ARTIFICIAL INTELLIGENCE, INTERNET OF THINGS

AI-Powered Smart Glasses Using the ESP32 Processor

Shibu Chacko, J. Harinand, Arya B. Shetty, P. Karthik, S N. Pavan

Department of Computer Science (IoT & Cybersecurity including Blockchain), Alva's Institute of Engineering and Technology

Published November 2026

Abstract

Vision impairment affects over 2.2 billion people worldwide and continues to impose severe barriers to independent daily life, from safe outdoor navigation to reading signage and recognising familiar faces. Conventional aids such as the white cane provide limited scene awareness and cannot describe objects, read text, or interpret traffic signals. The convergence of affordable edge-computing hardware, mature deep learning frameworks, and free-tier cloud AI services has spawned a new generation of AI-powered smart glasses that begin to close this gap. This paper presents a structured comparative review of five peer-reviewed conference publications spanning 2020 to 2025, each proposing a working smart-glasses prototype for blind and visually impaired users. The systems surveyed range from a minimal ESP32-based image-relay prototype to a classical Viola-Jones OpenCV pipeline, a YOLOv3 client-server deep learning design, a YOLOv5 and Kalman-filter sensor-fusion wearable, and a Google Coral Edge TPU driven autonomous navigation system. Each design is analysed across hardware platform, detection algorithm, sensing modalities, user feedback channel, and quantitative performance where available. A cross-paper comparison is supported by a generalised four-layer architectural model, a taxonomy of detection algorithms, and a hardware evolution timeline. The survey concludes with a discussion of open challenges and concrete future directions including large language model integration and hybrid cloud-edge processing.

Keywords: smart glasses; visually impaired; assistive technology; ESP32-CAM; object detection; YOLO; edge AI; Coral TPU; sensor fusion; wearable computing; deep learning; computer vision

1 Introduction

Vision loss is one of the most consequential disabilities affecting human quality of life today. The World Health Organization estimates that at least 2.2 billion people globally live with near or distance vision impairment, and a significant proportion of these cases remain either unaddressed or under-served by affordable assistive technology [6]. In India alone, surveys estimate that tens of millions of individuals are blind or severely visually impaired, yet access to advanced vision-assistance systems is constrained primarily by cost [7].

Traditional assistive tools – the white cane and the guide dog foremost among them – have served this population for decades, but they offer only blunt obstacle awareness. They cannot identify a face across a room, read a restaurant menu, describe the contents of a crowded street scene, or warn a pedestrian that a traffic light has turned red. These limitations have long motivated researchers in assistive technology and computer vision to explore wearable systems capable of providing real-time visual intelligence.

Three technological developments have recently made meaningful progress possible. First, single-board computers and microcontrollers such as the Raspberry Pi, the ESP32-CAM, and the Google

Coral Edge TPU have brought sufficient processing power and connectivity into a glasses-frame form factor. Second, single-pass convolutional object detectors – particularly the YOLO family – have made robust, near-real-time detection feasible on devices with limited computational budgets. Third, free-tier cloud AI APIs and open-source speech and OCR models have opened access to large language model capabilities at essentially zero recurring cost.

Against this backdrop, a growing body of conference publications has emerged proposing AI-powered smart glasses prototypes. This paper systematically surveys five of them, selected to span the breadth of architectural choices made between 2020 and 2025:

Das et al. (2021) – an ESP32-based low-cost image-relay prototype [1]; Lin et al. (2020) – an EPSON BT-300 client-server deep learning system using YOLOv3 [2]; Uma et al. (2023) – the "Third Eye" Raspberry Pi and YOLOv5 sensor-fusion wearable [3]; Babu et al. (2025) – an ESP32-CAM and OpenCV Haarcascade system with GPS [4]; Sharma et al. (2025) – a Coral Edge TPU and Raspberry Pi Zero W autonomous navigation system [5].

