Milestone Transactions on
MEDICAL TECHNOMETRICS
Vol. 02, Issue. 01, August 2024

RESEARCH ARTICLE OPEN ACCESS

iSignNet: A Novel Bidirectional Hybrid Sign Language Translation Framework Leveraging IoT and NLP for Intelligent Communication

Faizal Hussain . MD Omar Mukhtar . Kushal V . Vivek R Madabhavi . Chaithra M H . Syed Muzamil Basha

School of Computer Science and Engineering,
REVA University, Bengaluru, India

DOI: **10.5281/zenodo.14557864**
Received: 13 March 2024 / Revised: 18 May 2024 / Accepted: 21 July 2024

Abstract – Effective communication is essential, yet individuals with speech, hearing, or multiple impairments often face challenges, especially when others do not understand sign language. This paper introduces a system that translates sign language into text and speech, facilitating easier communication. The system includes five flex sensors attached to each finger to detect gestures, which are converted into binary signals processed by an Arduino Uno R3 microcontroller. These signals are displayed as text on a 16x2 LCD screen, supporting 32 gestures per phase, with multiple phases possible. Additionally, a DF Mini Player speaker module vocalizes the translated text for enhanced accessibility. The system also includes software that allows users to input text or speech. If a pre-recorded Blender 3D animation exists for a word, it is shown; if not, the system breaks the word down into characters and displays animations for each one. For sentences, it tokenizes and removes stop-words, showing animations for each word. If no animation exists, it displays each character individually. This hardware-software integration offers a cost-effective solution for those with speech and hearing impairments, using flex sensors to reduce costs and enhance precision, effectively bridging communication gaps.

Index Terms –Sign to text, Smart Gloves, Arduino UNO R3, Flex Sensor, Speech Impaired, LCD 16x2 Display, DF Mini Player, Blender, Text to sign, Speech Recognition.

I. INTRODUCTION

Communication is fundamental to human connection, driving understanding, collaboration, and progress. Yet, for individuals with speech, hearing, or multiple disabilities, effective communication can

MILESTONE
RESEARCH.IN
OPEN ACCESS
Electronic ISSN: **2584-072X**

be a significant challenge. Sign language serves as a vital tool for these individuals, offering a means of expression and connection in a predominantly verbal world. However, the widespread use and understanding of sign language remain limited, creating communication barriers for those who rely on it. This paper presents a system designed to overcome these challenges by translating sign language into accessible text and speech. The system integrates five flex sensors attached to the user's fingers to detect hand gestures, which are then converted into binary signals. These signals are processed by an Arduino Uno R3 microcontroller, and the translated gestures are displayed as text on a 16x2 LCD screen. To enhance accessibility further, a DF Mini Player speaker module is incorporated to read aloud the translated text, making it more inclusive for users who may benefit from auditory feedback. This setup enables sign language users to communicate more easily with those unfamiliar with their gestures.

In addition to the hardware component, the system includes a software solution that enhances flexibility for the user. The user can provide input through either text or speech, depending on their preference. For text-based input, the system checks for pre-recorded 3D animations of words using Blender. If an animation exists, it is displayed; if not, the word is broken down into individual characters, and corresponding animations are shown for each character. When the input is a sentence, the system uses the Natural Language Toolkit (NLTK) to tokenize the sentence, remove stop-words, and check if animations exist for each word. If an animation is missing, the system animates each character of the word. By combining both hardware and software, this system offers an efficient, cost-effective solution to improve communication accessibility for people with speech and hearing impairments. It creates an opportunity for meaningful communication between sign language users and those who are not familiar with sign language, fostering greater inclusivity and breaking down communication barriers.

II. LITERATURE SURVEY

Sparsha et al. [1] developed a model comprising flex sensors for each finger, a DF Mini Player, a speaker, and an Arduino Nano as the main microcontroller. In this proposed model, the resistance values from the flex sensors are transmitted to the Arduino Nano. The analog values captured are converted to digital values, which are then compared with previously stored data in the microcontroller's database. When fingers are bent, resulting in varying resistance, the digital values are mapped with existing values in the database to recognize various hand gestures. The speech output is obtained with the assistance of the DF Mini Player. Priyadharshini et al. [2] utilized an Arduino Uno as a microcontroller, an HC-05 Bluetooth module, an LCD Display, and four flex sensors mounted on gloves, excluding the thumb finger. Resistance values are altered based on the bending of fingers, and these values are converted using Analog-to-Digital Conversions. Signals are mapped to predefined codes, and if there is a match, the output is displayed on the LCD screen. Additionally, using the HC-05 module, an Android phone is connected to the Arduino, and with the aid of Text-to-Speech Conversion, audio output is obtained.

Md Imran Hossain et al. [3] in their research have developed a model comprising five flex sensors, an Olimexino (a clone of Arduino Leonardo) utilized as the microcontroller, an LM386 Audio Amplifier Module, an SD Card, and an SD Card Module. Initially, the microcontroller remains in a resting state. Upon detection of movement, it reads data from the flex sensors. If the data aligns with the programmed logic, the model is capable of providing voice output corresponding to the detected sign. A wearable model developed by Marvin S. Verdadero and Jennifer C. Dela Cruz [4] utilized a gyroscope, an accelerometer

(MPU6050), flex sensors, an LCD display, a speaker, and an Arduino Nano with an inbuilt Atmega328 microcontroller. Five flex sensors were mounted on the top of a glove, with each having one sensor. Flex sensor calibration was performed to determine the minimum and maximum values of each flex sensor. Different letters were assigned to different combinations of resistance values. In cases of overlapping resistance values for different letters, values from the gyroscope and accelerometer along the x, y, and z coordinates were used to predict the correct letter. Based on the resistance, gyroscope, and accelerometer readings, letters were displayed on the LCD display, accompanied by voice output.

