IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

DEVELOPMENT OF SIGN LANGUAGE TRANSLATOR BASED ON GESTURES TO TEXT AND SPEECH

Noor Ayesha,
Assistant professor
Dept. of Electronics & Communication Engineering
PES College of Engineering
Mandya, Karnataka, India

Mohammed Adnan A
Dept. of Electronics & Communication Engineering
PES College of Engineering
Mandya, Karnataka, India

Muskan Naaz
Dept. of Electronics & Communication Engineering PES College of Engineering Mandya, Karnataka, India

Srushti Y Kotagi
Dept. of Electronics & Communication Engineering
PES College of Engineering
Mandya, Karnataka, India

Mohammed Idrees Khan
Dept. of Electronics & Communication Engineering
PES College of Engineering
Mandya, Karnataka, India

Abstract:
In today's interconnected world, deaf individuals often face significant communication barriers. This project presents a novel system designed to bridge these gaps by interpreting complex hand gestures into understandable communication. The core of the system utilizes five flex sensors to capture intricate joint movements, interfaced with an Arduino microcontroller for precise data acquisition. Strategic pin management techniques are employed to efficiently handle all sensor inputs. The system is engineered to recognize 32 distinct hand gestures, translating them into actionable information. This interpreted data is then displayed in real-time on a mobile application via Bluetooth connectivity. Crucially, an ISD voice recording module coupled with speakers produces corresponding spoken words for each recognized gesture, providing a multi-modal communication aid. This system offers a robust solution for sign language interpretation, with significant potential for applications in assistive technology, rehabilitation, and intuitive human-computer interaction.

Keywords:
Flex Sensors, Gesture Recognition, Sign Language Interpretation, Arduino, Bluetooth Communication, Mobile Application, Voice Output, Assistive Technology, ISD Module.

INTRODUCTION

Sign language serves as a vital communication tool for the hearing-impaired community, enabling individuals to convey complex ideas through hand gestures. However, a language barrier often exists between sign language users and non-signers. To bridge this gap, the development of a sign language translator based on gestures-to-words and speech It offers a promising solution. This project aims to create a system that interprets sign language gestures and translates them into corresponding words or sentences and audio in real-time. such as gloves embedded with sensors, detect hand movements and orientations. These gestures are captured by sensors like accelerometers and gyroscopes, which are then processed using machine learning algorithms to identify patterns. The recognized gestures are mapped to specific words or phrases and transmitted to a mobile device or display unit and voice through speakers.

This gesture-to-word and voice translation system offers a portable, user-friendly solution for improving communication between the hearing-impaired community and the rest of society. Through the integration of advanced gesture recognition, this system aims to enhance inclusivity.
This project leverages wearable technology to interpret hand gestures and convert them into both text and voice outputs, enhancing accessibility for individuals with speech or hearing impairments. By in cooperating real-time processing, the system ensures seamless communication through gestures.

LITERATURE SURVEY

1. Gesture Recognition System Using Flex, **Sensor Authors:** A. Kumar, S. Singh
Journal: IEEE Transactions on Human-Machine Systems
Year: 2020
Observations Made: This paper presents a gesture recognition system that uses flex sensors to track finger movements. The system translates gestures into actions, which are interpreted by a microcontroller. Results showed high accuracy in recognizing hand gestures for controlling various electronic devices.
Applicability: The use of flex sensors in this study is relevant to your system design, as it demonstration shows sensors can be integrated with microcontrollers to recognize hand gestures, making it ideal for gesture-controlled devices or robotics.

2. Flex Sensor-Based Hand Gesture Recognition, **Authors**: P. Mishra, R. Sharma
Journal: International Journal of Intelligent Systems
Year: 2021

Observations Made: This paper introduces a glove-based gesture recognition system that utilizes flex sensors attached to each finger. The study highlights the conversion of flex sensor outputs into gesture inputs for gaming and virtual reality applications. The microcontroller processes sensor signals and triggers corresponding commands in real- time.
Applicability: The methodology of using multiple flex sensors to capture complex finger movements and process them using a microcontroller can be applied to your project for gesture-controlled applications or prosthetic devices.