The remainder of this paper reviews each system individually (Section 2), presents a comparative analysis and a generalised four-layer architecture (Sections 3 and 4), discusses core enabling technologies and reported performance (Sections 5 and 6), and identifies open challenges and future directions (Sections 7 and 8) before concluding in Section 9.

2 Literature review

2.1 Low-cost smart glass using ESP32 (Das et al., 2021)

Das, Saxena, and Rout [1] presented one of the earliest attempts to build a visual-assistance wearable around the ESP32 microcontroller at IEEE AESPC 2021, demonstrating that a functional glasses-form-factor device could be constructed at a fraction of the cost of commercial alternatives. The ESP32 was selected for its dual-core Xtensa LX6 processor, 520 KB SRAM, integrated Wi-Fi and Bluetooth 4.2 radios, and a unit cost under five US dollars – characteristics that make it attractive for deployment in developing countries.

The prototype paired the ESP32 with a basic camera module and relayed captured frames wirelessly to a companion device for further processing. No on-device AI inference was performed; the contribution lies in proving the mechanical and electronic viability of an ESP32-based glasses platform rather than demonstrating intelligence. By quantifying the power consumption and thermal behaviour of a camera-equipped ESP32 in a wearable enclosure, the paper surfaced the constraints – Wi-Fi transmission power spikes and thermal throttling – that subsequent, more capable designs would need to manage. In this sense, Das et al. laid the groundwork on which the more sophisticated ESP32-CAM design of Babu et al. [4] would build directly.

2.2 Deep learning-based smart glasses on EPSON BT-300 (Lin et al., 2020)

Lin et al. [2] described a client-server smart glasses architecture at Indo-Taiwan ICAN 2020 built on the commercially available EPSON BT-300 Android-based headset. All computation was offloaded to a backend workstation equipped with an Intel Core i7-8700K processor and an NVIDIA GTX 1080Ti GPU. Images captured by the glasses' 5-megapixel camera were transmitted over a TCP socket to the server, where a YOLOv3 model – originally trained in Darknet and converted to a Keras/TensorFlow implementation – performed object detection against the 80-class COCO dataset. Detected labels and

bounding-box coordinates were packaged into a natural-language sentence and returned to the glasses, where the Android text-to-speech (TTS) engine read the description aloud.

A particularly thoughtful design element was the spatial description scheme: the image plane was divided into a 3 × 3 directional grid so the system could report not just what was detected but also where in the field of view it appeared. Across four evaluation categories – life goods, vehicles, animals, and food – the system achieved an average detection rate of 96.3% with transportation vehicles reaching 98.3% owing to their large, visually distinctive appearance. End-to-end latency averaged 3.788 seconds, dominated by network round-trip overhead. While high for real-time navigation, this latency suits the slower query-on-demand interaction model the system targets.

2.3 Third Eye: YOLOv5 and Kalman-filter sensor fusion (Uma et al., 2023)

Uma et al. [3] proposed the "Third Eye" system at IEEE SCEECS 2023, arguably the most feature-complete prototype among the five reviewed here. Built around a Raspberry Pi 2 single-board computer, the design deployed YOLOv5 for object and face detection – selected over EfficientDet and earlier YOLO variants for its speed-accuracy balance on the Raspberry Pi's ARM Cortex-A7 – and distinguished itself through multi-sensor data fusion. Ultrasonic time-of-flight measurements and structured-light depth-camera data were combined via a Kalman filter to produce a more stable obstacle distance estimate than either sensor alone.

The complementarity of these two sensors is worth understanding precisely. Ultrasonic transducers are reliable even in direct sunlight and against transparent or reflective surfaces, but their narrow field of view means they can miss wide obstacles. Depth cameras offer rich per-pixel range maps over a wide field but degrade in bright illumination and on mirrors. The Kalman filter continuously reweights both sources according to their estimated measurement uncertainty, yielding an estimate that inherits the illumination robustness of ultrasonics and the spatial resolution of the depth camera.