Nitin Thoppey Muralidharan et al. [5] proposed a model implemented in TinkerCad, consisting of flex sensors, an LCD display, and an Arduino Uno. In this method, five flex sensors were used, with each considered as one bit, resulting in a total of 32 combinations in Mode 1 and Mode 2. Mode 1 was activated by the combination of 01000, containing a set of messages. Mode 2 was activated with the combination of 01100, also containing 32 combinations, including English alphabets. Customized messages with combinations of letters were generated using the alphabets in Mode 2. Radzi Ambar et al. [6] developed a device that employs five flex sensors from Spectra Symbol for angle displacement measurement, crucial in recognizing sign language gestures. These sensors operate by altering resistance when bent, enabling the calculation of finger bending angles. Additionally, a GY61 accelerometer, integrated into a white cotton glove, measures hand acceleration along the x-, y-, and z-axes, facilitating the determination of hand postures. Auditory feedback is provided using a 0.5 W speaker connected to an LM286 amplifier circuit, allowing for the playback of audio corresponding to recognized sign language gestures. The speaker's low power consumption and optimal frequency range make it suitable for this purpose. A potentiometer within the amplifier circuit enables volume adjustment for user convenience. To store audio speech files, a micro secure digital (SD) card reader module is utilized, employing the Serial Peripheral Interface (SPI) interface for communication with the Arduino Mega 2560 microcontroller. The device stores audio files in WAV format due to Arduino limitations, ensuring compatibility. Additionally, a mechanism is implemented to handle connection failures between the SD card reader and the microcontroller, ensuring robust operation.

S Yarisha Heera et al. [7] proposed a device that utilizes data from a 3-way accelerometer and gyroscope, specifically the MPU6050, to collect motion data from the hands. With hand movement along a particular axis, readings from each axis—X, Y, and Z—change, providing positive or negative values associated with each axis. By analyzing these values, the hand's position can be determined. Additionally, flex sensors integrated into the glove detect hand gestures, with resistance across each sensor changing based on the gesture performed. The microcontroller processes the data from both the MPU6050 and flex sensors to determine the overall hand position and gesture. This assembly, designed to be portable, is mounted on both hands. The gloves are interconnected via Bluetooth HC-05, with the left-hand glove transmitting data to the right-hand glove. The right-hand glove combines this data with its own sensor readings before transmitting it to an Android phone connected via another Bluetooth module. In the Android application, a database of pre-defined alphabets and words is stored. Each combinational value received from the gloves is matched against this database. Upon a successful match, the Android application converts the corresponding text to speech, providing real-time auditory feedback. Anirbit Sengupta et al. [8] have developed a model employing flex sensors, an accelerometer, a Bluebee Bluetooth module, and an ATMEGA 32 microcontroller. Additionally, a smartphone application was developed using MIT App Inventor software, comprising four buttons: Search Bluetooth, Connect, Play, and Clear. Finger movements were mapped to predefined data, and upon a match, the data was transmitted to the smartphone via the Bluebee module. Textual data was then converted to voice output.

Bonthu et al. [10] created a web-based system to support Indian Sign Language (ISL) communication through a 3D animated character. The platform, built with HTML, CSS, and JavaScript, accepts text and voice input, with the Web Speech API enabling speech recognition. NLTK processes the text to focus on key information, while the animated character—designed in Blender—displays ISL gestures, overcoming challenges like complex, two-handed signs. A hybrid cloud setup efficiently stores gestures, and the HOG algorithm refines animations, providing a user-friendly ISL tool accessible on demand**.** Naqvi et al. [11] created a user-friendly sign language communication system with Tkinter, allowing seamless interaction through text-to-sign and sign-to-text conversion. By using MediaPipe, the system efficiently tracks hand movements and identifies key points for accurate gesture recognition. To improve recognition, they integrated LSTM networks, which analyze the motion of the gestures over time. The system also allows real-time interaction with virtual assistants, enabling users to control applications using hand gestures, making it a more accessible solution for those with hearing impairments.

Gunasagari et al**.** [12] introduced a system designed to facilitate communication for individuals with hearing and speech impairments using Indian Sign Language (ISL) gestures. The system is divided into three main components: the Data Glove, the Processing unit, and the Display/Conversion unit. The Data Glove captures the movements of the hand and finger bending through sensors, including the MPU6050 and flex sensors. The data gathered from these sensors is then processed by a microprocessor, which translates the gestures into a recognizable format and sends the information to a connected Android app via Bluetooth. On the app, the meaning of the gesture is displayed, and the system also incorporates speech-to-text and text-to-speech features for bidirectional communication. With a recognition accuracy of 80% and low latency, this system enables effective real-time communication for users. Krunal Saija et al. [13] developed a comprehensive system for converting spoken English into Indian Sign Language (ISL) gloss, focusing on improving accessibility for the deaf and mute community. The process begins with voice-to-text conversion, where speech is transcribed into written text using the Hidden Markov Model (HMM), a widely used technique in speech recognition for its ability to handle the temporal nature of speech signals . Following this, text pre-processing is applied to enhance the quality of the text for translation. This involves removing stop words, tokenizing the text into individual words, and performing stemming to convert words to their root forms. The pre-processed text is then subjected to text-to-ISL gloss translation. In this phase, the system utilizes WordNet, a lexical database, to find appropriate replacements for words not present in the ISL dictionary, relying on LCH similarity to maintain meaning and ensure accurate translation . The final output is displayed as ISL gloss in capital letters, ensuring clarity in communication. This method not only improves real-time translation efficiency but also enhances the accuracy of the translation process, addressing the challenges posed by speech variability and ensuring effective communication in Indian Sign Language.