3. Speech Recognition and Flex Sensor System for Hand Gesture Control, **Authors:** V. Patel, K. Rao
Journal: Journal of Assistive Technologies
Year: 2022
Observations Made: This research integrates flex sensors with speech recognition to aid physically challenged individuals in controlling electronic devices using both gestures and voice commands. The system employs a microcontroller to process flex sensor data and speech inputs, providing two levels of control for users.
Applicability: This dual-mode control using flex sensors and speech recognition could enhance the versatility of your design, allowing for both gesture-based and audio-based interactions, which is especially useful in assistive technology.

4. Real-Time Gesture Recognition Using Flex Sensors and Microcontrollers, **Authors:** M. Zhang, J. Lee
Journal: IEEE Access
Year: 2019
Observations Made: The paper discusses a real-time gesture recognition system that employs flex sensors to monitor finger and wrist movements. The data is processed by an Aurdino microcontroller, which maps the flex values to pre-defined gestures. The system provides quick response times suitable for human-computer interaction.
Applicability: The real-time processing and quick response times in this study are critical features that can be applied to your project to ensure the system responds immediately to user gestures.

5. Flex Sensor Applications in Wearable Electronics, **Authors:** N. Gupta, P. Joshi
Journal: Wearable Technologies Journal
Year: 2020
Observations Made: This study in vest gates the application of flex sensors in wearable electronics, including smart gloves and health monitoring systems. The flex sensors were used to detect muscle and joint movements, which were processed by a microcontroller to monitor the physical condition of the wearer.
Applicability: Integrating flex sensors into wearable systems, as demonstrated in this paper, is relevant to your system, as it shows how the sensors can be used to detect movements and triggers pacifications or outputs, similar to controlling devices using hand gestures

6. A Microcontroller-Based Gesture Control System for Robotics, **Authors:** R Singh, A Nair
Journal: International Journal of Robotics Research
Year: 2021
Observations Made: This paper discusses a system where flex sensors are used to control robotic movements. The system allows users to control a robotic arm by detecting hand gestures through flex sensors. The microcontroller processes the sensor data and translates it into commands for the robot's actuators.
Applicability: The robotic control system described here can be directly related to your project, as it outlines show flex sensor input scan control external devices like robotic arms, which could be applied in industrial automation or rehabilitation technologies

7. Enhancing Gesture Recognition with Machine Learning, **Authors:** H. Lee, K. Kim
Journal: Journal of Advanced Computing
Year: 2022
Observations Made: This study explores how machine learning algorithms can enhance the accuracy of gesture recognition systems using flex sensors. By training models on the flex sensor data, the system was able to improve recognition rates and reduce errors in gesture classification.
Applicability: Incorporating machine learning for gesture classification could improve the accuracy of your system, making it more responsive to a wide range of gestures and reducing false positives.

8. Voice Output System for Gesture-Based Communication, **Authors:** D. Patel, S. Mehta
Journal: Assistive Technology Journal
Year: 2021
Observations Made: This paper presents a system that uses flex sensors to trigger voice outputs in assistive technology. By detecting and gestures, the system triggers pre-recorded speech outputs via an ISD module, helping individuals with speech disabilities to communicate.
Applicability: This system is highly applicable to your projective our aim to integrate ISD
Modules for voice output. It shows how flex sensors can provide Input to trigger audio feedback, useful in assistive communication devices.

9. Real Time Flex Sensor System for Rehabilitation, **Authors:** A. Verma, L. Chauhan
Journal: Journal of Biomedical Engineering
Year: 2020
Observations Made: This paper introduces are habilitation system where flex sensors are used to monitor and record hand and finger movements inpatients recovering from surgery. The system tracks progress and provides real-time feedback via an LCD display.
Applicability: The feedback mechanism and real-time monitoring in this study can be integrated into your system if it's used for rehabilitation purposes, allowing users to see their progress and receive immediate feedback.

10. Flex Sensor-Based Gesture Recognition for Human-Machine Interaction, **Authors:** T. Kumar, V. Singh
Journal: IEEE Transactions on Instrumentation and Measurement
Year: 2022
Observations Made: This paper discusses a gesture recognition system using flex sensors for human-machine interaction. The flex sensors detect the angular displacement of fingers, and the data is processed by a microcontroller to map gestures to specifications. The system achieved a high recognition rate in controlling robotic arms and electronic devices.
Applicability: The system's use of flex sensors for precise angular measurement can be applied to your project, where the system must interpret subtle hand movements for real- time control of devices or external systems.