Beyond detection, the Third Eye incorporated a voice assistant built on PyAudio and Pyttsx3 enabling two-way spoken interaction. A hybrid solar-and-battery power supply and GPS-based positioning were also proposed. Results were qualitative in nature, presenting successful detections in indoor and outdoor scenarios without a controlled accuracy figure.

2.4 ESP32-CAM and OpenCV Haar-cascade glasses (Babu et al., 2025)

Babu et al. [4] presented a design at IEEE WAMS 2025 that prioritised affordability and independence from cloud services by basing the entire detection pipeline on an ESP32-CAM module paired with the classical Viola-Jones cascade classifier [10] as implemented in OpenCV. The Viola-Jones algorithm computes Haar-like rectangular features via the integral-image technique and applies an AdaBoost-trained cascade that progressively rejects background regions, concentrating computation on candidate face and object locations. This is computationally lighter than a deep network and runs on microcontroller-class hardware, though at the cost of reduced robustness to pose variation and limited category coverage.

The system complemented visual detection with a NEO6M GPS receiver for location tracking and emergency-location sharing with caregivers, using the ESP32's integrated Wi-Fi and Bluetooth radios without requiring a companion smartphone. An auto-sleep mode extended the LiPo battery's operational life. The authors reported successful face and object detection in both indoor and outdoor environments, though results were qualitative. The system's defining strength is its very low bill-of-

materials cost – the ESP32-CAM retails under three US dollars – and its complete independence from paid AI services, qualities that matter greatly for resource-constrained deployments.

2.5 Coral Edge TPU autonomous navigation glasses (Sharma et al., 2025)

Sharma et al. [5] described the most technically sophisticated system in this review at IEEE COMPUTINGCON 2025, targeting fully autonomous outdoor pedestrian navigation rather than on-demand object identification. Computation was split across two cooperating hardware units: a Google Coral Edge TPU embedded within the spectacle frame handled latency-critical perception – obstacle detection, object classification, and traffic-light colour and shape recognition – while a belt-mounted Raspberry Pi Zero W managed voice-command processing, Google Maps API route calculation, and turn-by-turn guidance.

The Edge TPU delivers per-inference latencies below ten milliseconds at approximately two watts, enabling always-on perception without rapid battery drain. Audio feedback was delivered through bone-conduction speakers rather than in-ear headphones, a deliberate safety-driven choice: preserving ambient sound awareness is critical for outdoor pedestrians who must react to vehicle horns and spoken warnings. Safe movement was formalised as a conjunction of three conditions – minimum obstacle clearance, maximum classification confidence, and a confirmed green traffic signal – before forward motion was permitted. Against a conventional ultrasonic cane and a smartphone navigation app, the system achieved 90% obstacle detection accuracy at under 100 ms latency, over 85% traffic-light recognition confidence, device weight below 60 g and approximately 5 hours of continuous battery life. Degraded performance under low-light conditions is the principal acknowledged limitation.

3 Comparative analysis

Table 1 consolidates the five reviewed systems across the dimensions most relevant for assistive-wearable evaluation. Three clear patterns emerge when reading across the table.

Table 1. Comparative summary of the five reviewed smart-glasses prototypes.

Study	Year	Platform	Detection	Sensing	Feedback
Das [1]	2021	ESP32	None	Camera	Display
Lin [2]	2020	BT-300 + server	YOLOv3	Camera	TTS
Uma [3]	2023	Raspberry Pi 2	YOLOv5	Multi-sensor	Earphone TTS
Babu [4]	2025	ESP32-CAM	Haar cascade	Camera, GPS	Speaker
Sharma [5]	2025	Coral TPU + RPi	Edge CNN	Camera, mic, GPS	Bone conduction

The first pattern is a structural shift from cloud-dependent to on-device inference. Lin et al.'s 2020 system required a desktop GPU for every query; by 2025, both Babu et al. and Sharma et al. executed all vision processing within the wearable hardware itself. This shift was driven by two practical pressures: the unacceptable latency of network roundtrips for navigation tasks, and the reliability requirement of assistive devices that must function outdoors where Wi-Fi is unavailable.

