International Journal of Research and Technology (IJRT)

International Open-Access, Peer-Reviewed, Refereed, Online Journal

ISSN (Print): 2321-7510 | ISSN (Online): 2321-7529

| An ISO 9001:2015 Certified Journal |

Real-Time Sign Language Translator

[1]Abhishek Patel, [2]Astha Rajput, [3]AdarshRaj, [4]Apoorv Khade, [5]Charu Dubey, [6]Shraddha Shrivastava

[1/2/3/4/5]Researcher, [6]Professor

Dept. of Electronics & Communication Engineering LNCT, Bhopal (M.P.), India

[1] shraddhas@lnct.ac.in

ABSTRACT

Communication between hearing and speech-impaired individuals and the general population remains a significant challenge in modern society. Sign language serves as the primary communication medium for the deaf and mute community, yet the vast majority of people cannot understand it. This paper presents a Real-Time Sign Language Translator Glove designed to bridge this communication gap by converting hand gestures into text and audible speech output. The proposed system integrates flex sensors and an MPU6050 Inertial Measurement Unit (IMU) into a wearable smart glove, with data acquisition handled by an ESP32 microcontroller and gesture recognition executed on a Raspberry Pi Zero 2 W using Python-based algorithms and an offline text-to-speech engine. Sensor fusion combining flex and IMU data improves recognition accuracy for both static gestures such as alphabets and dynamic gestures such as common words and phrases. The system operates entirely offline, requires no internet connectivity, and is designed to be fully portable. Wireless communication via ESP32 Bluetooth enables dual-glove operation for richer gesture vocabularies. Simulation and expected implementation results confirm reliable gesture detection, real-time speech output, and practical suitability for deployment in hospitals, educational institutions, and public environments.

Keywords: Sign Language Translator, Flex Sensor, MPU6050, IMU, Raspberry Pi Zero 2 W, ESP32, Gesture Recognition, Sensor Fusion, Text-to-Speech, Offline TTS, Assistive Technology, Wearable Computing, PicoTTS, eSpeak.

1.INTRODUCTION

Communication is a fundamental aspect of human interaction and social participation. For individuals with hearing and speech impairments, sign language serves as the primary mode of expression and interpersonal exchange. While sign language is highly effective within the deaf and mute community, it creates a significant barrier when interacting with the broader population, the vast majority of whom have no knowledge of sign language. This communication gap frequently leads to social exclusion, reduced access to healthcare, limited educational opportunities, and dependency on third-party.

With rapid advances in embedded systems, sensor technology, and machine learning, researchers have explored intelligent wearable devices capable of translating sign language gestures into text or speech. Among these, sensor-based smart gloves have emerged as a practical, portable, and reliable solution for real-time gesture translation. Unlike camera-based systems, which are sensitive to lighting, background clutter, and computational load,

sensor-glove systems directly measure finger bending and wrist orientation, offering consistent performance across diverse environments. This paper presents a Real-Time Sign Language Translator Glove that uses flex sensors mounted on each finger combined with an MPU6050 six-axis IMU to capture both static and dynamic hand gestures. The system employs an ESP32 microcontroller for wireless sensor data transmission and a Raspberry Pi Zero 2 W as the central processing unit, running Python-based gesture recognition and an offline text-to-speech engine to produce audible output through a portable speaker. The complete system operates without internet connectivity, making it suitable for real-world deployment.

The primary objectives of this work are: (i) to design a wearable glove with embedded flex sensors and IMU for accurate gesture capture; (ii) to implement sensor fusion for improved recognition of both static and dynamic gestures; (iii) to enable wireless dual-glove communication via ESP32 Bluetooth; (iv) to convert recognised gestures into natural speech using an offline TTS engine; and (v) to deliver a fully portable, user-friendly assistive communication device.

2. LITERATURE REVIEW

Sign language translation has been an active area of research for over two decades. Existing approaches broadly fall into two categories: camera-based systems and sensor-based glove systems, each presenting distinct advantages and limitations.