Pradeep Sudhakaran et al. [14] created a system that translates spoken English into Indian Sign Language (ISL) through a multi-step process. Initially, the system captures speech using a microphone and converts it into text via speech recognition. This text is then broken down into individual words and phrases through tokenization. Next, a probabilistic context-free grammar (PCFG) parser analyzes the grammatical structure of the text, generating a syntactic tree that is adapted to fit the rules of ISL. The text undergoes further processing with techniques like stemming and lemmatization to normalize word variations. Finally, the system generates animated gestures that represent the translated ISL, providing an intuitive interface for users. The approach combines natural language processing, rule-based translation, and animation to bridge the gap between spoken English and ISL.

III. METHODOLOGY

Our proposed model delineates the systematic approach to developing a portable real-time translation device. This device seamlessly integrates advanced technologies, including Arduino Uno R3, flex sensors, an LCD 16x2 display with an I2C module, and a DF Mini Player with a speaker module, into a glove form factor. By harnessing the capabilities of these components, the device enables users to translate finger gestures into text and audio output in real time, facilitating communication for individuals with speech impairments or language barriers.

A. Hardware Implementation

Sensor Calibration

- **Determination of Threshold values:** The calibration process commences with the establishment of threshold values for the flex sensors integrated into the glove. Through meticulous experimentation, threshold values are meticulously set to accurately discern between bent and unbent states of the fingers.

  The predetermined threshold values, tailored for optimal performance, are as follows:
  - Sensor 1: 800
  - Sensor 2: 800
  - Sensor 3: 800
  - Sensor 4: 920
  - Sensor 5: 1003

Gesture Recognition

- **Monitoring sensor readings:** Upon calibration, the Arduino Uno continuously monitors the resistance values of the flex sensors embedded within the glove. Through real-time analysis, the microcontroller discerns the bending or straightening of each finger based on the sensor readings.
- **Binary encoding of gesture states:** A binary encoding scheme is employed to represent the states of finger gestures detected by the flex sensors.
- When a sensor's resistance exceeds its respective threshold value, a binary value of 1 signifies a bent state, while 0 denotes an unbent state.
- The combination of binary values from all five flex sensors generates unique 32-bit representations, characterizing distinct finger gestures.

Text And Audio Mapping

- **Assigning the language alphabets and emergency messages:** Each 32-bit combination corresponds to specific alphabets or emergency messages, facilitating seamless translation of finger gestures into meaningful output. An intricate mapping system associates each combination with its corresponding textual and auditory feedback, enhancing the device's usability.
- **Audio track preparation:** Pre-recorded audio tracks, encompassing alphabets and emergency messages, are meticulously prepared and stored on the SD card of the DF Mini Player. These audio tracks are intricately mapped to their respective messages within the lookup table, ensuring synchronized audio-visual feedback.

User Interface

- **Display of translated text:** The translated text is prominently displayed on the LCD 16x2 display, seamlessly integrated into the glove's design. The visual feedback provided by the display enhances user interaction, enabling efficient communication through intuitive gesture translation.
- **Playback of audio feedback:** Simultaneously, the DF Mini Player initiates the playback of corresponding audio tracks through its speaker module, augmenting the user experience. This dual-channel feedback mechanism ensures comprehensive communication accessibility, catering to diverse user preferences and needs.

Compact And Detachable Design

- **Component integration:** All components, including the Arduino Uno, flex sensors, LCD display, and DF Mini Player, are ingeniously integrated into the glove, optimizing portability and convenience. The flex sensors are strategically positioned within the glove's fabric, ensuring seamless gesture detection without compromising comfort or mobility.
- **Modular connectivity:** The device's modular design facilitates effortless connectivity and disconnection of components, empowering users to customize and maintain the device with unparalleled ease. This innovative design approach underscores the device's adaptability and user-centric functionality, establishing a new paradigm in portable communication aids.

B. Software Implementation

The software component complements the hardware by providing a flexible interface for users to input text or speech, process it, and display or animate it in a way that enhances accessibility. This part of the system is built using a blend of front-end technologies, text preprocessing tools, and a 3D animation platform to visualize words or gestures.

Front-End Interface Development

HTML, CSS, and JavaScript: The front-end of the system is developed using HTML, CSS, and JavaScript to create a user-friendly interface. This interface allows users to enter text or initiate speech recognition for communication. HTML structures the interface, CSS ensures a visually appealing layout, and JavaScript enables interactive elements for seamless user engagement.

Speech Recognition

JavaScript web speech API: To offer an additional means of input, the system integrates the JavaScript Web Speech API. This API enables users to speak words or phrases, which are then converted into text. The speech recognition feature makes the system more accessible for users who prefer voice input over typing, ensuring that spoken words are effectively captured and processed.

Text Preprocessing

Natural Language Toolkit (NLTK): Once the text is either entered manually or converted from speech, it undergoes preprocessing. Using the Python-based Natural Language Toolkit (NLTK), the system tokenizes the text, breaking down sentences into individual words for easier processing. Additionally, NLTK is employed to remove stop words—commonly used words that add little meaning, such as "and," "is," or "the." This step allows the system to focus on meaningful keywords, making the translation process more efficient and accurate.