The second pattern is an evolution in detection methodology. Das et al. performed no detection at all; Babu et al. used the classical Viola-Jones cascade; the three remaining designs all employed deep

convolutional networks. This trajectory mirrors the broader shift in computer vision from hand-engineered features toward end-to-end learned representations.

The third pattern concerns feedback modality and its relationship to the target use case. Systems oriented toward indoor scene description are content with in-ear headphones or speakers. Sharma et al.'s navigation-focused design explicitly rejects in-ear audio, selecting bone-conduction speakers to preserve the environmental sound awareness that pedestrian safety demands. This distinction is not cosmetic – it reflects a fundamental difference in threat model between a stationary scene-querying task and walking outdoors.

4 Generalised system architecture

Despite hardware differences, all five systems share a common pipeline that can be abstracted into four layers, illustrated in Figure 1.

The Sensing Layer continuously acquires raw environmental data – a camera frame supplemented by ultrasonic ranging (Uma et al.), microphone audio (Uma et al., Sharma et al.), and GPS coordinates (Babu et al., Sharma et al.). The Processing / Inference Layer is where the AI computation occurs, whether through a Haar-cascade filter on an ESP32, a YOLOv5 network on a Raspberry Pi, or a neural network on a Coral TPU. The Decision / Language Layer translates raw detection outputs into a user-comprehensible form: a natural-language sentence describing object and direction, a navigation instruction, or a binary go/no-go movement decision. The Output / Feedback Layer delivers results through synthesised speech, an OLED display, or bone-conduction audio, with modality choice determined by the safety and operational requirements of the target use case.

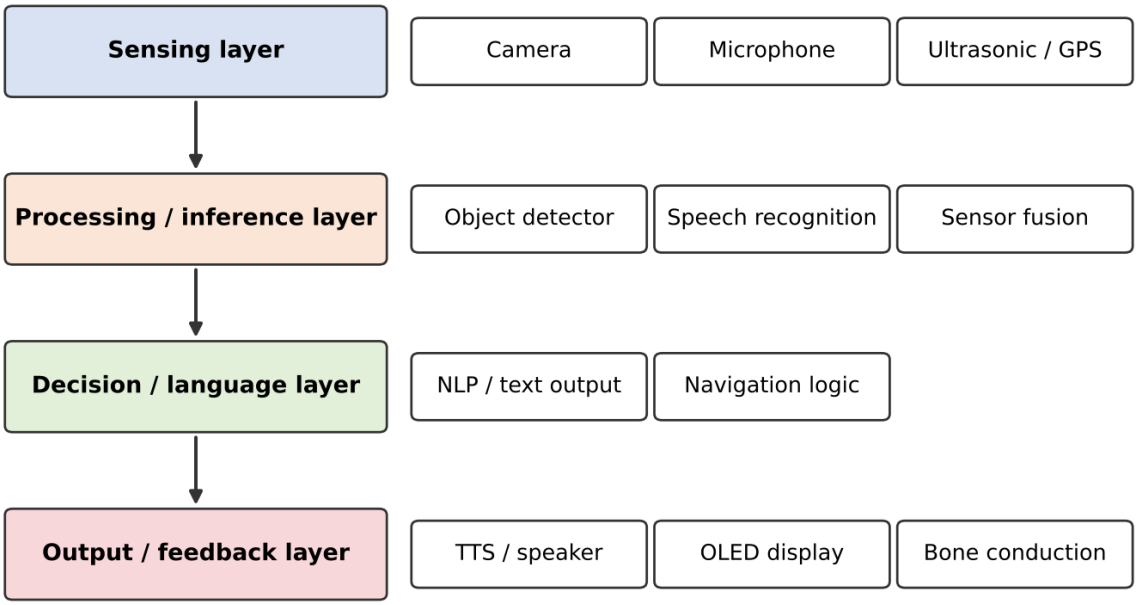

Figure 1. Generalised four-layer pipeline abstracted from the five reviewed systems. Cloud-offloaded designs (Lin et al.) host the Processing and Decision layers on a remote GPU server; fully on-device designs (Uma et al., Sharma et al.) embed all layers within the wearable hardware.

