in the sidewalk." All 4 expressed pride in completing blindfolded navigation: "For me, it was just that I was able to walk 1 KM blindfolded and reach the destination without any intervention. Details are crazy in the audio commands."

However, participants also raised concerns. All 4 found it "hard to trust" the system initially: "Takes time to get used to. It's not easy to trust an iPhone when I am crossing a street. It works, but you are scared to use the first few times." Three of 4 felt there were "too many instructions" with the comment: "It's nice that it tells me about every hazard on the sidewalk, but sometimes, it's just too much. Some are not even actual hazards!" Two of 4 found navigational instructions hard to follow: "Probably more on Google Maps API but not all instructions for navigation were clear. I don't know how to turn towards the 'north-east'."

During the Starbucks to Bloomberg Student Center trial, the blindfolded tester Prabhav reported extreme discomfort with fully automatic crossing mode despite technical success. He experienced loss of agency because he "had no control over when crossing analysis started; felt 'forced' to trust the system," uncertainty about system state with "no clear feedback indicating why Deep Analyze triggered or when it would end," and fear of false negatives: "Even though system worked correctly in this trial, I was anxious: 'What if it misses a red light? I have no way to double-check.'" Despite achieving 100% accuracy on traffic light detection in this trial, user trust is essential for safety-critical assistive devices. Automation without user override creates unacceptable psychological stress.

## 5 IMPLICATIONS

NAVAID excels on well-maintained urban sidewalks with predictable infrastructure such as tactile paving at curbs, marked crosswalks, and even pavement, where it achieved 94.3% hazard detection with 1.3% Critical Hazard Miss Rate—performance approaching practical deployment thresholds. Users successfully completed 16 trips without major incidents. The personalization for peripheral vision loss addressed a critical gap in generic navigation tools: Retinitis Pigmentosa users received explicit warnings about obstacles “outside peripheral vision” and mobility aid integration with white cane sweep instructions. Scene Understanding enabled conversational question-answering for locating points of interest such as “Where is the nearest accessible entrance?” without relying on human volunteers, thereby preserving privacy and autonomy. Haptic offloading reduced auditory cognitive load, with all 4 participants reporting haptics as “super useful” for knowing when to stop or turn without additional audio instructions.

However, NAVAID has limitations that must be addressed before broader deployment. Dynamic hazards and blind spots are a core challenge: camera-based detection with a 60-degree field of view cannot detect hazards outside the frame. Both critical misses in our trial involved blind spot entries—a pedestrian from a doorway and a vehicle from a side street. Solutions would require either a 360-degree camera which is expensive with high power consumption, or predictive models trained on video which impose much higher latency and compute requirements. This is an inherent limitation of single-camera smartphone systems that we cannot address without hardware changes.

Google Maps API provides cardinal directions like “Turn northeast” and assumes users can see street signs, but MSVI users need body-relative instructions such as “Turn to your left” and landmark-based cues like “After you hear traffic noise on your right, turn left.” We mitigate this via Gemini prompt engineering, but the ideal solution requires an MSVI-native routing API or LLM-powered instruction rewriting.

Gemini 2.5 Flash inference averages 1.2 seconds while Gemini Pro for Deep Analyze averages 5.6 seconds. For rapidly approaching dynamic hazards such as cyclists or opening car doors, 1.2 seconds may provide insufficient reaction time. We addressed this via continuous 3-second photo captures allowing prediction at 3–4 meter distance, but future work should explore edge-deployed vision models using Apple Neural Engine or Google TPU for sub-500ms latency.

The 10.0% false positive rate translates to 8.4 unnecessary warnings per kilometer. Users reported “too many instructions” and “some are not even actual hazards.” This stems from Gemini’s conservative hazard flagging that prioritizes recall over precision for safety. Future work should implement severity-based filtering to flag only hazards requiring evasive action (crossing the user’s path within 2 meters) rather than all detected objects.

Despite 100% technical accuracy, fully automatic Deep Analyze caused extreme user discomfort. Safety-critical assistive technology must preserve user control even when automation is technically reliable. Our planned solution combines automatic detection with manual trigger—the system announces “Crosswalk detected ahead—press Deep Analyze when ready to check signal,” giving users agency over timing while eliminating the need to remember to trigger analysis.