Word and Character-Based Animation Selection

- **3D Animation with Blender**: For visual representation, the system relies on 3D animations created with Blender. When a word is processed, the system checks if there is a pre-existing animation for that specific word. If an animation is available, it is displayed to provide users with a visual representation of the word, simulating sign language gestures.
- **Character Animation for Unavailable Words**: In cases where a word does not have a predefined animation, the system breaks it down further into individual characters. Each character is then matched with a corresponding animation in Blender, effectively "spelling out" the word letter by letter. This process ensures that users still receive visual feedback, even for less common or new words that lack dedicated animations.

Display and Interaction

Animation Display for Enhanced Understanding: The final output—whether a full word or character-based animation—is displayed on the screen, allowing users to visualize the translation. This approach bridges the gap between text and sign language, giving users an accessible, interactive experience.

C. Hardware Components

- **Arduino Uno R3:** The Arduino Uno R3 serves as the central microcontroller of the real-time translation device. It processes sensor data, controls output devices, and executes the translation algorithm. With its versatility and ease of programming, the Arduino Uno facilitates seamless integration of various hardware components.
- **Flex Sensors:** Five flex sensors are strategically integrated into the glove, positioned to detect finger gestures accurately. These sensors measure the degree of bending in each finger by detecting changes in resistance. Flex sensors play a pivotal role in gesture recognition, enabling the translation of finger movements into digital signals.

- **LCD 16x2 Display with I2C Module:** The LCD 16x2 display provides visual feedback by showcasing translated text and system status. With the integrated I2C module, the display interface is simplified, requiring fewer pins for connection to the Arduino Uno. Its compact size and high contrast make it suitable for embedding within the glove, ensuring user accessibility without compromising mobility.

- **DF Mini Player with Speaker Module:** The DF Mini Player, equipped with a speaker module, facilitates audio playback of pre-recorded messages. It stores audio files on an SD card, enabling seamless access to a library of alphabet pronunciations and emergency messages. The compact size and low power consumption make the DF Mini Player an ideal choice for integrating audio capabilities into the real-time translation device.

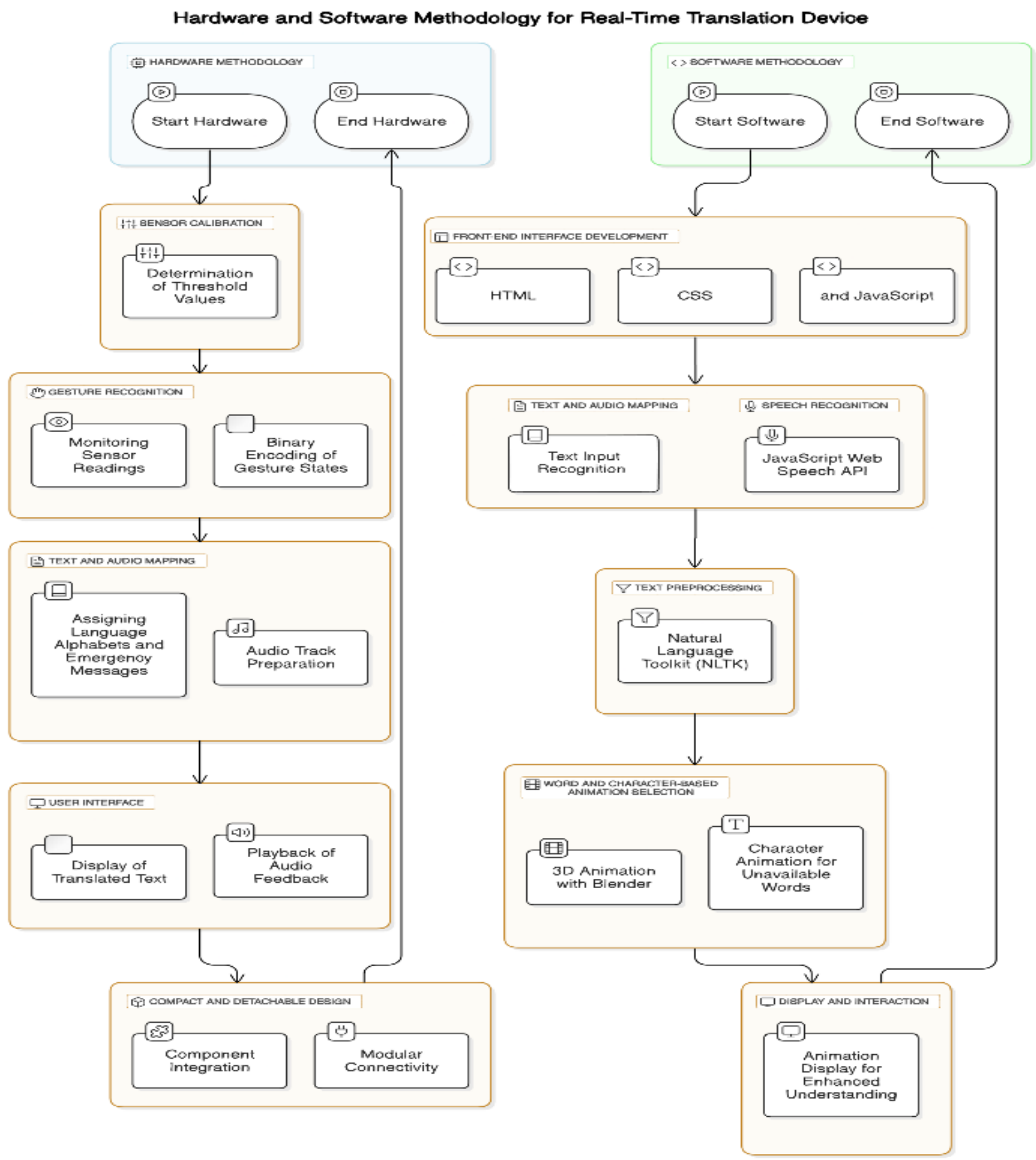

Fig. 1. Unified Methodology Flow Diagram

- **Glove:** The glove serves as the physical interface of the real-time translation device, housing all hardware components. Designed for comfort and flexibility, the glove allows natural hand movements while ensuring accurate gesture detection. Its compact and detachable design enhances portability and usability, making the device suitable for various communication scenarios.