5 Core enabling technologies

5.1 Object detection algorithms

Figure 2 maps the detection algorithms in the five papers onto a two-branch taxonomy.

The Viola-Jones cascade classifier [10, 11], deployed by Babu et al., computes Haar-like rectangular features through the integral-image technique and applies an AdaBoost-trained cascade that discards background regions at progressively finer detail levels. Its primary advantage is speed on CPU-only hardware; its known weaknesses are limited robustness to pose variation and restricted category coverage beyond faces and a small set of trained object classes.

The YOLO (You Only Look Once) family [8] frames detection as a single regression over a grid of cells, jointly predicting bounding-box locations and class scores in one forward pass through the network. This single-pass design eliminates the region-proposal stage of two-stage detectors and enables real-time inference. YOLOv3 [9] added feature-pyramid multiscale prediction, improving small-object detection at the cost of higher memory usage – acceptable on the server-grade GPU used by Lin et al. but challenging on resource-constrained edge hardware. YOLOv5, selected by Uma et al., streamlined the architecture for faster inference on Raspberry Pi hardware. Sharma et al.'s Coral-optimised CNN represents the further step of designing the model for tensor-processing hardware from the outset, accepting a narrower task scope in exchange for minimal latency and power consumption.

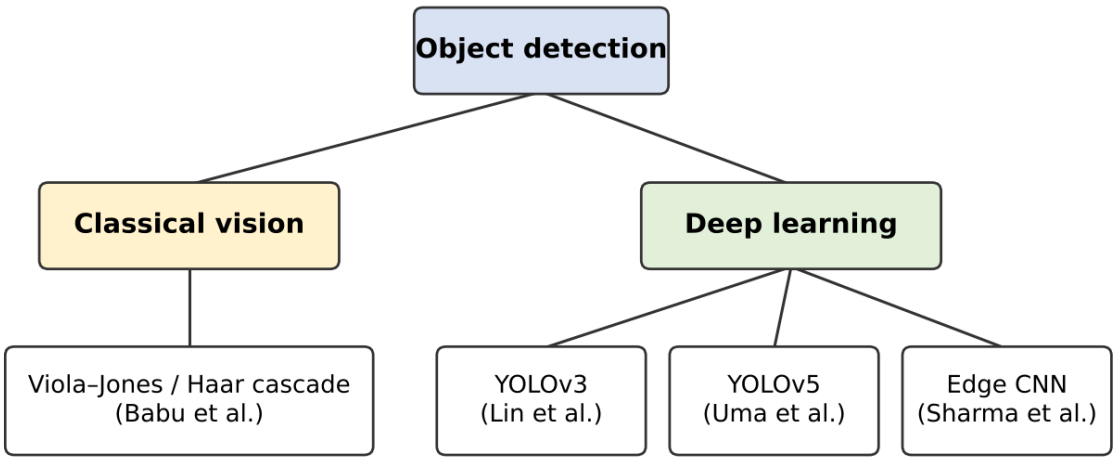

Figure 2. Taxonomy of detection algorithms across the five reviewed systems. Classical cascade methods require less memory and no GPU but offer narrower category coverage and lower robustness than deep learning alternatives.

5.2 Sensor fusion for distance estimation

Visual detection identifies what is present but provides limited depth information without a dedicated depth-estimation network. Uma et al. [3] addressed this by fusing ultrasonic and depth-camera measurements through a Kalman filter, which models obstacle distance as a Gaussian state and updates it at each sensor reading, assigning higher weight to whichever source is momentarily more reliable. The result inherits the illumination robustness of ultrasonics and the spatial resolution of the depth camera. Sharma et al. took a complementary approach, fusing the CNN's per-object confidence score with geometric distance to define a formal safety threshold, effectively treating model uncertainty as a second sensor channel.