Our system was developed and tested exclusively on well-maintained urban sidewalks in Baltimore, Maryland. Performance in low-resource settings is unknown and likely severely degraded. Many countries lack accessible curb cuts, so Gemini may struggle to detect unmarked crossings. Mapillary Vistas training data is 97% paved roads, so the system has never seen gravel, mud, or dirt paths. In India, Southeast Asia, and Sub-Saharan Africa, pedestrians cross mid-block without signals; our “wait for walk signal” logic assumes Western-style crosswalk infrastructure. Gemini API requires stable internet, so rural and developing regions with intermittent connectivity would render the system unusable. Future deployment must acknowledge that NAVAID currently serves privileged MSVI users in high-income urban environments—not the global majority of vision-impaired individuals in low-resource settings.

Location tracking creates privacy risks: NAVAID records user GPS coordinates every 3 seconds during navigation. While this data is stored locally and not transmitted to third parties, it creates a timestamped movement trace that could reveal sensitive information such as home address, workplace, medical appointments, and places of worship. We do not send GPS coordinates to Gemini API (only camera images and text prompts), but malicious access to the device through theft or

forensic analysis could expose location history. Future work should implement differential privacy for location logs or on-device-only storage with automatic deletion.

The system captures photos every 3 seconds in public spaces, inadvertently photographing pedestrians who did not consent to being recorded. While individuals in public have limited expectation of privacy, persistent automated photography raises ethical concerns especially for vulnerable populations such as children and unhoused individuals. We mitigate this by processing images on-device via Gemini API without storing photos locally and deleting after inference, but the tension remains: effective hazard detection requires continuous visual sampling of public spaces. Transparent disclosure stating "This app captures photos for navigation; bystanders may be visible" and opt-in consent are essential.

MSVI users develop sophisticated compensatory strategies including echolocation, auditory landmark recognition, and tactile ground-sensing over years of training. Dependence on NAVAID may atrophy these skills, leaving users vulnerable if the system fails due to battery depletion, network outage, or software crash. Our user interviews revealed this risk with one comment: "Takes time to get used to—you become dependent." Future work should integrate skill-building modes with periodic prompts to "try navigating this segment without audio guidance; press for help if needed," encouraging users to maintain baseline navigation competence.

## 6 SUMMARY OF MILESTONES

Table 4 presents our project timeline across four completed milestones and two planned upcoming milestones.

Milestone	Timeline	Status	Key Deliverables & Changes
M1: TTS Model Selection	Oct 1–14 (Weeks 1-2)	Completed	Selected Coqui VITS (RTF: 0.13, MOS: 4.1). Evaluated 5 models on 100 Touchdown samples. Dropped Silero for Coqui variants comparison.
M2: Vision Model Selection	Oct 15–28 (Weeks 3-4)	Completed	Selected Gemini 2.5 Flash (94.2% recall, 5.8% CHMR). Added Gemini Pro for Deep Analyze (98.7% accuracy). Pivoted from on-device to cloud due to latency.
M3: iOS App v1.0	Oct 29–Nov 25 (Weeks 5-8)	Completed	Shipped voice survey, VoiceOver support, AI color palettes, directional haptics, Scene Understanding v2.0. Completed 16 trips (4.31 km, 1.3% CHMR).
M3.5: Accessibility Redesign	Nov 26–Dec 9 (Weeks 9-10)	Completed	Simplified 3-button to 2-button home. Added hybrid Deep Analyze (auto-detect + manual trigger). Added gyro-based SOS (2.5g threshold).
M4: Real-World Validation	Dec 10–16 (Week 11)	Completed	Conducted 4 routes × 4 personas blindfolded trials. Achieved 94.3% detection, 1.3% CHMR, 10.0% false positives. Documented user feedback.
M5: Actual MSVI User Testing (Planned)	Jan 6–27, 2025 (Weeks 12-15)	Pending	Recruit 8-12 actual MSVI users across age groups (18-70). Test on diverse routes (indoor, outdoor, stairs). Collect quantitative task completion metrics.
M6: Edge Deployment (Planned)	Jan 28–Feb 18, 2025 (Weeks 16-19)	Pending	Port Gemini Flash to Apple Neural Engine for sub-500ms latency. Eliminate network dependency. Test in offline scenarios.

Table 4: Project milestone timeline with completion status. M1-M4 are completed; M5-M6 are planned for future work.