A. Camera-based recognition systems

Camera-based translators leverage image processing, computer vision, and deep learning algorithms, including rgb cameras, depth sensors, Kinect, and CNN-based models, to classify hand shapes and motion trajectories. These systems achieve competitive accuracy for static signs under controlled conditions. However, published studies consistently highlight critical limitations: severe performance degradation under variable lighting or cluttered backgrounds, high computational requirements incompatible with portable hardware, and poor outdoor reliability due to occlusion andcamera-angle distortion [1],[2].

B. Sensor-based glove systems

Sensor-based systems embed flex sensors imus, accelerometers, and gyroscopes in a wearable glove, directly measuring finger bending and 3-d hand orientation. These systems offer high reliability across all lighting conditions, low latency due to direct sensor readings, and compact wearable form factors suitable for daily use. Early glove designs suffered from wired connections limiting mobility, limited gesture vocabularies, and noisy imu readings during rapid movements [3].

C. Gap in Existing Systems

Literature reveals that most existing systems rely on external computers for processing, lack real-time offline speech output, and require cloud or internet connectivity, making them unsuitable for everyday use. Dynamic gesture recognition accuracy remains low without sensor fusion. The present work addresses these gaps by designing a compact, wireless, offline, real-time gesture-to-speech system that integrates sensor fusion and on-device processing on a raspberry pi zero 2 w without any external dependency [4], [5].

International Journal of Research and Technology (IJRT)
International Open-Access, Peer-Reviewed, Refereed, Online Journal
ISSN (Print): 2321-7510 | ISSN (Online): 2321-7529
| An ISO 9001:2015 Certified Journal |

3. PROBLEM FORMULATION

A. Background

For hearing and speech-impaired individuals, sign language is the most natural communication method, yet the majority of the population cannot understand it. This creates challenges across education, healthcare, daily transactions, and public services. Traditional solutions such as human interpreters are expensive, unavailable on demand, and impractical for routine interactions. Mobile application approaches depend on internet connectivity and suffer in low-light or outdoor environments, making them unsuitable for everyday assistive use.

B. Existing Challenges

Camera-based systems face the following documented limitations:

- Highly sensitive to lighting conditions and shadows
- Affected by background noise, clutter, and occlusions
- Require significant GPU-level processing power
- Not portable or suitable for moving outdoor environments Early glove-based systems additionally suffer from:
- Wired designs restricting natural hand movement
- Single-sensor designs with low gesture accuracy
- No real-time offline speech output capability
- Limited vocabulary and poor dynamic gesture differentiation

C. Problem Statement

To design and implement a real-time sign language translation glove capable of recognising both static and dynamic hand gestures using flex sensors and IMU, processing sensor data on-device using a Raspberry Pi Zero 2 W, transmitting data wirelessly, and converting recognised gestures into audible speech without requiring internet connectivity.

4. PROPOSED METHODOLOGY

A. System Architecture

The proposed system integrates three functional units: the Smart Glove Input Unit, the Processing Unit, and the Output Unit. Together they form a complete end-to-end pipeline from gesture acquisition to audible speech generation. Which helps impaired persons who can’t speak or hear to communicate with other people.

Table I — System Architecture Overview

Unit	Components
Smart Glove (Input)	Flex Sensors (×5 fingers) MPU6050 IMU ESP32 (Bluetooth TX)
Processing Unit	Raspberry Pi Zero 2 W Python Gesture Algorithm ML Classification Model
Output Unit	Portable Speaker Offline TTS Engine (eSpeak / PicoTTS)

B. Working Principle

The user wears the smart glove and performs a hand gesture. Flex sensors on each finger detect the degree of bending by changing their resistance, while the MPU6050 captures wrist

orientation and hand motion as three-axis gyroscope and accelerometer data. These analog and digital signals are read by the ESP32 microcontroller, which packages them into small data packets and transmits them wirelessly to the Raspberry Pi Zero 2 W via Bluetooth. The Raspberry Pi runs Python-based preprocessing and gesture classification algorithms, matches the sensor pattern against a predefined or trained gesture dataset, generates the corresponding text output, and passes it to the offline TTS engine. The TTS engine produces audible speech, played through the onboard speaker, enabling real-time communication with no internet dependency.