5.3 Cloud versus on-device AI

The clearest architectural divide concerns where the heavy AI computation runs. Lin et al.'s server-GPU pipeline delivers high detection accuracy using a model impractical to run locally, but at a 3.788-second query latency and with a hard dependency on Wi-Fi connectivity. On-device designs – particularly Sharma et al.'s Coral TPU – achieve sub-100 ms latency and full connectivity independence, constrained only to models small enough to fit within the accelerator's memory budget. A hybrid strategy, processing latency-critical perception locally while delegating heavy language understanding to free-tier cloud APIs, represents an increasingly attractive middle ground as cloud AI costs have fallen [12].

5.4 Feedback modalities and user safety

Text-to-speech synthesis is the universal output mechanism across all five systems, yet the physical delivery hardware differs meaningfully. In-ear headphones and conventional speakers fully occlude ambient sound – acceptable for indoor or static-scene tasks. Sharma et al.'s adoption of bone-conduction speakers preserves environmental sound awareness and should be regarded as a design best practice for any navigation-focused assistive wearable deployed outdoors.

6 Performance evaluation

Among the five systems, only two published quantitative performance figures. Lin et al. reported a 96.3% average detection rate and 3.788 s average end-to-end latency across four object categories evaluated on a controlled image set. Sharma et al. reported 90% obstacle-detection accuracy and sub-100 ms latency evaluated outdoors in daylight conditions, along with over 85% traffic-light recognition confidence. The remaining three systems provided qualitative accounts of detection success without controlled benchmarks, as shown in Figure 3.

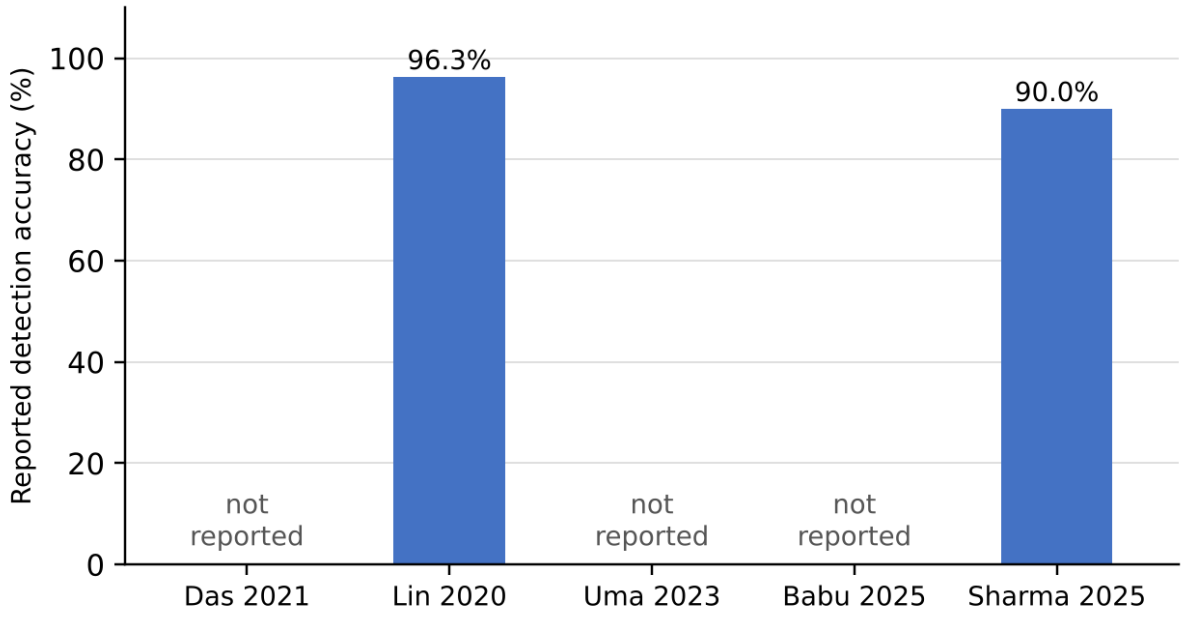

Figure 3. Reported detection accuracy across the five surveyed systems. Das et al. deployed no AI model; Uma et al. and Babu et al. report qualitative results without a controlled benchmark, so no accuracy figure is available for these three systems.