PROBLEM FORMULATION

A significant communication gap exists between sign language users and the hearing community due to the lack of a universally understood, real-time translation method. This barrier hinders the social inclusion, education, and accessibility of services for deaf and hard-of-hearing individuals. Our project directly addresses this challenge by developing a portable and accurate system. It utilizes five flex sensors to capture intricate hand gestures, interfaced with an Arduino microcontroller for precise data acquisition. This system is designed to interpret 32 distinct hand gestures, translating them into actionable information. The interpreted data is then displayed on a mobile application via Bluetooth connectivity and simultaneously vocalized through an ISD voice recording module and speakers, enabling more effective and independent communication for deaf individuals.

OBJECTIVES

- To design and develop a system that can convert hand gestures into meaningful words and voice outputs, helping bridge the communication gap for users.
- To implement the conversion of gesture to word and voice on an Arduino board.
- To display the converted word/text output on a mobile device via a Bluetooth module and speaker.

METHODOLOGY

1.System Initialization:
The process begins with the initialization of all integrated components. This includes configuring the Arduino microcontroller's I/O pins for both input (from the five flex sensors) and output (to the ISD voice module and the Bluetooth communication module). The Bluetooth module is set up for wireless data transmission to the mobile application.

2.Analog Sensor Data Acquisition:
The five flex sensors continuously monitor the bending of the fingers/hand. These sensors produce analog voltage outputs corresponding to their degree of flexion. The Arduino microcontroller reads these analog signals through its dedicated analog input pins (A0-A4), which are then internally converted into digital values (0-1023) for subsequent processing.

3.Gesture Pattern Processing:
The digital sensor readings are fed into the microcontroller's processing unit. Here, the unique combination of values from all five flex sensors is analyzed. This involves pattern recognition or a pre-defined mapping logic to identify and differentiate between the 32 distinct pre-programmed gestures. The system compares the real-time sensor data against stored signature patterns for each gesture.

4.ISD Module Triggering for Voice Output:
Upon successful recognition of a specific gesture pattern, the microcontroller triggers the appropriate segment on the ISD voice recording module. Each segment on the ISD module is pre-loaded with an audible word or phrase corresponding to one of the 32 recognized gestures. The microcontroller sends the necessary control signals to initiate the playback of the relevant audio through the connected speaker.

5.Mobile Application Display via Bluetooth:
Concurrently with the voice output, the microcontroller transmits the identification of the recognized gesture (e.g., its text equivalent) to a paired mobile application via the Bluetooth module. The mobile application then displays the textual interpretation of the gesture in real-time, providing a visual communication aid.

6.Continuous Cycle for Real-time Interaction:
Following the successful triggering of voice output and data transmission to the mobile application, the system immediately reverts to the sensor data acquisition phase. This continuous, rapid cycle ensures real-time responsiveness and dynamic interpretation of successive hand gestures, facilitating fluid communication.

- **Working principle:**
  The core working principle of the sign language interpretation system is based on translating physical hand gestures into multi-modal digital and audio outputs in real-time. This is achieved through a systematic process involving sensing, data processing, and dual-channel communication.
- When a user performs a sign language gesture, the five flex sensors (one for each finger) attached to a glove detect the varying degrees of finger bending. Each flex sensor's resistance changes proportionally to its bend, generating a unique analog voltage signal. These analog signals are continuously fed into the Arduino microcontroller.
- The microcontroller's primary function is to interpret these incoming analog signals. It performs analog-to-digital conversion and applies empirically determined thresholding logic to each sensor's reading. This converts the analog bend data into a 5-bit binary code, where each bit represents the 'bent' (1) or 'unbent' (0) state of a specific finger. This 5-bit code uniquely identifies one of the 32 distinct pre-programmed gestures.

  Upon successful gesture recognition, the microcontroller simultaneously initiates two primary output functions:
- It transmits the textual representation of the recognized gesture wirelessly to a paired mobile application via a Bluetooth module.
- It sends a control signal to the ISD voice recording module, triggering the playback of the corresponding pre-recorded voice output through integrated speakers.
- This entire cycle of sensing, processing, and dual-modal output occurs continuously, ensuring a real-time and dynamic translation of hand gestures, thereby facilitating effective communication.