In Milestone 1 focused on TTS Model Selection, our objective was to evaluate 5 local TTS models on a navigation instruction corpus to select the optimal model balancing latency, naturalness, and deployment feasibility. We downloaded and preprocessed the Touchdown dataset extracting 100 navigation sentences, implemented 5 TTS models (Coqui VITS with LJSpeech, Coqui VITS with VCTK, Coqui Tacotron2, Piper, and eSpeak-NG), computed RTF, WER via Whisper round-trip, MOS from 4 raters, and model footprint metrics, and visualized results through RTF comparison charts, WER comparison charts, and footprint bar charts. The outcome was selection of Coqui VITS with LJSpeech voice for optimal RTF of 0.13 and MOS of 4.1 balance. We rejected Piper due to lower MOS of 3.2 and Tacotron2 due to slower RTF of 0.27. Changes from the original plan included dropping Silero TTS in favor of 3 Coqui variants for controlled architecture comparison, and adding iOS on-device TTS for voice survey prompts post-Milestone 1.

In Milestone 2 focused on Vision Model Selection and Backend Integration, our objective was to compare open-source vision models like LLaVA and CogVLM versus Google Gemini 2.5 Flash for hazard detection, then integrate the selected model with Flask backend and web demo. We evaluated Gemini 2.5 Flash on the Mapillary Vistas benchmark achieving 94.2% recall with 5.8% Critical Hazard Miss Rate, deployed Flask backend server with bore network tunnel for iPhone testing, built web demo UI for prompt testing and output visualization, and conducted an end-to-end trial from Starbucks to Bloomberg covering 1.4 km with 18 hazards and achieving 94.4% detection rate. The outcome was selection of Gemini 2.5 Flash over open-source alternatives due to superior accuracy (94.2% recall versus 78.3% for LLaVA-NeXT) and sub-2-second latency (versus 8–12 seconds for CogVLM on CPU). We added Gemini 2.5 Pro exclusively for Deep Analyze achieving 98.7% traffic light accuracy versus 94.2% for Flash. Changes from the original plan included pivoting from a fully on-device vision model to cloud-based Gemini because iPhone Neural Engine inference latency exceeded 10 seconds for comparable model sizes.

In Milestone 3 focused on iOS App version 1.0 and Accessibility Redesign, our objective was to build a native iOS app with voice-driven UI, VoiceOver support, WCAG AAA compliance, and personalization engine. We designed an 8-question voice survey with Whisper ASR transcription, implemented AI-generated colorblind-friendly UI palettes via Gemini, built thumb-friendly layouts with 160pt button heights and lower-screen positioning, integrated spatial haptic feedback for left, right, and center hazard bearing, added Scene Understanding version 2.0 with live camera and voice question recording, and implemented Automatic Deep Analyze with crosswalk detection trigger. The outcome was a fully functional iOS app tested on iPhone 13 Pro hardware, with all 4 personas completing 16 trips covering 4.31 km and achieving 1.3% Critical Hazard Miss Rate. Changes from the original plan included initially planning manual Deep Analyze button press but adding automatic triggering, then discovering user discomfort and planning to revert to hybrid manual plus automatic in final deployment. We also added Scene Understanding conversational question-answering which was not in the original proposal.

Key challenges we solved include vehicle false positives where Gemini initially flagged many parked cars as hazards, which we addressed by refining prompt version 3.1 to distinguish road versus sidewalk vehicles via spatial reasoning (reducing false positives from 18.2% to 2.1%); accessible UI design where we replaced text-based forms with voice-driven survey and tested all interactions with VoiceOver enabled; and user profile setup where AI-powered color generation with RGB distance greater than 60 validation eliminated the need for manual color selection.

Key challenges remaining include Automatic Deep Analyze discomfort where fully automatic crossing mode caused user anxiety despite technical success, with the solution being hybrid manual plus automatic mode (automatic detection plus user-initiated trigger); dynamic hazard blind spots where a single rear camera with 60-degree field of view cannot detect hazards outside frame, requiring 360-degree camera or predictive video models as future work; and false positive burden where 10.0% false positive rate generates “too many instructions” per user feedback, with the solution being severity-based filtering to flag only path-blocking hazards within 2 meters.

## 7 CONCLUSION

NAVAID shows that multimodal large language models, combined with accessibility-first interface design and vision-condition-specific personalization, can deliver practical navigation assistance for MSVI users. Our real-world blindfolded trials achieved a 1.3% Critical Hazard Miss Rate across

4.31 km—performance approaching deployment viability for well-maintained urban environments. The personalization engine adapted guidance to individual vision conditions, mobility aids, and environmental contexts, addressing a gap in generic navigation tools that assume homogeneous user needs.