IV. WORK FLOW

The real-time translation device is a comprehensive solution designed to facilitate communication between users of sign language and those who rely on spoken or written text. By integrating hardware and software workflows, the system enables bidirectional translation, with each component serving a distinct yet complementary purpose. Together, they address the critical need for bridging communication gaps in an inclusive and efficient manner. The hardware workflow focuses on the translation of sign language gestures into text or speech. The process begins with a glove embedded with flex sensors that detect finger movements by measuring changes in resistance as the fingers bend or straighten. These movements are converted into binary combinations, each representing specific gestures. These combinations are then mapped to corresponding letters, words, or emergency messages. The translated text is displayed in real-time on an LCD screen, providing immediate visual feedback to the user. Additionally, a speaker module plays pre-recorded audio associated with each gesture, delivering auditory feedback. The glove is designed to be lightweight and portable, ensuring ease of use in various real-world applications.

TABLE 1: Bit Value and Word Mapping Table

P1 BIT VALUE	P1 MESSAGE ENGLISH	P2 BIT VALUE	P2 MESSAGE FRENCH
00000	(HAND AT REST)	00000	(HAND AT REST)
00001	I AM	00001	JE SUIS
00010	HOW	00010	COMMENT
00011	ARE	00011	SONT
00100	GOOD	00100	BIEN
00101	YOU	00101	TOI
00110	HAPPY	00110	HEUREUX
00111	BEAUTIFUL	00111	BELLE
01000	HI	01000	SALUT
01001	HELP	01001	AIDE
01010	THANK YOU	01010	MERCI
01011	NEED	01011	BESOIN
01100	FOOD	01100	NOURRITURE
01101	MEDICINE	01101	MEDICINA
01110	WATER	01110	EAU
01111	MEETING	01111	REUNION
10000	GOOD MORNING	10000	BONJOUR
10001	USE	10001	UTILISER
10010	WASHROOM	10010	TOILETTE
10011	PLEASE	10011	S'IL TE PLAIT
10100	EXCUSE ME	10100	EXCUSEZ-MOI
10101	DAY	10101	JOUR
10110	HAVE	10110	AVOIR
10111	NICE	10111	BONNE
11000	GOOD EVENING	11000	BONNE SOIREE
11001	TIRED	11001	FAIGUEE
11010	SAD	11010	TRISTE
11011	SORRY	11011	DESOLEE
11100	SICK	11100	MALADE

P1 BIT VALUE	P1 MESSAGE ENGLISH	P2 BIT VALUE	P2 MESSAGE FRENCH
11101	WELCOME	11101	BIENVENUE
11110	BYE	11110	AU REVOIR
11111	MOVING STATE 1	11111	MOVING STATE 0

Complementing the hardware workflow, the software workflow addresses the reverse process, enabling the conversion of spoken or written language into sign language. This workflow is supported by a web-based interface, developed using HTML, CSS, and JavaScript, which serves as the primary medium for interaction. Users can input spoken language via speech recognition technology powered by the Web Speech API or directly type text into the interface. The system processes the input in real-time, converting speech to text with high accuracy. To enhance the clarity and usability of the output, the text is refined using natural language processing techniques implemented with the Natural Language Toolkit (NLTK). Once processed, the system translates the text into sign language through a visual representation. This is achieved using a 3D animated character created with Blender, which demonstrates the corresponding gestures. The animated character provides a dynamic and intuitive visual aid, helping users understand and learn the signs while ensuring accurate representation of the input.

Together, these workflows form a cohesive, bidirectional communication system. The hardware enables individuals to express themselves through gestures, translating them into spoken or written language. In contrast, the software provides the capability to convert spoken or written input into sign language, facilitating communication with individuals who rely on gestures. This dual functionality is depicted in the unified workflow diagram (Fig. 1), highlighting the integration of both components into a single, user-centric system. By addressing the requirements of both sign language users and non-sign language users, the device demonstrates its potential as a transformative tool for inclusive communication.

V. RESULTS

A. Hardware Implementation

In this section, the outcomes pertaining to the proposed model are discussed, emphasizing its functionality in translating gestures into meaningful outputs. Upon powering up the Arduino UNO, the system validates its connection to the DF Mini Player, with a confirmation message displayed on the serial monitor (Fig. 2). This initialization step ensures that all hardware components are prepared for gesture detection. The default state of the hand, where all fingers are fully extended, corresponds to the bit combination '00000.' This state indicates that the hand is at rest and no gesture is being made (Fig. 3). Additionally, this neutral position acts as a reset command, clearing the display of the previously shown output and preparing the system for the detection of subsequent gestures. Each gesture is mapped to a specific word, as detailed in TABLE 1. For instance, the gesture for the word "HI" corresponds to the bit combination '01000,' which represents the bent state of sensor 2, associated with the index finger (Fig. 4). When this combination is detected, the system displays the word "HI" on the screen and simultaneously plays the corresponding audio track (e.g., 008.mp3) through the speaker.