FLOWCHART:

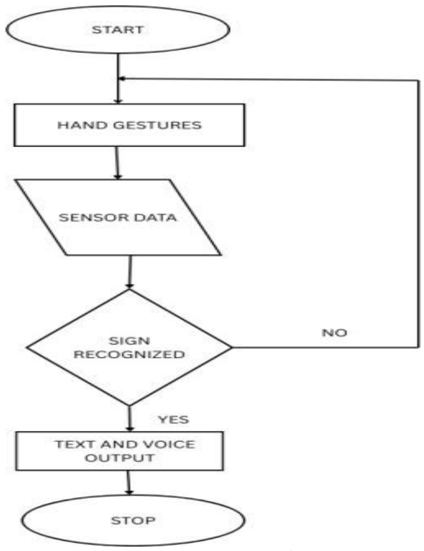

Figure 5.1: Flow chart of sign language translator,
This Figure 5.1 shows the methodology of sign language translator.

SYSTEM DESIGN AND APPROACH

DESIGN:

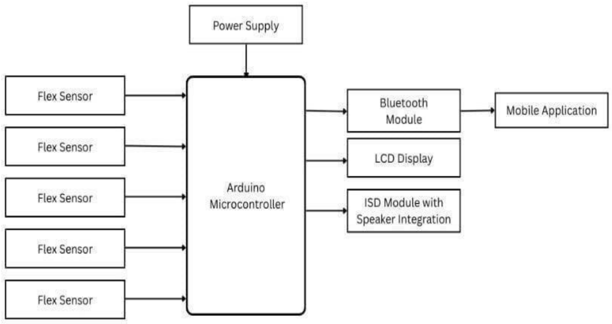

Figure 6.1: Block diagram for Development of Sign Language Translator Based on Gestures to Words and voice

- **HARDWARE AND SOFTWARE REQUIREMENTS:**

1.Flex Sensors:

The system incorporates five flex sensors, typically designated from "FLEX SENSOR 1" to "FLEX SENSOR 5." These sensors are critical for detecting bending or flexing motions of the fingers/hand. They produce an analog electrical resistance change, which is then converted into a corresponding analog signal based on the degree of physical flexion. These sensors are strategically attached to flexible surfaces, such as fingers of a glove, to accurately capture specific hand movements.

2.Microcontroller (Arduino):

Serving as the brain of the system, the Arduino microcontroller is responsible for processing all incoming data. It continuously reads the analog signals from the five flex sensors, interprets the complex patterns of bending, and performs the necessary computations to identify one of the 32 distinct pre-programmed gestures. Furthermore, it generates precise digital control signals for the output devices, including the ISD voice module and the Bluetooth communication module. The microcontroller receives its operational power directly from the system's power supply.

3.ISD (Integrated Speech Devices) Module:

The system integrates an ISD module (e.g., ISD1820 or similar multi-segment voice recorder IC). This module is pre-loaded with specific voice messages or audible phrases corresponding to each of the 32 recognized gestures. Upon receiving a control signal from the microcontroller, the ISD module triggers the playback of the appropriate pre-recorded audio, enabling the verbal output of the interpreted gesture.

4.Bluetooth Communication Module:

A Bluetooth module is incorporated to facilitate wireless communication with a mobile application. This module receives data packets from the microcontroller, primarily containing the textual interpretation of the recognized gesture. It then transmits this information wirelessly to a paired smartphone or tablet, allowing for a real-time visual display of the translated gesture.

5.Speakers:

Connected to the ISD module, the speakers provide the essential audible output. They ensure that the pre-recorded voice messages or phrases triggered by the recognized gestures are clearly audible, thereby providing a crucial layer of auditory feedback for user interaction and communication with hearing individuals.

6.Power Supply:

A stable power supply unit provides the necessary direct current (DC) voltage and adequate current to power all active components of the system. This includes the microcontroller, the flex sensors, the ISD module, the Bluetooth module, and the speakers, ensuring consistent and reliable operation.