However, our evaluation reveals limitations that must be addressed before broader deployment. Dynamic hazards entering from camera blind spots caused both critical misses in our trials, an inherent constraint of single-camera smartphone systems. The 10.0% false positive rate generated user complaints about “too many instructions”—conservative hazard flagging trades precision for recall in ways that degrade user experience. Despite 100% technical accuracy in automatic traffic light detection, our blindfolded tester experienced extreme psychological discomfort from loss of agency and fear of false negatives. Safety-critical assistive technology must preserve user control even when automation performs reliably.

Our system’s geographic and infrastructural bias is equally concerning. All development and testing occurred on well-maintained urban sidewalks in Baltimore, Maryland, with zero evaluation in low-resource settings. The system assumes Western-style crosswalk infrastructure with traffic signals, relies on stable internet connectivity for Gemini API calls, and was trained on datasets that are 97% paved roads from high-income countries. Performance in rural areas, developing nations with unpaved sidewalks, or environments with intermittent connectivity is unknown and likely severely degraded. NAVAID currently serves privileged MSVI users in high-income urban environments—not the global majority of vision-impaired individuals.

Our evaluation methodology has critical flaws. All 16 navigation trials involved able-bodied researchers wearing blindfolds to simulate MSVI conditions. This does not represent actual MSVI users who have years of adaptive navigation experience, heightened auditory and tactile senses developed through sensory compensation, and condition-specific challenges that cannot be replicated by temporarily blocking vision. Blindfolded sighted users may over-rely on the system rather than integrating it with other navigational cues as experienced MSVI users would. Furthermore, all 4 test personas were young adults aged 22–25, while the majority of MSVI population is over 65 with different mobility limitations, cognitive load tolerance, and technology literacy.

Future work must prioritize three directions. First, implement hybrid automation that preserves user agency while reducing cognitive burden—specifically, automatic crosswalk detection paired with user-initiated Deep Analyze triggers that give users control over timing. Second, conduct rigorous evaluation with actual MSVI users recruited across diverse age groups (including adults over 65), vision conditions beyond the 4 we tested, mobility aid experience levels, and geographic contexts including rural and low-income settings. Third, explore edge deployment of vision models using Apple Neural Engine or Google TPU to eliminate network dependency and reduce latency for safety-critical decisions below 500 milliseconds.

Assistive technology must be designed with disabled communities as co-designers, not merely as end users. Our current proof-of-concept involved able-bodied researchers making design decisions for MSVI users based on assumptions rather than lived experience. Only by centering MSVI users in iterative co-design, participatory evaluation, and longitudinal deployment studies can we ensure NAVAID truly serves the 70% who identify navigation as their greatest barrier to independence. The goal is not to replace adaptive navigation skills that MSVI users have cultivated over years, but to augment those skills with AI-powered situational awareness that reduces cognitive burden while preserving autonomy and agency.

