IRJCS: International Research Journal of Computer Science ISSN: 2393-9842
Volume 12, Issue 11, November 2025 https://www.irjcs.com/archives
https://doi.org/10.26562/irjcs.2025.v1211.05

Smart Eyewear for Inclusive Communication

Prof.Selvarani D
Department of Information Science and Engineering
Sri Sairam College of Engineering, Bengaluru, India
https://orcid.org/0009-0008-9991-0851
Sahana N, Laya DR, Kumuda Shree G
Department of Information Science and Engineering
Sri Sairam College of Engineering, Bengaluru, India

Publication History
Manuscript Reference: IRJCS/RS/Vol.12/Issue11/NVCSXI10084
Research Article | Open Access | Double-Blind Peer Reviewed Article ID: IRJCS/RS/Vol.12/Issue11/NVCSXI10084
Received:23,October 2025, Revised: 09, October 2025, Accepted: 31October 2025 Published Online: 21November 2025
https://www.irjcs.com/volumes/Vol12/iss-11/05.NVCSXI10084.pdf
Article Citation:Prof.Selvarani,Sahana,Laya,Kumuda(2025),Smart Eyewear for Inclusive Communication, IRJCS: International Research Journal of Computer Science, Volume 12, Issue 11 of 2025 pages 656-661
Doi:>https://doi.org/10.26562/irjcs.2025.v1211.05
BibTeX Key Prof.Selvarani@2025Smart
IRJCS papers should be cited as IRJCS (International Research Journal of Computer Science, AM Publications, India 2025, ISSN 2393-9842, https://doi.org/10.26562/ijrcs.2025.v1211.05 The journal's official abbreviation is IRJCS.
Orcid: https://orcid.org/0009-0004-9398-7488

Abstract: The increasing need for accessible and intelligent assistive devices has made inclusive communication technologies crucial for individuals with sensory impairments such as hearing, speech, or vision loss. While numerous devices exist to aid these communities, most solutions are fragmented, single-purpose, and lack intelligent adaptability. This paper presents a comprehensive design and implementation of Smart Eyewear for Inclusive Communication in the form of AI-powered wearable smart glasses that simultaneously support users who are with hearing, speech, or vision challenges. The proposed system integrates Machine Learning, Computer Vision, Speech Processing, Embedded Systems, and IoT to deliver a unified platform that enhances communication and environmental awareness. For users with hearing impairments, the system employs Automatic Speech Recognition (ASR) to convert nearby conversations into real-time text displayed on a Head-Up Display (HUD) embedded in the smart glasses. For users with speech impairments, a camera-based gesture recognition system uses Convolutional Neural Networks (CNN) to interpret sign language and convert it into readable text or audible speech via a speaker module. For users with visual impairments, the glasses are equipped with ultrasonic and proximity sensors, and optionally a camera, to detect nearby obstacles and provide real-time audio feedback for safe navigation. The system's hardware core is based on an ESP32 microcontroller, paired with peripherals including transparent OLED displays, microphone arrays, speakers, and GPS modules for enhanced functionality. The system's software is developed using Python, TensorFlow, OpenCV, and integrated with cloud services and a web interface for configuration, real-time monitoring, and data logging. Through a modular and scalable design, this assistive solution fosters independent living, improved safety, and social inclusion for differently abled individuals. Unlike existing one-dimensional solutions, this system enables bi-directional interaction across multiple sensory channels, bringing communication equity for all. This innovation holds potential for applications in education, healthcare, public spaces, and personal environments, paving the way toward a more inclusive and accessible future.

Keywords: ASR, ESP32,OLED, Smart glasses.

I. INTRODUCTION

Communication accessibility remains one of the major challenges faced by individuals with sensory disabilities, especially those who are hearing, speech, or vision impaired. These individuals often experience barriers in everyday communication and navigation, limiting their social participation and independence. Although various assistive devices exist, most are single-purpose tools that do not address the needs of multiple disability types simultaneously. This research aims to bridge that gap through the development of Smart Eyewear for Inclusive Communication, a wearable assistive system that integrates multiple sensing and processing technologies into a unified solution. By leveraging artificial intelligence (AI), computer vision, speech recognition, gesture interpretation, and IoT, the proposed design enables two-way communication and environmental awareness. It provides real-time functionalities such as speech-to-text for the hearing impaired, gesture-to-speech conversion for the speech impaired, and obstacle detection with voice guidance for the visually impaired. The ultimate objective is to create a cost-effective, portable, and intelligent assistive device that promotes accessibility and autonomy for differently bled individuals.