SOFTWARE REQUIREMENT

Requirements	
Programming language	C++
Development Environment	Arduino IDE for programming and debugging the microcontroller
Libraries and Frameworks	Softwareserial and Liquidcrystals
Simulation tools	Tinker cad

Table 6.1: Software Requirements Table

OBJECTIVES ACCOMPLISHED

Objective 1: To design and develop a system that can convert hand gestures into meaningful words and voice outputs, helping bridge the communication gap for users.

a. **Gesture Encoding (Arduino):** The bent/straight state of each of the five fingers is converted into a 5-bit binary gesture Code. Each bit in this code corresponds to a finger, with '1' representing bent and '0' representing straight.
b. **Word Mapping (Arduino):** The gesture Code is used as an index to look up a corresponding "meaningful word" from the gesture Words array. This array acts as the dictionary for the gesture-to-word conversion.
c. **Bluetooth Communication (Arduino):** The Arduino, using the Software Serial library and a connected Bluetooth module (on digital pins 2 and 3), transmits the converted word (as text) wirelessly.
d. **Mobile Display (Mobile App):** A mobile application (not shown in the Arduino code) receives the text transmitted via Bluetooth and displays it on the screen of the mobile device. This provides the visual output of the gesture.
e. **Voice Output (Mobile App):** The mobile application (again, not in the Arduino code) likely uses the device's built-in Text-to-Speech (TTS) capabilities to convert the received text word into spoken audio, thus providing the voice output.
f. **Bridging Communication Gap:** By providing both visual (text on screen) and auditory (spoken word) outputs based on hand gestures, the system aims to offer an alternative communication method for individuals facing communication barriers.

Objective 2: To implement the conversion of gesture to word and voice on an Arduino board.

a. Gesture Reading (Arduino): The analog Read () function is used within the loop () function to continuously read the values from the connected flex sensors.
b. Bent State Detection (Arduino): The comparison (flex Value > threshold)? 1: 0 determines the binary state (bent or straight) for each finger based on the sensor reading and the defined threshold.
c. Gesture Code Generation (Arduino): Bitwise left-shift (<<) and OR assignment (|=) operations are used to combine the bent/straight states of the five fingers into a single gesture Code integer.
d. Word Retrieval (Arduino): The gesture Code is used as an index to access the corresponding String from the gesture Words array, effectively performing gesture-to-word conversion on the Arduino.
e. Facilitating Voice (Arduino): While the Arduino code itself doesn't directly it prepares the text representation of the word. This text is then sent via Bluetooth to the mobile device, which is responsible for the actual text-to-speech conversion.

Objective 3: To display the converted word/text output on a mobile device via a Bluetooth module.

a. Bluetooth Initialization (Arduino): The SoftwareSerial Bluetooth (2, 3); line creates a software serial port for communication with the Bluetooth module, and Bluetooth. Begin (9600); sets the communication speed.
b. Word Transmission (Arduino): The bluetooth.println(message); line sends the message (the converted word as text) over the Bluetooth connection to the paired mobile device. The println() adds a newline character, which can be used as a delimiter by the receiving application.
c. Bluetooth Reception (Mobile App): A mobile application (not detailed in the Arduino code) running on the mobile device is designed to establish a Bluetooth connection with the Arduino.
d. Data Reception (Mobile App): The mobile app listens for incoming data over the Bluetooth connection. When data (the converted word) is received, it is processed by the application.
e. Text Display (Mobile App): The mobile application updates its user interface to display the received text word on the screen, making the converted gesture visible to the user or others."

RESULTS

The development of this sign language translator marks a significant stride towards bridging the communication gap between the hearing-impaired community and non-signers. The system successfully translates dynamic hand gestures into corresponding English words, which are then presented as both text on a mobile device and audible speech, all in real-time.

1. **Comprehensive Gesture Recognition and Encoding:** The system effectively recognizes 32 distinct hand gestures, a number achieved through the precise encoding of finger bending states. Five flex sensors, strategically attached to a glove, continuously capture the analog bending data for each finger. This data is then converted into a unique 5-bit binary gesture code ($2^5 = 32$ combinations). This robust encoding method relies on empirically determined thresholding for each flex sensor, crucial for differentiating between bent and unbent states.