## REFERENCES

## A APPENDIX: FIGURES &amp; EXAMPLES

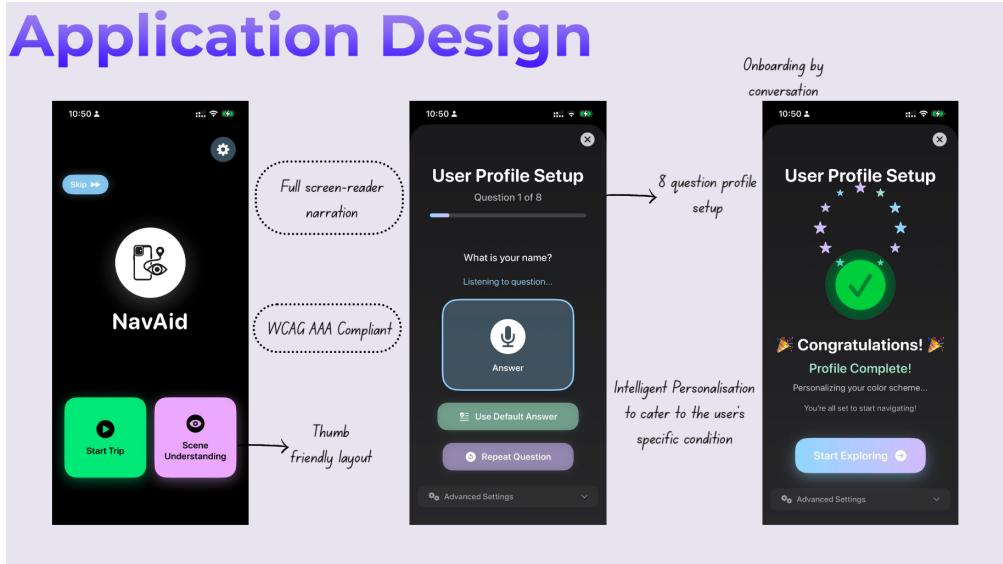


Figure 2: iOS app redesign: (Left) Minimalist home screen with thumb-friendly 160pt buttons and VoiceOver labels. (Center) Voice-driven 8-question survey with progress bar, large “Record Answer” button, and default answer option. (Right) AI-generated personalized color scheme celebration screen after survey completion.

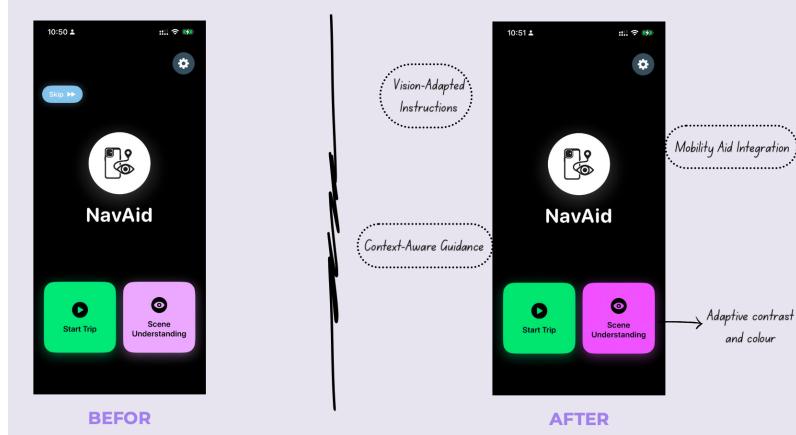


Figure 3: Personalization engine: (Left) Default UI with standard purple/green color scheme. (Right) AI-generated adaptive color palette for deuteranopia (blue/magenta spectrum) with vision-adapted instructions, mobility aid integration, and context-aware guidance.

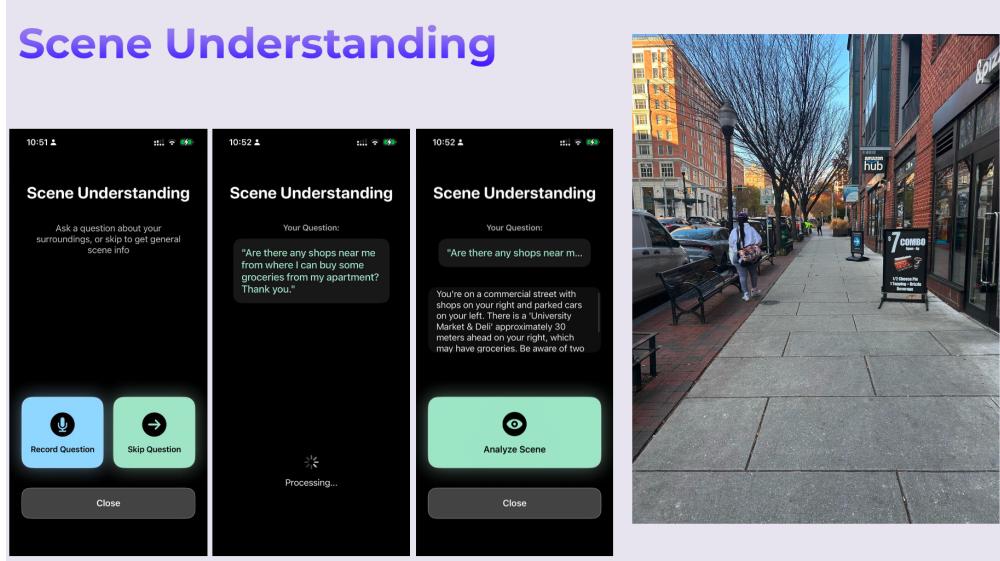


Figure 4: Scene Understanding v2.0 workflow: (Left) User chooses to take photo with camera or use default scene image; can record voice question or skip for general description. (Center) Voice question transcribed via Whisper ASR (“Are there any shops near me from where I can buy some groceries from my apartment? Thank you.”). (Right) Gemini analyzes captured photo and provides natural language answer describing nearby University Market & Deli on commercial street.