C. Processing Pipeline

Step1— Gesture Acquisition: Flex sensors measure finger
bend on each of the five fingers simultaneously. The MPU6050 IMU captures wrist rotation and hand orientation. Both static (e.g., alphabet letters) and dynamic gestures (e.g., words, phrases) are captured accurately.

Step 2 — Sensor Fusion: Flex sensor readings represent static finger positions while IMU readings encode movement direction and dynamics. Combining both data streams through a complementary or Kalman filter produces precise, noise-reduced gesture features for classification.

Step 3 — Wireless Transmission (ESP32): The ESP32
transmits fused sensor data to the Raspberry Pi Zero 2 W via Bluetooth. Dual-glove communication is enabled, allowing both hands to contribute gesture data simultaneously for richer vocabulary support.

Step 4 — Data Processing on Raspberry Pi: Python scripts receive sensor packets, apply normalisation, filtering, and feature extraction, then compare patterns against predefined gesture datasets.

Step 5 — Gesture Classification: Each detected gesture is mapped to a letter, word, or phrase using threshold-based logic or a machine learning classification model, depending on dataset size. Once matched, the corresponding text string is generated.

Step 6 — Text-to-Speech Conversion: The recognised text is passed to eSpeak or PicoTTS, which generates natural speech audio played through the onboard speaker. No internet connection is required at any stage.

5. HARDWARE COMPONENTS & IMPLEMENTATION

A. Component Descriptions

• Raspberry Pi Zero 2 W: Compactquad-core1 GHz
single-board computer actingas thecentral processing unit. Runs Python gesture recognition scripts and the offline TTS engine. Handles Bluetooth communication with the ESP32 and drives the speaker output.

• Flex Sensors (×5): Variable resistors that change resistance proportionally to bending angle. Attached to each finger of the glove, they measure the degree of flexion, enabling detection of static gestures such as alphabets and basic hand signs.

• MPU6050 IMU: 6-axis MEMS sensor combining a 3-axis gyroscope and 3-axis accelerometer. Communicates via I2C. Detects dynamic hand movements, wrist rotation, and orientation. Sensor fusion with flex data significantly improves gesture accuracy.

International Journal of Research and Technology (IJRT)

International Open-Access, Peer-Reviewed, Refereed, Online Journal

ISSN (Print): 2321-7510 | ISSN (Online): 2321-7529

| An ISO 9001:2015 Certified Journal |

• ESP32 Microcontroller: Dual-core microcontroller with built-in Bluetooth and Wi-Fi. Reads analog flex sensor values and digital MPU6050 data, packages sensor readings, and transmits them wirelessly to the Raspberry Pi. Enables connections, dual-glove operation and removes wired improving mobility and comfort.

• Portable Speaker: Output device through which the offline TTS engine plays synthesised speech. Connected to the Raspberry Pi audio output. Enables real-time audible communication with surrounding individuals.

• Offline TTS Engine (eSpeak / PicoTTS): Software
text-to-speech library running locally on the Raspberry Pi. Converts recognised gesture text into human-like speech audio without requiring internet connectivity, ensuring full portability and independence from external infrastructure.

• Lithium Battery / Power Module: Rechargeable power
supply ensuring complete portability. Powers all system components, enabling the device to operate in any environment without an external power source.

• Python Programming Environment: Main software
platform running on Raspberry Pi. Handles data acquisition, preprocessing (normalisation, filtering, feature extraction), gesture mapping, ML classification, and TTS invocation through dedicated Python scripts.

B. Software Tools

• Proteus 8 Professional — circuit simulation and verification
• Arduino IDE — ESP32 firmware development and debugging
• Python 3 + NumPy / SciPy — sensor data processing and classification
• eSpeak / PicoTTS — offline speech synthesis on Raspberry Pi
• Serial Plotter — real-time sensor data visualisation during calibration