Latency comparison highlights the most dramatic difference in the dataset. Lin et al.'s client-server design incurred 3.788 seconds per query; Sharma et al.'s on-device Coral TPU achieved under 100 milliseconds – a 38-fold improvement. This gap is a direct consequence of the architectural choice between cloud offload and edge inference, making a compelling case that for time-critical navigation tasks, local processing is not merely preferable but necessary.

It must be acknowledged that comparing these figures directly is methodologically problematic: Lin et al. evaluated against a curated image set with four coarse categories, while Sharma et al. tested in live outdoor pedestrian environments. Neither study used a community-standard benchmark such as COCO AP. The absence of a shared evaluation framework is itself one of the most significant findings of this review.

7 Open challenges

Across five years and five distinct implementations, the literature converges on a set of recurring limitations that remain only partially resolved.

- **Low-light and adverse-illumination robustness.** Uma et al.'s depth camera and Sharma et al.'s vision pipeline both degrade at night or under poor lighting. Infrared, thermal, or low-light-enhanced imaging could help, but each adds cost, weight, or power beyond current prototype budgets.
- **Network dependence.** Lin et al.'s server-offloaded design cannot function without reliable Wi-Fi, yet users often need assistance where connectivity is unavailable. The shift toward on-device inference solves this, but at the cost of model capacity and detection category breadth.
- **Battery endurance.** Every reviewed system operates for two to five hours of continuous use, while practical daily deployment demands eight to ten. The tension between always-on inference power demands and glasses-frame weight and bulk constraints remains unresolved across all five designs.
- **Absence of real-user validation.** Apart from Sharma et al.'s benchmark against a conventional cane, none of the five studies involved actual blind or visually impaired participants. Lab-based detection assessment does not validate real-world utility in dynamic, unpredictable environments.
- **No standardised evaluation framework.** The lack of a shared benchmark makes it impossible to compare systems or track progress over time. A standardised assistive-navigation evaluation suite – analogous to COCO AP – would substantially accelerate meaningful comparison.

8 Future research directions

Several directions appear both technically tractable and likely to deliver meaningful improvements for end users.

- **Large language model integration.** Free-tier multimodal LLM APIs – Meta Llama 4 Scout via Groq, for instance – open the possibility of rich, conversational scene description rather than template-driven sentences. A hybrid design could run latency-critical detection locally on an Edge TPU while routing image-understanding queries to a cloud LLM [13].
- **Multimodal sensor fusion beyond visible light.** Adding infrared, thermal, or millimetre-wave radar sensing would directly address the low-light limitation seen across the reviewed systems.

Lightweight feature-level fusion with RGB camera data could deliver this robustness within a wearable's power envelope.

- **Adaptive power management.** Dynamically adjusting detection frequency, model complexity, or cloud-versus-local routing based on inferred activity – stationary, walking, or crossing a road – could extend battery life without sacrificing safety-critical responsiveness.
- **Custom PCB miniaturisation.** All reviewed prototypes used off-the-shelf boards on breadboards or in 3D-printed enclosures. Purpose-designed PCBs would cut device weight below 30g and improve durability, bringing these designs closer to a wearable glasses form factor.
- **Inclusive field trials and standardised evaluation.** Most urgently, the field needs large-scale trials with visually impaired participants, standardised tasks, and validated usability metrics, to determine which architectural choices deliver the greatest practical benefit.