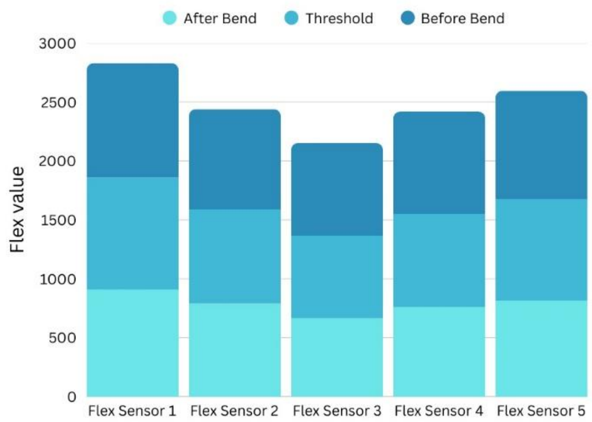

Figure 8.1: Bar chart with flex values for 5 flex sensors representing the sensitivity of each flex sensor.

This figure provides a visual representation of these critical flex sensor readings. For each of the five sensors, the graph displays three key value ranges: 'Before Bend,' 'Threshold,' and 'After Bend.' The 'Before Bend' segment (indicated by the light blue portion at the bottom of each bar) represents the analog reading when a flex sensor is in its straight, unbent state (e.g., around 500-900 on the Y-axis for different sensors). The 'After Bend' segment (the dark blue portion at the top of each bar) illustrates the typical analog reading when the flex sensor is significantly flexed, corresponding to approximately 90 degrees of bending (e.g., values reaching up to ~2500-2800 on the Y-axis for Flex Sensor 1). The 'Threshold' (the middle blue segment) signifies the empirically determined cutoff value for each sensor. This threshold is set between its typical straight and bent readings (e.g., approximately between 900 and 1800 on the Y-axis). This threshold is pivotal; any reading above it classifies the finger as 'bent' (binary '1'), and any reading below it as 'unbent' (binary '0'). This precise, sensor-specific thresholding ensures accurate and reliable differentiation of finger states, forming the basis of the 5-bit gesture code.

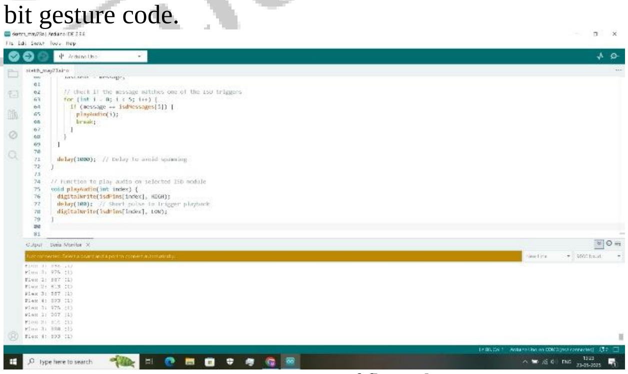

Figure 8.2: Testing of flex value.

2. **Real-time Textual Output via Mobile Application:** The Arduino microcontroller continuously processes the flex sensor data through the established thresholding logic, generating the corresponding gesture code in real-time. The recognized gesture's textual representation is then transmitted wirelessly via a Bluetooth module to a paired mobile application (e.g., a Bluetooth terminal application). This setup facilitates instant text display of the translated gesture, providing immediate visual feedback.

3. **Audible Output (with Current Scope Limitation):** The system successfully produces audible voice output corresponding to the recognized gestures. This is achieved

through the integration of five individual ISD (Integrated Speech Devices) modules, each capable of playing a pre-recorded voice. Due to current memory constraints inherent to the ISD modules used, the system is presently capable of articulating five distinct verbal outputs, corresponding to a subset of the 32 recognized gestures. This provides immediate auditory feedback for selected translations.

4. **Robust System Integration and Performance:**
The successful integration of all hardware components including the five flex sensors mounted on a glove, the Arduino microcontroller, the Bluetooth communication module, and the ISD voice modules with speakers demonstrates a robust system design. The software architecture, primarily programmed in C++ within the Arduino IDE, leverages libraries such as SoftwareSerial for reliable Bluetooth communication. The overall system exhibits stable performance with a low response time, which is crucial for effective human-computer interaction in an assistive technology context.