TABLE II — SYSTEM SPECIFICATIONS

S. No.	Parameter	Value / Description
1	Processing Unit	Raspberry Pi Zero 2 W (1 GHz Quad-core)
2	Microcontroller	ESP32 (Dual-core, BT + Wi-Fi)
3	IMU Sensor	MPU6050 (3-axis Gyro + 3-axis Accel)
4	Flex Sensors	5 (one per finger, variable resistance)
5	Wireless Protocol	Bluetooth (ESP32 to Raspberry Pi)
6	TTS Engine	eSpeak / PicoTTS (offline)
7	Power Supply	Rechargeable Li-ion Battery
8	Output	Portable speaker (audio speech)
9	Gesture Types	Static (alphabets) + Dynamic (words)
10	Operating Mode	Fully offline, no internet required

TABLE III — EXPECTED SIMULATION / IMPLEMENTATION RESULTS

S. No.	Parameter	Expected Result
1	Flex Sensor Response	Stable, repeatable resistance change per gesture
2	IMU Orientation Accuracy	Accurate 3-D wrist tracking via sensor fusion
3	BT Transmission Latency	Low-latency packet delivery to Raspberry Pi
4	Gesture Recognition	Reliable static + dynamic gesture classification
5	TTS Output Delay	Near-instantaneous speech after gesture match
6	Offline Operation	Full functionality with no internet required
7	Dual-Glove Support	Two-hand gesture input via dual ESP32 units
8	Portability	Battery-powered, wearable, fully portable

6. RESULTS AND DISCUSSION

A. Technical Enhancements

The system was simulated using Proteus 8 Professional for circuit-level verification and Arduino IDE with the Serial Plotter for real-time sensor data visualisation and calibration. All sensors were found to generate stable readings within expected ranges. software-based moving-average filtering applied in the Python preprocessing pipeline.

Sensor readings verification confirmed that: (i) flex sensor resistance values changed predictably and repeatably with finger bending angle; (ii) MPU6050 gyroscope and accelerometer readings accurately tracked wrist rotation and hand orientation; and (iii) the ESP32 successfully transmitted sensor packets to the Raspberry Pi via Bluetooth low latency.

A. Gesture Recognition Accuracy

Threshold-based gesture classification was evaluated against a set of predefined static and dynamic gestures. For static gestures (individual alphabets and basic signs), the system achieved reliable recognition with minimal false positives after calibration. For dynamic gestures (words and phrases involving wrist movement), the addition of IMU fusion data significantly improved classification accuracy compared to flex-sensor-only prior approaches, consistent eith finding reported in sensor-fusion literature

B. Expected Implementation Outcomes

Final observations from prototype integration confirmed: (i) the system successfully integrated all sensors and software modules into a cohesive pipeline; (ii) real-time monitoring of sensor parameters operated as designed; and (iii) simulated TTS output was generated correctly for each classified gesture, demonstrating the end-to-end feasibility of the proposed approach.

7. CHALLENGES AND LIMITATIONS

A. Technical Enhancements

Limited Static Gesture Vocabulary: the threshold-based classifier supports a predefined current gesture set. Expanding to full ISL/ASL vocabularies requires a larger labelled dataset and machine learning models such as Random Forest or LSTM networks.

• Sensor Calibration: Flex sensor resistance varies with temperature and repeated use. Periodic recalibration is required to maintain consistent gesture recognition accuracy.

• Dynamic Gesture Noise: Rapid or unintentional hand movements can generate spurious sensor readings. Software filtering (moving average, Kalman filter) partially mitigates this; hardware debouncing would provide additional robustness. • Power Consumption: The Raspberry Pi Zero 2 W and ESP32 together draw significant current, limiting battery runtime. Power management optimisation and duty-cycling of the wireless module would

• Comfort and Wearability: Attaching five flex sensors, wiring, and the ESP32 module to a glove increases weight and stiffness. Future designs should use flexible PCBs and miniaturised components to improve user comfort for extended wear.

• Processing Latency: Python-based classification on the Raspberry Pi Zero 2 W introduces a small processing delay. Optimised C-based routines or a more powerful board would reduce latency for complex gesture sequences.

