Wearable Electronic Speech Aid for Smart Home

Onur Erbay, Enes Devecioğlu, Abdullah Burhan Doğan, Ahmet Çaymaz
Department of Electrical & Electronic Engineering
Abdullah Gül University
Kayseri/Türkiye

Dr. Abdulkadir Köse
Department Computer Science
Abdullah Gül University
Kayseri/Türkiye

***Abstract*— This paper presents a wearable gesture recognition system built with resistive sensors and TinyML integration for efficient and immediate data processing. The system uses a resistive-to-time digital conversion method that allows fast and energy-efficient data capture without relying on traditional ADCs. A TinyML model processes a 10-channel input from potentiometers related to finger movements, achieving an accuracy of 96% in predicting letters. These predictions are flexibly combined to form words that can be completed with a reset action. The system also demonstrates smart home management features such as LED control by initiating actions based on user-specified terms. Designed for flexibility, the device fits different hand sizes and finger lengths, making it suitable for a wide range of users. This document highlights TinyML's capabilities in wearable assistive devices, bridging communication divides and providing scalable, low-energy options. Future efforts include extending dataset compatibility for various languages, incorporating AR-oriented visual feedback, and improving IoT connectivity.**

Keywords—potentiometer, sign language, resistance-to-time-to-digital converter, tinyML Introductıon

I. INTRODUCTION

Communication is a crucial aspect of human existence, enabling individuals to engage and interact in both personal and work environments. Nevertheless, for those with hearing and speech difficulties, communication may prove to be difficult. They frequently encounter substantial obstacles, resulting in isolation and challenges in everyday living. Worldwide, approximately 360 million individuals have hearing or speech disabilities [6]. According to the Ministry of Family and Social Services [4], as of 2023, Türkiye has 33,686 people with speech and language disabilities. These obstacles emphasize the necessity for creative approaches to enhance communication for individuals with these difficulties.

Sign language is an effective means of closing this gap, acting as a "language of hearing" for individuals with speech impairments. Nonetheless, many individuals without these disabilities are not knowledgeable about sign language, making communication challenging. Translators and assistive technologies are frequently needed to address this problem. There are two primary methods for recognizing sign language: vision-based systems and sensor-based approaches. Vision-based systems depend on cameras and sophisticated algorithms to identify gestures; however, factors such as lighting, background interference, and restricted movement frequently render them impractical. Conversely, sensor-driven systems utilize wearable technologies such as gloves fitted with sensors to detect gestures. These systems are more portable and dependable, making them more suitable for numerous applications.

This initiative goal on a sensor-driven method, utilizing gloves integrated with resistive sensors to convert Turkish Sign Language (TSL) into written form. The system assists individuals with speech and hearing challenges in communicating efficiently while also enabling them to manage smart home devices. This dual-function capability renders it a practical and essential instrument for daily living. Through the integration of sophisticated microcontrollers, optimized algorithms, and IoT features, the system provides a viable solution for communication and home automation. The system functions using a letter-oriented recognition method, where each gesture that aligns with letters in Turkish Sign Language is recognized and handled. These letters are subsequently joined together to create words and sentences, allowing users to communicate effectively and deliver their messages. This method connects solitary gestures with significant language formations, enhancing the system's adaptability for real-world communication situations. The system that has been proposed is illustrated in figure 1.

A key feature of this project is its focus on affordability, accessibility, and energy efficiency. The system uses lightweight gloves equipped with sensors, ensuring comfort for users while providing reliable performance with minimal power consumption. By employing a resistance-to-time-to-digital conversion approach, the system efficiently measures gesture data, converting physical resistance changes into precise digital signals. This method not only reduces computational overhead but also enhances accuracy and responsiveness. The system uses Wi-Fi for wireless communication between the glove modules and the receiver. It employs the HTTP protocol to efficiently send and receive sensor data, ensuring reliable and real-time performance for gesture recognition and smart device control. This efficient and innovative approach makes the system well-suited for a variety of real-world applications, including communication support and smart home control.