Milestone Transactions on
MEDICAL TECHNOMETRICS
Vol. 02, Issue. 01, August 2024

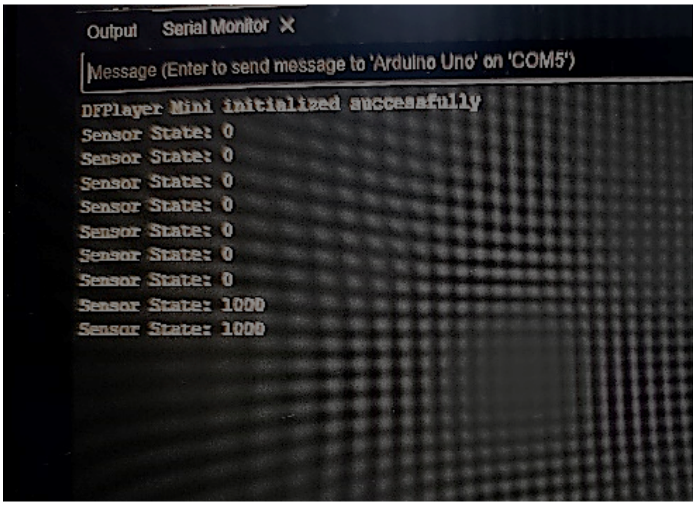

Fig. 2: DF mini player initialization status on serial monitor

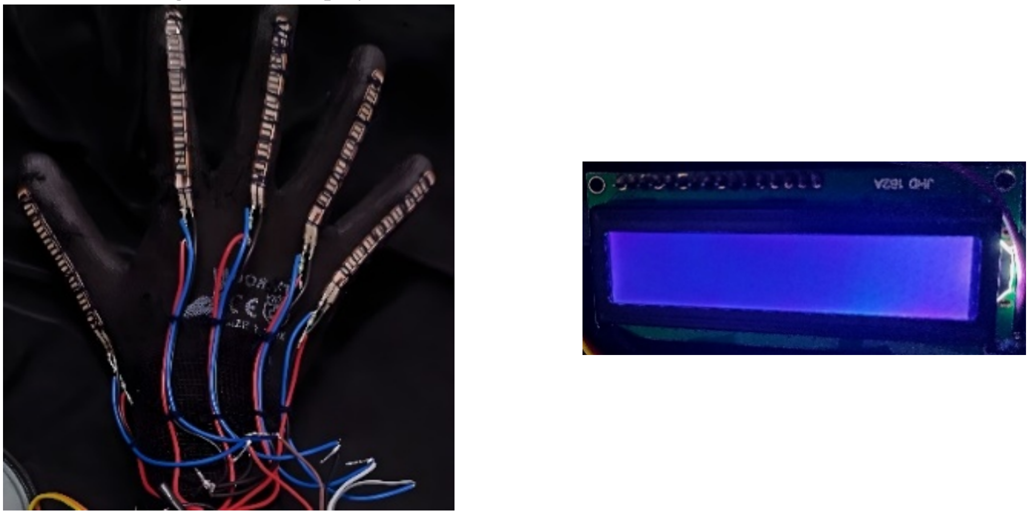

Fig. 3: Sensors in '00000' State (i.e. Hand at rest) no action

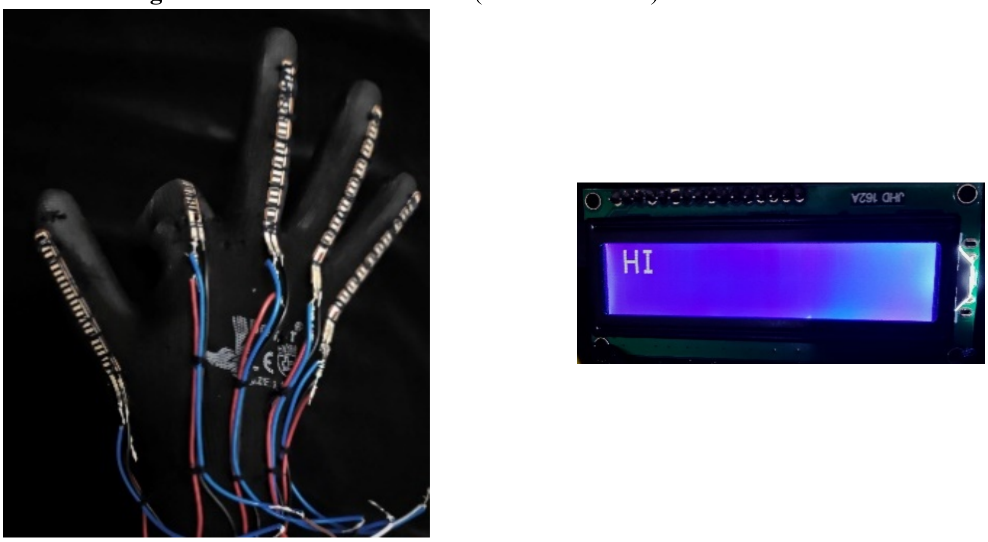

Fig. 4: Sensor state '01000' mapping the letter 'HI'