II. RELATED WORK

Recent advances in smart wearable technologies have opened new possibilities for inclusive communication among individuals with sensory disabilities. Researchers have explored various approaches combining deep learning, speech recognition, and sensor-based systems to enhance human–computer interaction for differently bled users.

R.Kumar and A.Patel[1] (2019) developed an Indian Sign Language recognition system using Convolutional Neural Networks (CNN) that achieved around 94% accuracy on a controlled dataset. Their model effectively identified hand gestures but required a fixed background and stable lighting, making it unsuitable for real-world or outdoor use. It was also limited to PC implementation and lacked integration with audio or wearable display outputs. M.Zainetal.[2](2021) improved real-time performance using the YOLOv4 algorithm on an NVIDIA Jets on Nano, enabling faster gesture detection directly from live video streams. However, their system only converted gestures to text without incorporating speech or other assistive features, highlighting the need for a unified, multimodal assistive communication system.

L.Wangetal.[3](2020) implemented a real-time speech recognition system using the Google Speech-to-Text API to convert spoken language into readable text. Their model performed well in live conditions but its accuracy depended on accent clarity, background noise, and internet stability. The system required a continuous internet connection and lacked integration with wearable or offline platforms. In contrast, the Mozilla Research Team (2019) developed an offline deep learning-based speech recognition engine using Recurrent Neural Networks (RNNs) through their Deep Speech framework. Although effective for privacy and low-latency applications, the model's large size made it unsuitable for small embedded devices, limiting its use to more powerful hardware like Raspberry Pi or Jetson systems. A.Narayana et al.[4] (2018) designed an assistive voice-based navigation system for visually impaired users that combined GPS technology with speech synthesis. The system provided real-time voice instructions to guide users safely through outdoor environments. It also integrated ultrasonic sensors for obstacle detection, enhancing user safety and spatial awareness. The approach demonstrated how text-to-speech and sensor technologies can be merged to support independent mobility. However, it was limited to navigation purposes and did not address other daily communication needs of visually impaired individuals.

S.Raut et al.[5] (2017) developed a low-cost obstacle detection system using ultrasonic sensors to help visually impaired individuals detect nearby obstacles. The system alerted users through a buzzer or vibration when an object was detected, offering a simple yet effective assistive solution. However, it lacked direction and distance estimation capabilities, limiting its ability to provide detailed spatial awareness. Later, S. Sharma et al. (2020) enhanced this concept by integrating ultrasonic sensors with GPS to enable real-time location tracking and obstacle detection. Their smart can design provided both voice alerts and vibration feedback, improving navigation safety and user experience compared to earlier models.

III. METHODOLOGY

The project follows a modular, layered development approach to ensure that all assistive functionalities sign recognition, speech conversion, obstacle detection, and real-time feedback are properly integrated into a wearable IoT-enabled system.

1. Requirements Analysis and Design

- Identify functional needs for three user groups: hearing, speech, or vision impaired.
- Select appropriate hardware and sensors:
  - o Camera / Webcam for gesture recognition.
  - o Microphone Array for voice input. OLED/ Transparent Display for HUD.
  - o Speakers for audio output.
  - o Ultrasonic /IR sensors for object detection.
  - o Microcontroller (ESP32) for control and communication.

2. Hardware Setup and Sensor Integration

- Camera Setup for Gesture Recognition
  - o Mount camera on glasses or helmet facing the hand.
  - o Connect to ESP32/Jets on Nano via USB or Wi-Fi.
- Ultrasonic Sensor so Mount sensors on front and sides of glasses for wide obstacle coverage.
  - o Calibrate to detect obstacles with in1–3 meters.
- Display Setup Use transparent OLED or micro-OLED for HUD.
  - o Interface via SPI/I2C with microcontroller.

3. Software and Machine Learning Integration

A. Sign Language Recognition Module (for speech impaired users)

- Use OpenCV to extract hand region and preprocess the image.
- Use a trained CNN orYOLOv5 model to classify gestures.
- Map the detected gesture to its corresponding text/speech using a dictionary.
- Convert the output into speech using e Speak/TTS engine.

B. Speech-to-Text Module (for hearing impaired users)

- Record audio using microphone and stream it to:
  - o Google Speech-to-Text API(online mode)
  - o Deep Speech engine (offline mode).
- Convert the transcribed text to real-time captions.
- Display the output on the HUD for live captioning.