Overall, this project aims to break down communication barriers while empowering users to interact with their surroundings more independently. By combining sign language recognition with smart home control, it demonstrates how technology can create a more inclusive and connected world.

The rest of the paper is organized as follows. In Section II, we explain the methodology, which includes the hardware and software design. In Section III, we present the results and discussion, including key finding. Finally, we conclude the study in section IV suggesting improvements.

II. Literature Review

Many efforts have been made to help individuals with hearing and speech impairments communicate more easily using modern technology. Traditional methods include vision-based systems that use cameras to interpret sign language gestures and sensor-based systems that rely on gloves with built-in sensors. Koli et al. [1] created a system that includes a web camera placed in front of the user, who has colored rings on the fingers. This enables the system to utilize image processing for tracking the precise positions of the rings in the pictures. These ring locations are contrasted with earlier saved images to interpret letter symbols. This approach enables the user to convey full sentences, that are subsequently converted into generated speech. In 2017, Rathi et al. [1] used image and video processing technology to create a two-way communication system for deaf individuals and hearing people. The system is capable of capturing a video of hand movements for deaf individuals, analyze this video, and transform it into text and/or audio that an average person can comprehend.

Satpute et al. [2] proposed employing a glove equipped with sensors and microcontrollers for processing the glove data and generate speech signals; however, it overlooked numerous hand movements, due to the system being confined to brief specific movements. Kasar et al. [7] introduced a system featuring gloves equipped with sensors to monitor finger movements and gestures. A microcontroller shows a message on an LCD display, and this message is also transformed into audio, which is emitted through a speaker. Nonetheless, this effort was restricted by only specific gestures for home control. Building on these ideas, our project introduces a glove-based system designed to address these challenges and provide a versatile solution for real-world use.

III. Methodology

The methodology of this project emphasizes creating a system that integrates hardware and software elements to accurately identify gestures and manage devices. The block diagram presented below illustrates how the hardware components gather sensor data, which is subsequently analyzed by the software to recognize gestures and execute actions.

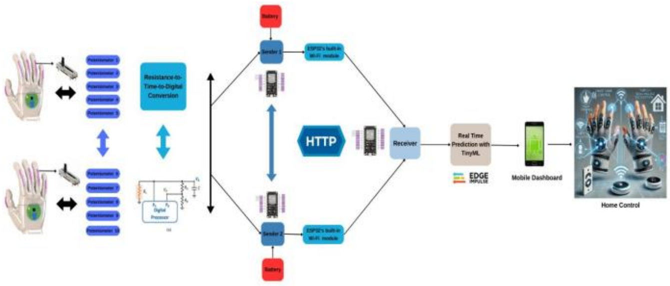

Figure 3.1. Block diagram of proposed system

A. Hardware Design

In the hardware part, choosing the right sensor for each finger plays a critical role. To achieve low cost and high accuracy and given the sensor reading mechanism we planned to use, a resistive sensor was deemed necessary. Initially, we considered using flexible sensors as they provide better visual integration with the glove. However, due to the high cost of commercially available flexible sensors, we decided to produce our own sensors. Unfortunately, the sensors we produced did not meet the expected standards of stability and accuracy because they were made using very simple and low-cost methods. For example, in flexible sensors produced using paper painting technique, it was not possible to distinguish between a half bend and a full bend of the finger, because even small bends would cause the sensor to drop to the minimum resistance value.

To address these issues, it has been decided to use potentiometers that meet the requirements of stability, accuracy and low cost. The potentiometers chosen have a resistance range of 0-50 kΩ, as this range gave the most accurate results with the resistance-to-time digital converter circuit (see figure 4.1).

Figure 3.2. The chosen potentiometers

After deciding which sensor to use, a resistance-to-time digital converter approach was implemented to enable real-time, fast sensor data reading while minimizing energy consumption.