Milestone Transactions on
MEDICAL TECHNOMETRICS
Vol. 02, Issue. 01, August 2024

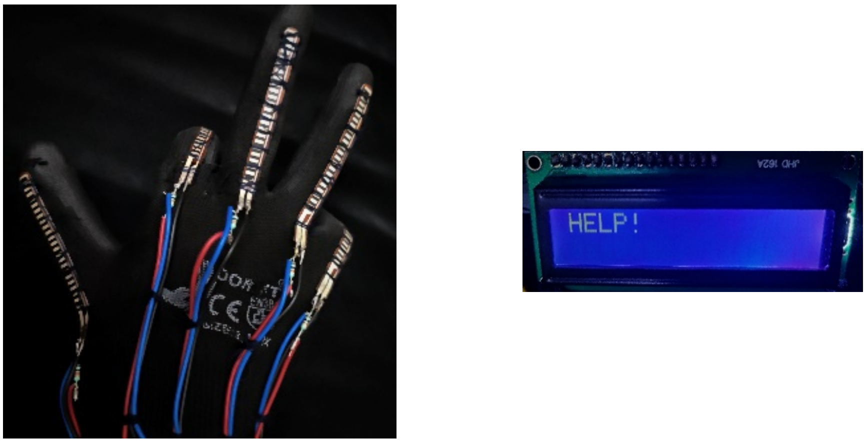

Fig.5: Sensor state '01001' indicating message 'HELP!'

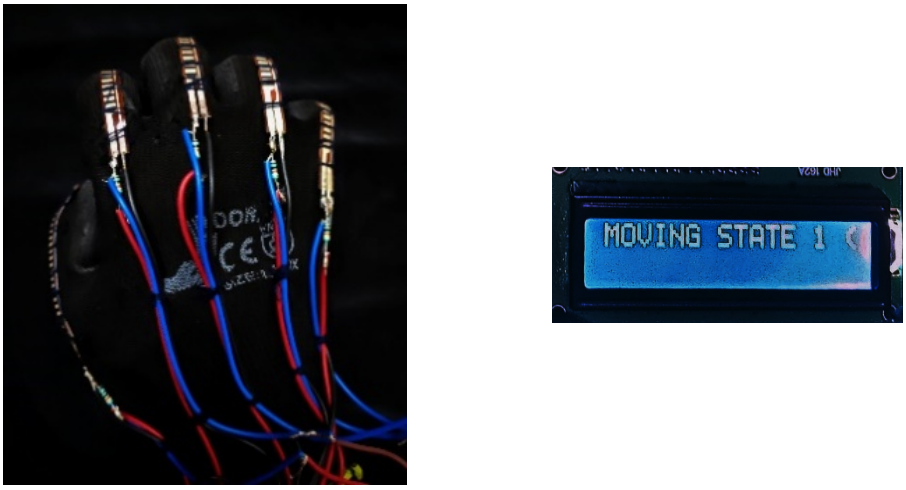

Fig.6: Sensor state '11111' acting as a system-level function

Similarly, the word "HELP" is associated with the bit combination '01001,' which involves bending both the index finger (sensor 2) and the pinky finger (sensor 5) (Fig. 5). Upon recognizing this gesture, the system displays the word "HELP" while playing its designated audio track (e.g., 009.mp3), ensuring immediate and clear communication. A more complex functionality is implemented for the bit combination '11111,' where all fingers are bent and all sensors are activated. This specific combination serves as a command to transition the system into Phase 1, which facilitates translation in the French language (Fig. 6). Rather than displaying a word, this gesture acts as a trigger for a system-level function, with an accompanying audiorompt confirming the activation of this new phase.

The system also incorporates the capability to form complete sentences by appending multiple words using successive gestures. A built-in delay mechanism ensures that the previously displayed word remains on the screen until the '00000' state is encountered to reset the display. For example, after the word "HI" is displayed, the system retains this output while the next gesture is detected. If the user then performs the gesture for "HELP," the new word is appended to the previous one, forming the phrase "HI HELP" on the display. This process continues, allowing users to construct sentences by combining words through consecutive gestures without resetting the display. Once the user extends all fingers to the resting state ('00000'), the display clears, and the system is ready for new input. This combination of single-word outputs, sentence formation through appending gestures, and advanced command functionalities highlights the versatility and practicality of the model. The integration of visual and auditory feedback ensures clear and effective communication, making the system an invaluable tool for real-time interaction.

B. Software Implementation

In this section, the outcomes of the software workflow are presented, highlighting its functionality in translating speech or text into corresponding sign language gestures. The software is designed to complement the hardware system, creating a bidirectional communication framework. Upon initializing the application, the web interface is loaded, providing a user-friendly platform for interaction. This interface, developed using HTML, CSS, and JavaScript, ensures accessibility and ease of use. The process begins when the user inputs a message, either through speech or text. If speech input is chosen, the Web Speech API processes the spoken words in real-time, converting them into text. The text is then displayed on the interface, offering immediate feedback to the user. In cases of typed input, the user simply enters the desired text into the designated field. This input is processed without delay, ensuring a smooth transition to the next step.

Once the input is received, the system employs the Natural Language Toolkit (NLTK) to process and refine the text. This step ensures that the input is clean, grammatically correct, and contextually relevant, removing unnecessary elements that could disrupt the translation. The processed text is then broken into smaller units, such as individual words or phrases, which serve as the basis for the subsequent translation into sign language. The translated gestures are visually represented through a 3D animated character developed using Blender. This character performs the corresponding sign language gestures for each word or phrase in sequence, providing users with clear and accurate visual feedback. For example, if the user inputs the phrase "HELLO, HOW ARE YOU?" the character sequentially performs the gestures for each word, ensuring the message is conveyed effectively. The animations are synchronized with the input to maintain a fluid and natural representation of the gestures.