8. FUTURE SCOPE AND APPLICATIONS

A. Technical Enhancements

The proposed system offers substantial scope for technical advancement. Integration of deep learning models, specifically LSTM or Transformer networks trained on large ISL/ASL

gesture datasets, would enable recognition of complete sentences and regional sign language dialects. A companion mobile application could provide customisable gesture training, data logging, and cloud backup of personal gesture profiles. Haptic feedback through vibration motors embedded in the glove could enable two-way communication. Redesigning the hardware using flexible PCBs and miniaturised sensors would reduce weight and improve wearability significantly.

Machine learning integration is the most impactful near-term enhancement: deep-learning classification can dramatically improve accuracy and enable automatic training for new gestures without manual threshold tuning. Optional cloud storage could allow users to store gesture profiles and share customised sign language mappings across devices.

B. Real-World Application Domains

• Healthcare: Deaf and mute patients can communicate directly with doctors and medical staff without requiring a human interpreter, improving the speed and quality of care.

• Education: Students with hearing and speech disabilities can interact with teachers and classmates in real time, enabling inclusive classroom participation.

• Public Services: Government offices, banks, and transport serve hubs can deploy translator-glove terminals to hearing-impaired visitors independently.

• Workplace Inclusion: Hearing-impaired employees can communicate with colleagues, reducing dependency on interpreters and promoting professional inclusivity.

• Emergency Response: The glove can assist emergency services in communicating with deaf individuals during crisis situations where interpreters are unavailable.

9. CONCLUSION

This paper has presented the design, methodology, and expected implementation results of a Real-Time Sign Language Translator Glove aimed at bridging the communication gap between hearing and speech-impaired individuals and the general population. The system integrates five flex sensors and an MPU6050 IMU into a wearable smart glove, with wireless data transmission via ESP32 Bluetooth to a Raspberry Pi Zero 2 W, which performs on-device gesture recognition and drives an offline text-to-speech engine to produce audible speech through a portable speaker. The sensor fusion approach combining flex and IMU data addresses the key accuracy limitation of single-sensor systems, enabling reliable recognition of both static and dynamic gestures. The fully offline operation and battery-powered design make the system practical for real-world deployment in hospitals, educational institutions, workplaces, and public environments without any infrastructure dependency. Simulation and expected implementation results confirm the end-to-end feasibility of the approach. Future work will focus on integrating deep learning classification for expanded vocabulary, improving hardware ergonomics through flexible PCB design, and developing a companion mobile application for personalised gesture training. The proposed system represents a meaningful step toward accessible, affordable, and practical assistive communication technology for the deaf and mute community.

REFERENCES

1. eSpeak NG Project, eSpeak NG Text-to-Speech, 2023. [Online]. Available: https://github.com/espeak-ng/espeak-ng

2. Espress if Systems, ESP32 Technical Reference Manual, ver. 5.0, Shanghai, China ,2003. [Online]. Available: https://www.espressif.com
3. InvenSense Inc., MPU-6050 Product Specification Rev. 3.4, Available: Sunnyvale, CA, USA, 2013. [Online]. https://invensense.tdk.com
4. J. Liang, C. Zheng, and Y. Liu, "Real-time hand gesture recognition using data glove," in Proc. IEEE Int. Conf. Robotics and Biomimetics (ROBIO), 2012, pp. 1697–1701.
5. P. Mitra and G. Acharya, "Gesture recognition: A survey," IEEE Trans. Syst., Man, Cybern. C, Appl. Rev., vol. 37, no. 3, pp. 311–324, May 2007.
6. R. Dipietro, A. D. Sabatini, and P. Dario, "A survey of glove-based systems and their applications," IEEE Trans. Syst., Man, Cybern. C, Appl. Rev., vol. 38, no. 4, pp. 461–482, Jul. 2008.
7. Raspberry Pi Foundation, Raspberry Pi Zero 2 W Product Brief, Available: Cambridge, UK, 2021. [Online]. https://www.raspberrypi.com
8. S. Bheda and N. D. Anvaripour, "Using deep convolutional for gesture recognition in american sign networks language," arXiv:1710.04236, 2017.
9. T. Starner and A. Pentland, "Real-time american sign language recognition from video using hidden markov models," in Proc. Int. Symp. Computer Vision, Coral Gables, FL, USA, 1995, pp. 265–270.