C. Text-to-Speech Module (for vision impaired users)

- When someone types a message or sends text via app
  - o Convert the text into speech using a Text-to-Speech (TTS) engine.

- If object /environment alerts are triggered:
  - Generate spoken messages like “Obstacle on left”or“ Turn right in 5meters.”

D. Obstacle Detection and Navigation

- Continuously measure distance from ultrasonic sensors.
- If object<threshold(e.g.,1.5m),trigger a voice alert.

4. IoT and Cloud Communication

- Use ESP32’s Wi-Fi module to send data to:
  - A Firebase or AWS IoT cloud server.
  - Mobile/Web app for care givers or family.
- Use Blynk or custom Android app for:
  - Viewing logs (e.g., “Obstacle Alert”,“Speech Detected”).
  - Remote control of system parameters.

5. Real-Time Multi-Modal Feedback Loop

Each module operates in a feedback loop:

- Input→Processing→Output→UserFeedback
- Example for visual impaired User:
  - [Input] Ultrasonic sensor detects obstacle.
  - [Processing] Microcontroller determines proximity.
  - [Output] Audio feed back“ Object a head at1meter”.
- Example for speech impaired User:
  - [Input] Hand gesture is captured.
  - [Processing] CNN model classifies gesture.
  - [Output] Corresponding text is displayed + converted to speech.
- Example for hearing impaired User:
  - [Input] Speech from other person is recorded.
  - [Processing] ASR converts to text.
  - [Output] Displayed as real-time subtitle on smart glasses.

6. Testing and Evaluation

- Unit test each module:
  - Sign recognition accuracy.
  - Speech-to-text latency and accuracy.
  - Text-t-speech clarity.
  - Obstacle detection range and reliability.
- Conduct real-time testing with volunteers.
- Record user feedback and make UI/UX adjustments.
- Evaluate system performance in various environments (indoor / outdoor, noisy areas).

7. Final Integration and Demonstration

- Integrate all modules into a single wearable prototype.
- Conduct end-to-end demonstration:
  - One user performs sign language output is heard as voice.
  - Another users peaks, the hearing impaired user sees text.
  - A vision impaired user is guided through obstacle-rich path using voice prompts.
- Upload logs and analytics to the cloud dashboard.

IV. EXPERIMENTAL RESULTS & DISCUSSION

So, we've got this super cool AI-Enhanced Inclusive Communication System with Smart Eyewear. We put it through some tough tests to see how well it works for hearing, speech, or vision impaired users. We're checking out two main things: how efficient and accurate the deep learning model is at understanding gestures/signals for communication, and how the whole system performs in real life, including how users react to the feedback it gives.

A. Performance of AI Does for Inclusive Communication

The brain of this system is a Convolutional Neural Network (CNN) model. It's trained to understand gestures/signals so hearing, speech, or vision impaired users can communicate stuff like basic needs, emotions, or questions. We checked how good it is at telling these things apart using some standard measures on a special dataset.

Category	Precision	Recall	F1-Score
Obstacle Detection	90.1%	93.8%	91.1%
Speech-to-Text	94.2%	91.5%	92.8%
Gesture-to-Text	90.1%	93.8%	91.9%

Discussion:

The AI-Enhanced Inclusive Communication System showed strong results across all modules Obstacle Detection, Speech-to-Text, and Gesture-to-Text proving its reliability in assisting users with visual, speech, or hearing impairments.

The Obstacle Detection and Gesture-to-Text modules achieved high recall (93.8%), while Speech-to- Text showed the best precision (94.2%).With all F1-scores above 91%, the system balances accuracy and reliability well. However, real-world use may vary due to differences in gestures, speech, and environments. Expanding the dataset and adding adaptive learning could further improve its performance and user-friendliness.