9 Conclusion

This paper has surveyed five peer-reviewed conference publications on AI-powered smart glasses for blind and visually impaired users, spanning 2020 to 2025. The designs exhibit a clear architectural trajectory: from a proof-of-concept ESP32 image-relay prototype with no AI capability, through a cloud-dependent server-GPU pipeline incurring multi-second latency, to fully on-device deep learning systems delivering sub-100 millisecond responses on dedicated edge-AI accelerators.

A common four-layer pipeline emerges across all five systems – Sensing, Processing/Inference, Decision/Language, and Feedback – with the key differences lying in where inference runs and how sophisticated the AI components are at each layer. The field has converged on YOLO-family detectors for on-device deep learning, on bone-conduction audio as the safety-conscious feedback modality for outdoor navigation, and increasingly on free-tier cloud APIs as a cost-free path to large-model AI capability.

At the same time, persistent challenges remain: low-light robustness, battery endurance, the absence of standardised evaluation, and the scarcity of real-user field validation. Addressing these gaps – through hybrid edge-cloud architectures augmented by large language models, richer multimodal sensing, adaptive power management, and rigorous evaluation with visually impaired communities – represents the most impactful path forward for a technology with the potential to meaningfully improve independent daily living for billions of people worldwide.

References

[1] Das S, Saxena S, Rout NK. Low cost smart-glass using ESP32. In: *Proceedings of the IEEE 2nd International Conference on Applied Electromagnetics, Signal Processing, & Communication (AESPC)*; 2021; Bhubaneswar, India.

[2] Lin JY, Yao CC, Chiang CL, Chen MC, Wu MJ. Smart glasses application system for visually impaired people based on deep learning. In: *Proceedings of the Indo-Taiwan 2nd International Conference on Computing, Analytics and Networks (Indo-Taiwan ICAN)*; 2020. p. 202–206.

[3] Uma R, Harish PS, Ragav NG, Navin A. Third Eye – A smart wearable glass with deep learning technology powered with artificial intelligence for visually impaired people. In: *Proceedings of the IEEE International Students' Conference on Electrical, Electronics and Computer Science (SCEECS)*; 2023; Bhopal, India.

[4] Ramesh Babu BSSV, Jayakrishna G, Mahendra B, Gowtham M, Manasa ALSS, Chandrakanth M. AI-powered smart glasses for the blind using OpenCV & Python. In: *Proceedings of the IEEE Wireless Antenna and Microwave Symposium (WAMS)*; 2025; Visakhapatnam, India.

[5] Sharma P, Babu AS, Sadasivan M, Aroan Robert KT, Vasumathi MT, Ashok Kumar TA. Smart glasses for the blind: A real-time AI-driven wearable system for autonomous navigation. In: *Proceedings of the International Conference on Computing and Communications (COMPUTINGCON)*; 2025; Pune, India.

[6] World Health Organization. *Blindness and vision impairment* [Internet]. Fact sheet; 2023. https://www.who.int/news-room/fact-sheets/detail/blindness-and-visual-impairment

[7] Malhotra S, Prasad M, Vashist P, Kalaivani M, Gupta SK. Prevalence of blindness in India: A systematic review and meta-analysis. *The National Medical Journal of India*. 2019;32:325–333.

[8] Redmon J, Divvala S, Girshick R, Farhadi A. You only look once: Unified, real-time object detection. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*; 2016; Las Vegas, NV, USA. p. 779–788.

[9] Redmon J, Farhadi A. YOLOv3: An incremental improvement [Preprint]. *arXiv*:1804.02767; 2018.

[10] Viola P, Jones M. Rapid object detection using a boosted cascade of simple features. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*; 2001.

[11] Viola P, Jones MJ. Robust real-time face detection. *International Journal of Computer Vision*. 2004;57:137–154.

[12] Chen J, Ran X. Deep learning with edge computing: A review. *Proceedings of the IEEE*. 2019;107(8):1655–1674.

[13] Groq Inc. *Groq API documentation* [Internet]. 2024. https://console.groq.com/docs