Direct Interface Circuits (DICs) are simple and efficient systems used for sensor readings, especially with resistive sensors. These circuits perform digital conversion from resistance to time using only the sensor, calibration resistors, one or two capacitors and a digital processor (DP). Traditional DICs usually require multiple charges and discharge cycles of a capacitor to estimate the sensor resistance (R_x). None of these circuits need to incorporate analog-to-digital converters (ADCs). The DIC actually performs a time-to-digital-to resistance conversion in which the time measurements come from carrying out several capacitor charging-discharging processes which increase acquisition time and energy consumption [5].

In the proposed system, Single Capacitor Interface is used as seen in figure 3.3. The circuit utilizes only two DP pins are used to control the process to estimate R_x which is potentiometer in the system. The circuit has a single capacitor, C, but two more resistors, R_A and R_B. The figure 3.4 shows the waveforms of V_A and V_B during the single charging-discharging process carried out by the circuit to estimate R_x.

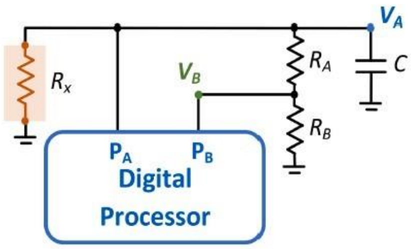

Figure 3.3. Proposed single capacitor interface (SCI) circuit

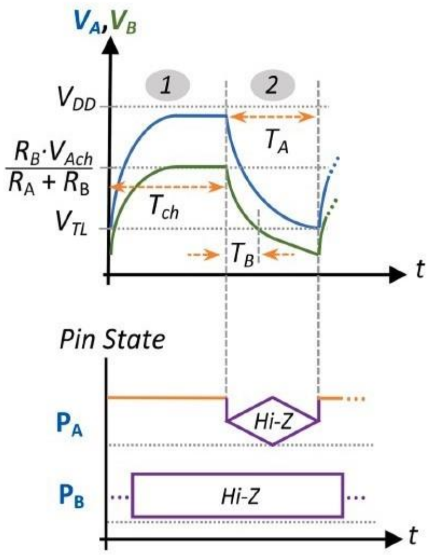

Figure 3.4. Waveforms and pin states for the circuit reading process with Fixed charging time, T_{ch}

As shown in the graph, initially, pin A should be set to logic 1 to charge the capacitor, while pin B remains in high-Z mode. Once the capacitor reaches its maximum voltage ($V_A = V_{Ach}$), it should be discharged by setting both pins to the high-Z state. V_A is slightly lower than V_{DD} when charging, but in this case due to the ground connections of R_x and the equivalent resistance $R_A + R_B$. The final voltage stored in V_B, V_{Bch}, is given by

$$V_{Bch} = \frac{R_B \cdot V_{Ach}}{R_A + R_B} \tag{1}$$

The design condition must fulfil the following condition, where V_{TL} is the threshold voltage of the microcontroller.

$$\frac{R_A}{R_B} < \frac{V_{Ach}}{V_{TL}} - 1 \tag{2}$$

Also, it is important to mention that the initial value of V_A during the charging process is not critical for the proper functioning of the circuit.

Based on this principle, the T_{ch} is calculated to ensure that the capacitor is fully charged in the circuit. For this calculation, it is essential to analyze equivalent circuit of SCI circuit. In charging phase, capacitor is charging directly via P_A. Additionally, R_A and R_B have an effect because they will reduce the charging current of capacitor as the voltage V_A increases. In the discharging phase, the capacitor is discharged by R_x and $R_A + R_B$ in parallel (R_{eq}). Based on the considerations, the equivalent circuit is given in figure 3.5.