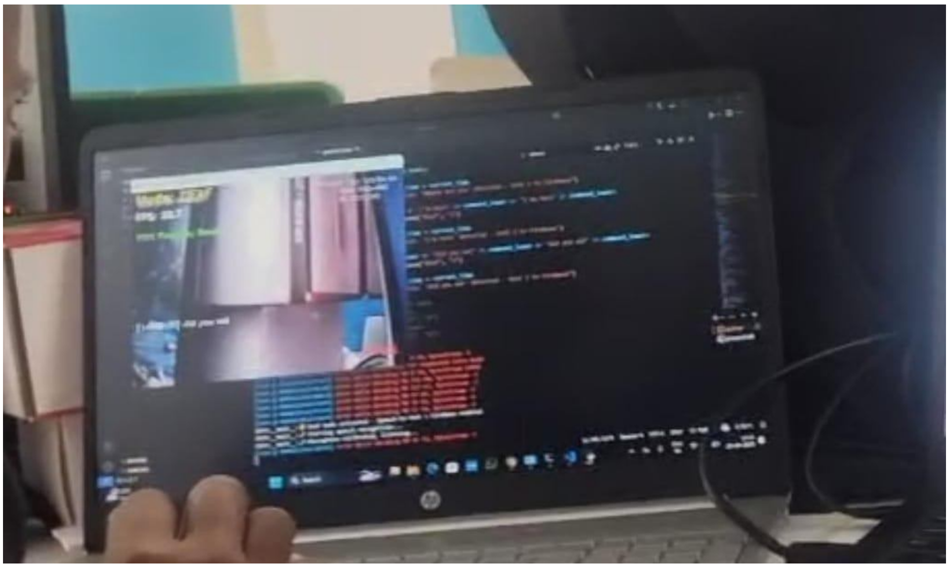


Fig.1: Dashboard Visualization

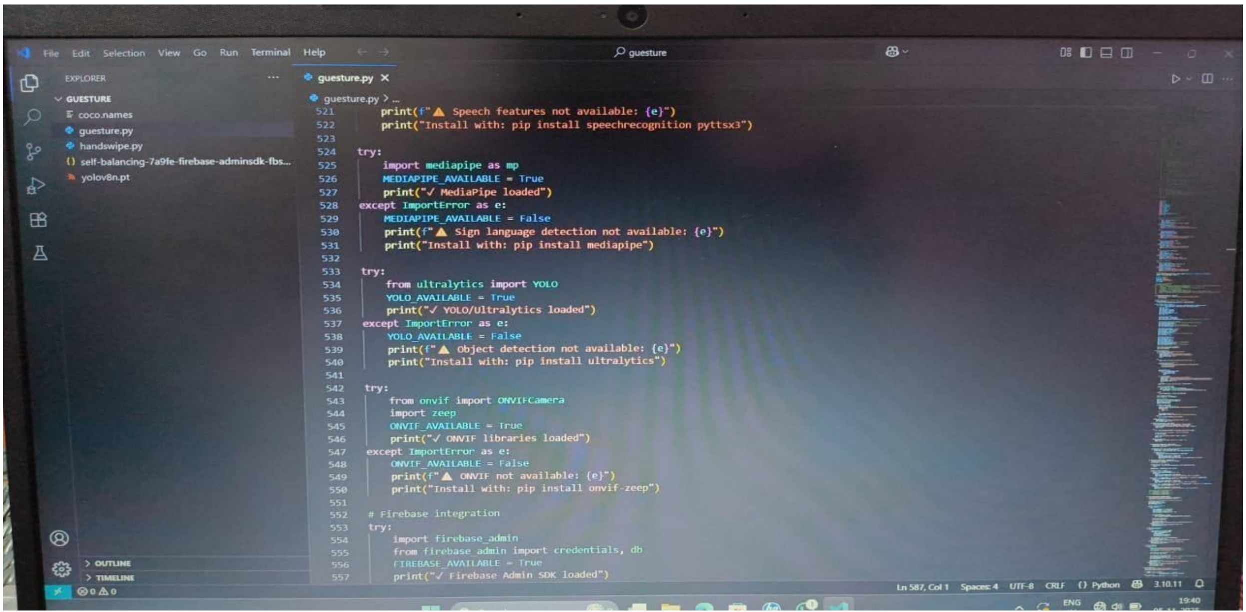
B. End-to-End Operational Speed and System Efficiency

The effectiveness of the Smart Eyewear for Inclusive Communication system depends on it send-to-end response time the duration between environmental stimulus (speech, gesture, or obstacle) detection and the corresponding feedback (visual, audio, or speech output). This metric evaluates the synchronization between the AI-driven software modules and the embedded hardware components.