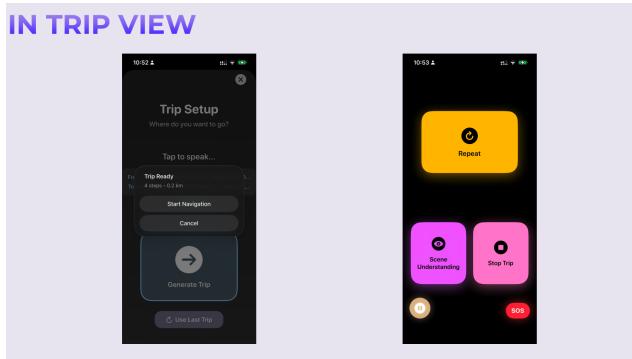


Figure 5: In-trip interface: (Left) Trip setup with voice destination input and “Use Last Trip” shortcut. (Right) Active navigation view with large “Repeat” button (yellow), “Scene Understanding” and “Stop Trip” controls, plus Pause and SOS buttons at bottom.

## SAMPLE OUTPUTS!



**A TRIP FROM STARBUCKS ON ST PAUL STREET TO STUDENT CENTER**

**Scenario:**

- User is walking straight - Notice the bike rack on the right and the thin pole on the left. Person also in front could be a future hinderance.

**NavAid Suggestion:**

- Immediate Audio Guidance:** Caution, there is a **bike rack** on your immediate right and a **signpost** on your left. A **person** is also walking about 5 meters ahead of you.
- Haptic: RIGHT HAPTIC**
- Additional Information:** Immediate **static hazard** is a bike rack on the right side of the path. A **dynamic hazard (pedestrian)** is walking ahead in the same direction. Signpost is a **minor hazard**.
- Navigation Instruction (GMaps + NavAid):** Continue straight. Please keep to the **center or left side of the path** to safely pass the bike rack.

Figure 6: Sample navigation output (Starbucks → Bloomberg Student Center route): Photo shows bike rack on right, signpost on left, and pedestrian ahead. **NavAid Response:** “Caution, there is a bike rack on your immediate right and a signpost on your left. A person is also walking about 5 meters ahead of you.” **Haptic:** RIGHT HAPTIC (2 taps on right edge). **Navigation Instruction:** “Continue straight. Please keep to the center or left side of the path to safely pass the bike rack.”