The system supports the concatenation of multiple phrases, allowing users to input longer sentences or conversations. The interface dynamically displays each word or phrase as it is processed and translated, ensuring clarity and coherence in communication. The 3D animations play continuously until the entire message is conveyed, providing a comprehensive representation of the input. To enhance user engagement, the software incorporates additional features such as gesture playback and learning aids. Users can replay gestures to study and practice sign language, making the system a valuable tool for both communication and education. The modular design also allows for future expansions, such as integrating additional languages or customizing gestures for specific phrases. The software workflow demonstrates its effectiveness in providing real-time translation of speech or text into sign language, bridging communication gaps and promoting inclusivity. By combining advanced speech recognition, natural language processing, and 3D animation, the system delivers a robust and user-centric solution for translating spoken or written language into visual gestures.

VI. CONCLUSION

In conclusion, our research presents a significant advancement in the domain of assistive technology, particularly in the development of communication aids for individuals with disabilities. Through the comparative analysis with the model proposed by John.K et al. [9], it becomes evident that their model offers limited functionality and complexity compared to our innovative solution. Our model, incorporating five flex sensors, provides a broader range of signs for communication, whereas the model proposed by [9] is confined to displaying only eight predefined messages using two flex sensors. This expanded sign repertoire in our model enhances the communicative capacity for individuals with disabilities. Moreover, our model excels in simplicity, mobility, and cost-effectiveness. Unlike the model proposed by [9], which lacks mobility and is relatively complex due to its limited number of sensors and components, our model features a compact design and strategic placement of sensors on the glove. This enhances user flexibility and ease of communication. Furthermore, our model achieves similar functionality at a significantly lower cost, with expenses not exceeding 2800 rupees, compared to the 6000 rupees incurred by the model proposed by Aparna Sreekumar et al. This substantial cost reduction makes our model more accessible and viable for a wider range of users, particularly those from marginalized communities.

On the software side, our model leverages a user-friendly interface built with HTML, CSS, and JavaScript. This web interface ensures smooth and efficient communication by converting finger gestures into text. The use of the Web Speech API allows for speech recognition, enabling hands-free operation and enhancing accessibility for users. Moreover, integrating the Natural Language Toolkit (NLTK) ensures that the translated text is meaningful, providing more accurate and contextually appropriate translations. To further improve the user experience, a 3D character animation created using Blender visualizes the gestures, making the system more engaging and easier to understand. The software elements, combined with the intuitive interface, ensure that users can interact with the system effortlessly, promoting ease of use and accessibility., Our research underscores the importance of continual innovation and improvement in assistive technology to meet the diverse needs of individuals with disabilities. By combining advanced sensor technology with effective and intuitive software, our model serves as a significant step toward enhancing communication for individuals with disabilities and contributing to a more inclusive society.

REFERENCES

1. Haas, J. D., & Brownlie IV, T. (2001). Iron deficiency and reduced work capacity: a critical review of the research to determine a causal relationship. *The Journal of nutrition*, *131*(2), 676S-690S.
2. Kozuki, N., Lee, A. C., & Katz, J. (2012). Child Health Epidemiology Reference G. Moderate to severe, but not mild, maternal anemia is associated with increased risk of small-for-gestational-age outcomes. *J Nutr*, *142*(2), 358-62.
3. Steer, P. J. (2000). Maternal hemoglobin concentration and birth weight. *The American journal of clinical nutrition*, *71*(5), 1285S-1287S.
4. Amin, M. N., & Habib, M. A. (2015). Comparison of different classification techniques using WEKA for hematological data. *American Journal of Engineering Research*, *4*(3), 55-61.
5. Dogan, S., & Turkoglu, I. (2008). Iron-deficiency anemia detection from hematology parameters by using decision trees. *International Journal of Science & Technology*, *3*(1), 85-92.
6. Abdullah, M., & Al-Asmari, S. (2016). Anemia types prediction based on data mining classification algorithms. In *Communication, management and information technology* (pp. 629-636). CRC Press.
7. Veluchamy, M., Perumal, K., & Ponuchamy, T. (2012). Feature extraction and classification of blood cells using artificial neural network. *American journal of applied sciences*, *9*(5), 615.
8. Bashir, S., Qamar, U., Khan, F. H., & Javed, M. Y. (2014, December). An efficient rule-based classification of Diabetes using ID3, C4. 5, & CART ensembles. In *2014 12th International Conference on Frontiers of Information Technology* (pp. 226-231). IEEE.
9. Sathiyamoorthi, V., Ilavarasi, A. K., Murugeswari, K., Ahmed, S. T., Devi, B. A., & Kalipindi, M. (2021). A deep convolutional neural network based computer aided diagnosis system for the prediction of Alzheimer's disease in MRI images. *Measurement*, *171*, 108838.

10. Swamy, R., Ahmed, S. T., Thanuja, K., Ashwini, S., Siddiqha, S., & Fathima, A. (2021, January). Diagnosing the level of Glaucoma from Fundus Image Using Empirical Wavelet Transform. In *Proceedings of the First International Conference on Advanced Scientific Innovation in Science, Engineering and Technology, ICASISET 2020, 16-17 May 2020, Chennai, India*.
11. El-kenawy, E. S. M., Eid, M. M., & Ibrahim, A. (2021). Anemia estimation for covid-19 patients using a machine learning model. *Journal of Computer Science and Information Systems*, *17*(11), 2535-1451.
12. Sreedhar Kumar, S., Ahmed, S. T., & NishaBhai, V. B. (2019). Type of supervised text classification system for unstructured text comments using probability theory technique. *International Journal of Recent Technology and Engineering (IJRTE)*, *8*(10).