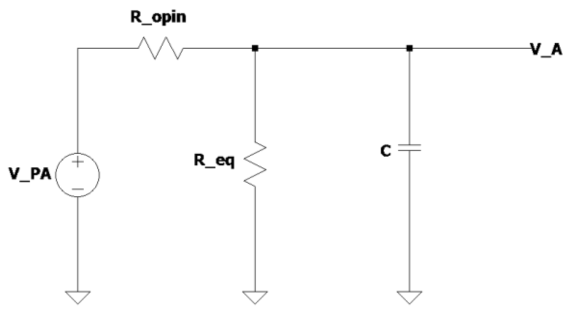

Figure 3.5. Equivalent circuit of SCI circuit

In the given equivalent circuit, R_{opin} is the output resistance of a pin. The output resistance of a digital pin for the ESP32 microcontroller is assumed to be approximately 50 Ω based on its typical electrical characteristics. This value represents the internal resistance of the pin when it drives a load in either high or low state. V_{PA} is the supplied voltage from P_A.

$$R_{eq} = R_x || (R_A + R_B) \tag{3}$$

$$V_A = V_{DD} \cdot \frac{R_{eq}}{R_{eq}+R_{opin}} + \left(V_{TL} + V_{DD} \cdot \frac{R_{eq}}{R_{eq}+R_{opin}}\right) \cdot e^{-t/R_{||}} \tag{4}$$

$$R_{||} = R_{opin} \,||\, R_{eq} \approx R_{opin} \tag{5}$$

In eq.5, $R_{||}$ is approximately same with R_{opin} because output resistance of a pin is very low. In our design, R_A is chosen as 10kΩ, R_B is chosen as 20kΩ. The capacitor value is $10\mu F$. From these values, the charging time of the capacitor (T_{ch}) is given as 20ms in the algorithm. After that time, the capacitor is discharged by changing the pin state and Times T_A and T_B are measured in this step via microcontroller correspond to the intervals from the start of the discharge at the instants when $V_A = V_{TL}$ and $V_B = V_{TL}$ respectively. Therefore, the resistance of potentiometer is estimated by eq. 6.

$$R_x = \frac{(R_A + R_B) \cdot \Delta T}{(R_A + R_B) \cdot C \cdot \ln\left(\frac{R_A + R_B}{R_B}\right) - \Delta T} \tag{6}$$

$$\Delta T = T_A - T_B \tag{7}$$

In summary, this method allows for real-time or near real-time reading of the resistance values of the potentiometer based on finger bending movements, without using an ADC, while ensuring fast operation and low energy consumption. After testing the system, a separate circuit was built for each potentiometer and soldered onto a phenolic board for integration into the glove. In total, 10 circuits were built with identical values, 5 for each hand as shown in figure 3.6.

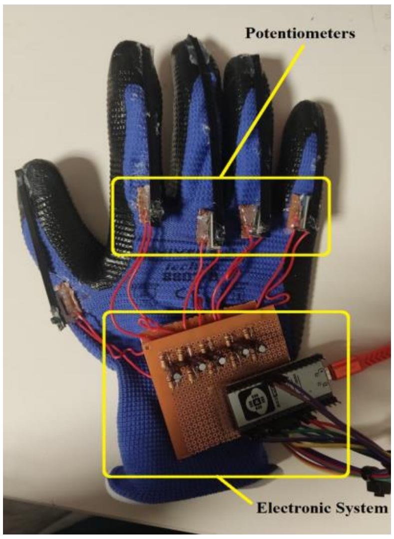

Figure 3.6. Glove based system design

B. *Software Part*

The software component of this project supports its functionalities, including smooth interaction between transmitter and receiver units, an easy-to-use dashboard for real-time monitoring, and the integration of machine learning algorithm (TinyML) for efficient gesture recognition. This part examines the main algorithms, communication protocols, and design choices that ensure dependable performance and adaptability of the system.

The software part of the system is developed using Arduino C++ on the ESP32 platform. The system consists of two sender devices (ESP32), one receiver device (ESP32) and a web interface working together to measure, process and visualize data in real time. The system's overall framework involves two sender ESP32 modules integrated into two hands, which simultaneously transmit the measured resistance values to a receiver ESP32 module.