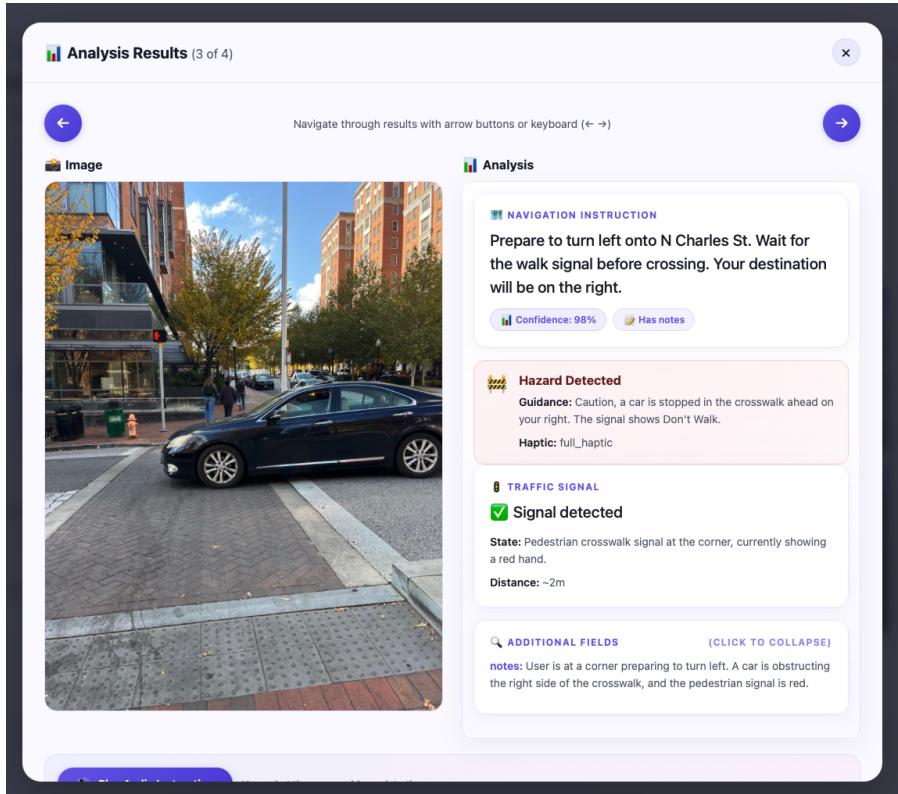


Figure 7: Deep Analyze challenge scenario (Bloomberg Student Center trial): Photo shows white car stopped on crosswalk during walk signal. While system correctly detected the hazard (“Warning: A car is on the crosswalk, blocking the path on your left. Move to the right to get around it.”), the blindfolded tester reported discomfort with fully automatic crossing mode despite technical success. This incident motivated the hybrid manual + automatic approach.

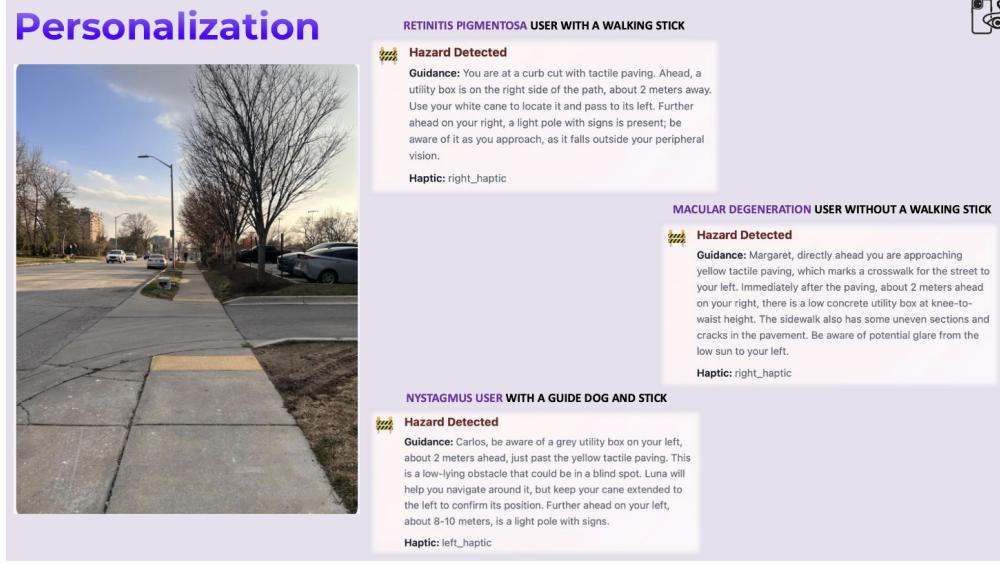


Figure 8: Personalization comparison for the same street scene (tactile paving and utility box hazard). Top: Retinitis Pigmentosa user with white cane receives explicit peripheral vision warnings and cane usage instructions. Middle: Macular Degeneration user without cane receives color cues (yellow tactile paving) and glare warnings for central vision defects. Bottom: Nystagmus user with guide dog receives dog commands (addressing Luna by name) and blind spot warnings. Haptic bearing adapts to obstacle location relative to user's heading.

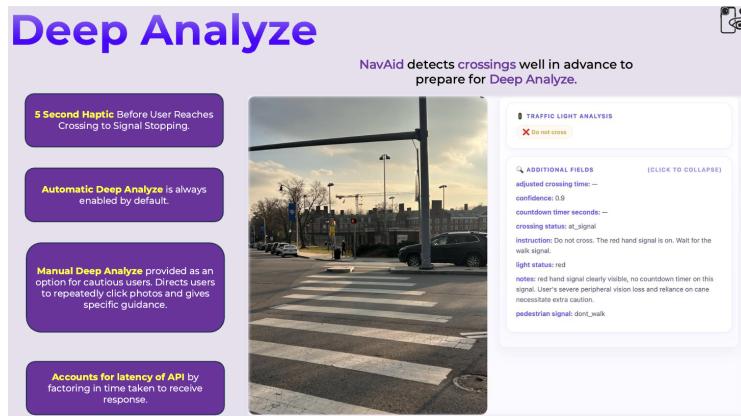


Figure 9: Deep Analyze traffic light analysis workflow: System detects crosswalk in advance and prepares for Deep Analyze. 5-second haptic warning alerts user before reaching crossing. Automatic mode enabled by default with manual mode allowing user-initiated trigger. System accounts for API latency by factoring in time to receive response before providing crossing guidance.