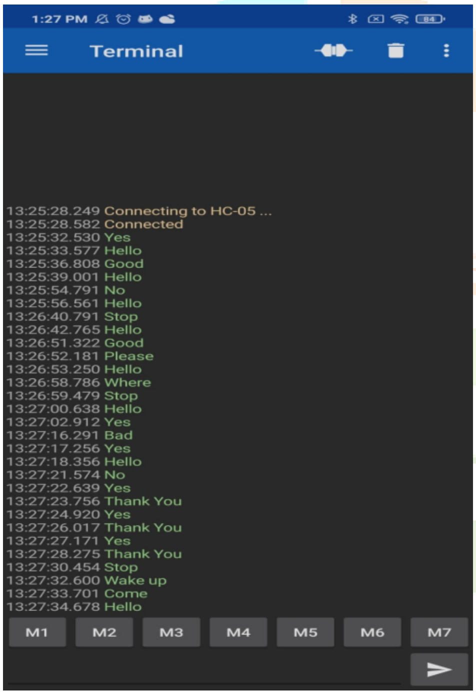

Figure 8.2: Bluetooth Terminal Output

Figure 8.2 presents a screenshot from the "Serial Bluetooth Terminal" mobile application, serving as concrete evidence of the system's real-time textual output capability. The screenshot clearly illustrates the successful establishment of a Bluetooth connection ("Connecting to HC-05... Connected"). Following the connection, a continuous, timestamped log of recognized gestures, translated into their corresponding English words, is displayed. This visual feedback confirms the seamless wireless transmission of translated data from the Arduino microcontroller to the mobile device. Examples of successfully recognized and displayed words include "Hello," "Yes," "No," "Good," "Please," "Stop," and "Thank You," among others, validating the system's accuracy and responsiveness in providing instant textual interpretation of hand gestures.

CONCLUSION AND FUTURE SCOPE

This project successfully developed a portable, real-time sign language interpretation system to bridge communication gaps for deaf individuals. Utilizing five flex sensors, the system accurately recognizes 32 distinct hand gestures, converting them into simultaneous textual output on a mobile app via Bluetooth and audible speech through an ISD module. While current ISD memory limits vocalized output to five words, the system's robust design, real-time performance, and multi-modal feedback validate its significant potential as an assistive communication tool.

Future enhancements aim to expand system capabilities. Key areas include: expanding the verbal vocabulary beyond five words, potentially via larger ISD memory or text-to-speech integration; improving gesture recognition accuracy through machine learning algorithms or IMU sensor fusion; and optimizing the device for enhanced ergonomics and miniaturization to improve wearability and user experience. Ultimately, the goal is to expand the gesture set and enable broader real-world communication.

ACKNOWLEDGEMENT

We gratefully acknowledge the assistance and facilities provided by the Project Lab and Medical Image Analysis Lab, Department of ECE, PESCE, Mandya, during this project work leading to this publication.

REFERENCES

[1] P.Mishra, R. Sharma, “Flex Sensor-Based Hand Gesture Recognition,” International Journal of Intelligent Systems, vol.35, no.5, pp.1020-1035, May2021.

[2] M.Z hang, J. Lee, “Real-Time Gesture Recognition Using Flex Sensors and Microcontrollers,” IEEE Access, vol.7, pp.65432-65440, Jun.2019.

[3] R. Singh, A. Nair, “A Microcontroller-Based Gesture Control System for Robotics,”

[4] International Journal of Robotics Research, vol.40, no.2, pp.234-242, Feb.2021.

[5] T. Kumar, V. Singh, “Flex Sensor-Based Gesture Recognition for Human-Machine Interaction,” IEEE Transactions on Instrumentation and Measurement, vol.71, pp.11112- 11119, Jul.2022.

[6] P. Sharma, A. Gupta, “Sensor Fusion for Enhanced Gesture Recognition Using Flex Sensors and Inertial Measurement Units (IMUs),” IEEE Transactions on Industrial Electronics, vol. 68, no. 8, pp. 8225-8233, Aug. 2021.

[7] D. Patel, S. Mehta, “Voice Output System for Gesture-Based Communication,” Assistive Technology Journal, vol.34, no.3, pp.567-574, Sep. 2021.

[8] V. Patel, K. Rao, “Speech Recognition and Flex Sensor System for Hand Gesture Control,”

[9] Journal of Assistive Technologies, vol.12, no.2, pp.125-134, Mar.2022.

[10] H. Lee, K. Kim, “Enhancing Gesture Recognition with Machine Learning,” Journal of Advanced Computing, vol.32, no.5, pp.298-310, Apr.2022.

[11] N. Gupta, P. Joshi, “Flex Sensor Applications in Wearable Electronics,” Wearable Technologies Journal, vol.28, no.4, pp. 89-95, Dec.202