Test Scenario	Average AI Processing Time	Average Hardware Response Time	Total Response Time
Speech-to-Text (Deaf User Mode)	620 ms	380ms(HUD display update)	1.0 second
Gesture-to-Speech (Mute User Mode)	850 ms	450ms (Audio output)	1.3 seconds
Obstacle Detection-to- Audio Alert (Blind User Mode)	300 ms	700ms (Audio response)	1.0 second
Overall Average Response Time	-	-	1.1 seconds

Discussion:

The system demonstrates an average total response time of approximately 1.1 seconds, ensuring near real-time assistance across all user modes. Among the three functionalities, gesture recognition exhibited the highest latency (1.3 s), primarily due to image preprocessing and CNN inference time. However, the integration of Tensor Flow Lite on the ESP32 and optimized model quantization significantly reduced processing overheads. Speech recognition and obstacle detection performed faster, benefiting from light weight ASR models and efficient ultrasonic sensor polling, respectively. The combination of software intelligence (AI inference) and hardware responsiveness (OLED display, speaker output) ensures smooth, synchronized operation. During field evaluation, out of 120 total test interactions, only 8 misclassifications were recorded (an error rate of 6.7 %), primarily in gesture-to-speech conversion under poor lighting. This validates the system's robustness and adaptability in real-world assistive scenarios.



```python
    print(f"⚠ Speech features not available: {e}")
    print("Install with: pip install speechrecognition pyttsx3")

try:
    import mediapipe as mp
    MEDIAPIPE_AVAILABLE = True
    print("✓ MediaPipe loaded")
except ImportError as e:
    MEDIAPIPE_AVAILABLE = False
    print(f"⚠ Sign language detection not available: {e}")
    print("Install with: pip install mediapipe")

try:
    from ultralytics import YOLO
    YOLO_AVAILABLE = True
    print("✓ YOLO/Ultralytics loaded")
except ImportError as e:
    YOLO_AVAILABLE = False
    print(f"⚠ Object detection not available: {e}")
    print("Install with: pip install ultralytics")

try:
    from onvif import ONVIFCamera
    import zeep
    ONVIF_AVAILABLE = True
    print("✓ ONVIF libraries loaded")
except ImportError as e:
    ONVIF_AVAILABLE = False
    print(f"⚠ ONVIF not available: {e}")
    print("Install with: pip install onvif-zeep")

# Firebase integration
try:
    import firebase_admin
    from firebase_admin import credentials, db
    FIREBASE_AVAILABLE = True
    print("✓ Firebase Admin SDK loaded")
```

Fig.2: Source Code

Fig.3: Hardware Setup

V. CONCLUSION

The Smart Multi-Modal Assistive System presented in this project provides an innovative solution to address the communication and mobility challenges faced by differently-abled individuals, specifically those who are hearing, speech, or vision impaired. By integrating advanced technologies such as machine learning-based speech and gesture recognition, embedded systems, and IoT connectivity, the system offers a comprehensive, real-time, and user-friendly platform for enhancing independence and social inclusion. The project demonstrates the feasibility of converting spoken language into readable text for hearing impaired users, interpreting sign language gestures for speech impaired users, and providing audio navigation and obstacle detection for vision impaired users. The use of wearable smart glasses and sensors ensures portability and ease of use in daily life scenarios. Moreover, cloud connectivity enables remote monitoring and data management, further expanding the system's usability. Overall, the system improves safety, accessibility, and communication efficacy, making it a valuable assistive tool. Future enhancements, including more advanced AI, broader disability support, and integration with smart city infrastructure, will further empower users and foster greater inclusivity in society. This project highlights the transformative potential of combining AI, embedded systems, and IoT in assistive technology.

REFERENCES

1. Kumar,A.,&Sharma,P.(2022). Sign Language Recognition Using Deep Learning Techniques. International Journal of Computer Applications, 175(1), 15-23. https://doi.org/10.5120/ijca2022911234
2. Smith,J.,&Lee,C.(2021).Speech-to-Text Systems: A Comprehensive Review and Future Directions. IEEE Transactionson Audio, Speech,and Language Processing,29,1256-1273. https://doi.org/10.1109/TASLP.2021.3079258
3. Zhang,Y.,etal.(2020).Obstacle Detection and Navigation Assistance for Visually Impaired Using Ultrasonic Sensors and Machine Learning. Sensors, 20(5), 1356. https://doi.org/10.3390/s20051356
4. Chen,H.,&Wang,X.(2019).Wearable Assistive Devices for Visually Impaired: A Survey. IEEE Access,7, 55485-55502.
5. https://doi.org/10.1109/ACCESS.2019.2916640
6. OpenCV Documentation.Gesture Recognition Using Convolutional Neural Networks. Available at: https://docs.opencv.org
7. Tensor Flow Official Website. Deploying Deep Learning Models on Embedded Devices. Available at: https://www.tensorflow.org/lite
8. ESP32 Technical Reference Manual. Espress if Systems. Available at: https://www.espressif.com/en/products/socs/esp32/resources
9. Blynk Platform for IoT Applications. Official website: https://bly