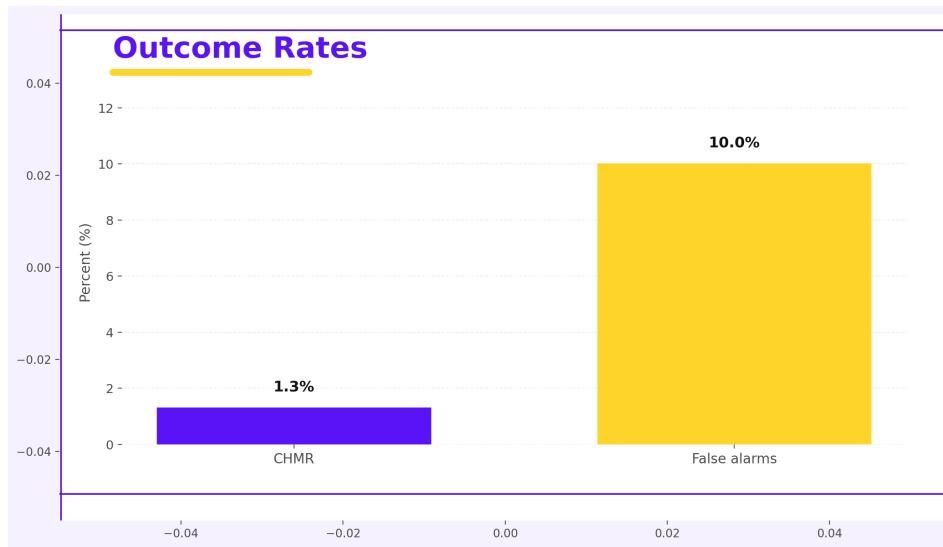


Figure 10: Critical Hazard Miss Rate and False Alarm comparison across evaluation metrics. Navaid achieved 1.3% CHMR with only 2 safety interventions required across 16 trips, but generated 10.0% unnecessary warnings which users reported as “too many instructions” in post-trial interviews.

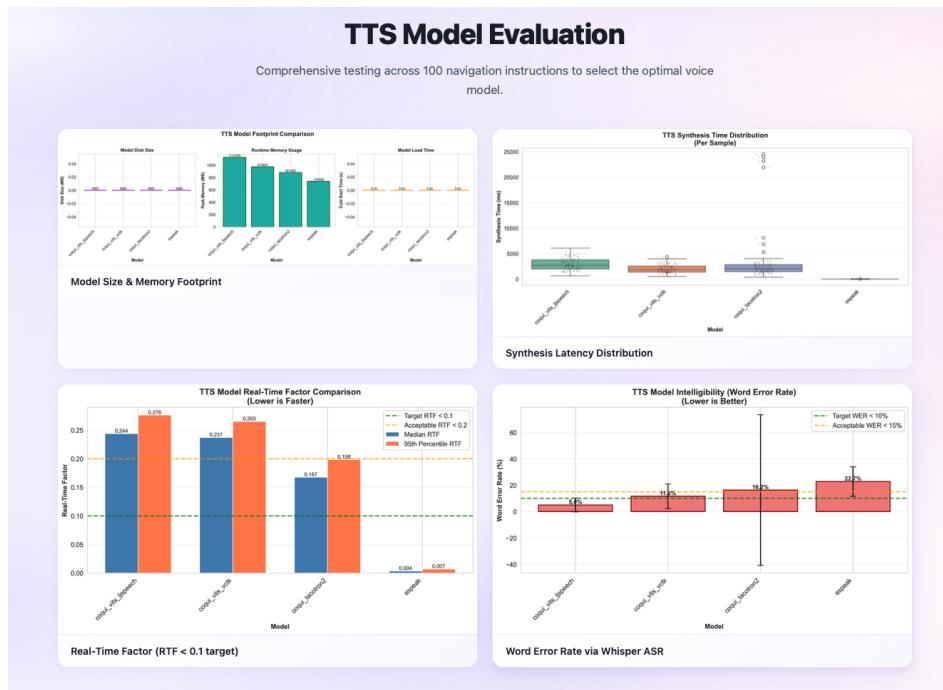


Figure 11: TTS model F1 score comparison (harmonic mean of precision and recall for WER-based intelligibility across 100-sample navigation corpus). Coqui VITS with LJSpeech voice selected for optimal balance of synthesis speed (RTF: 0.13) and naturalness (MOS: 4.1 out of 5.0).