Each sender module captures resistance changes caused by finger bending by measuring resistance values from five sensors attached to the fingers. Sender 1, placed on the right hand, measures values labelled R1 to R5, while Sender 2 on the left-hand measures R6 to R10. These resistance values, generated by potentiometer readings corresponding to finger movements, are formatted in a JSON structure to enable efficient and standardized data exchange. Known for its lightweight and easily parseable format, the use of JSON enables optimized transfer of sensor data between devices. The sending modules connect wirelessly to a Wi-Fi network and transmit these formatted values to the receiving ESP32 using the HTTP protocol, ensuring reliable and low latency communication. This wireless architecture supports accurate gesture recognition and enables streamlined integration with downstream processing stages.

The receiver collects data from both senders, processes it, and provides outputs. It uses a TinyML model to predict letters based on the resistance data received from the senders, combining these predictions dynamically to form words. A word is finalized when all sensors return to their reset positions, indicating that all fingers are fully extended with sensor values at zero. Finalized words are used to control physical components such as LEDs, demonstrating the system's potential for remote interaction. For instance, specific words can trigger actions like turning a LED on or off, illustrating how the glove can be used for various control tasks.

Furthermore, the system enables seamless communication between the user and others who may not understand Turkish Sign Language. By simply entering the receiver ESP32's IP address into a browser, a user can provide their smartphone to the other person to facilitate real-time communication. This feature highlights the glove's versatility, showing its potential application in smart home systems and remote-control scenarios. This project demonstrates how the glove can bridge communication gaps and control devices remotely through simple gestures and letter-based word construction.

The web interface displays processed data and provides a visual representation of the system's output. In this customized interface resistance values in a bar chart, showing real time sensor activity. Above the bar chart, the current letter being predicted and the word being finalized and finalized word are displayed as can be seen in figure 3.7. The web interface design is implemented in the receiver code. The web interface implemented using HTML, CSS and JavaScript, ensuring real-time visualization and user-friendly interaction.

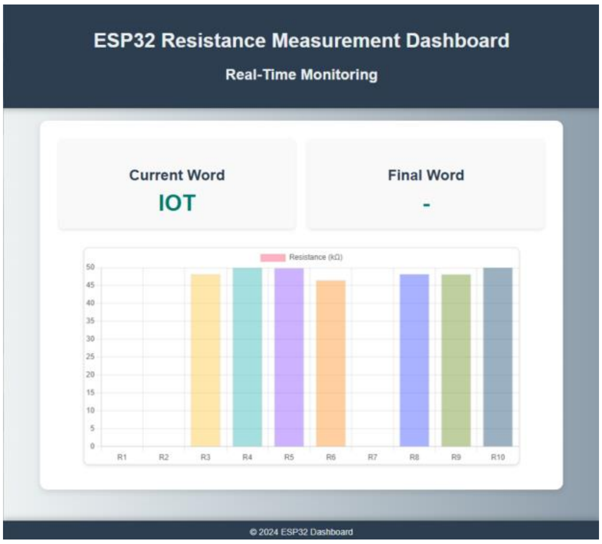

Figure 3.7. User interface

For the machine learning part, TinyML was used to implement the machine learning aspect of the system. It is a technology that aims to run machine learning models on small, low-power devices. It is widely used especially in embedded systems and IoT devices because it reduces cost and optimizes resource usage [3]. TinyML uses machine learning models optimized to run on microcontrollers. These devices usually consume milliwatts of power. In our system, one of the purposes of using TinyML is its low power consumption alongside our analog circuit. On the software side, it has saved time and reduced complexity by minimizing the amount of code. In the system, TinyML can also learn data that varies between different users. Thus, the system adapts to each user.

To implement TinyML in our system, a data set was first created. This data set was created by creating CSV files with an algorithm written by phyton. Each line in the CSV files contains 10 pieces of data. The resistance values for the potentiometers on each finger are shown in this data. While creating the 10 data, specific resistance values were determined for each letter in the Turkish alphabet. Since these values contain different combinations for each of the 29 letters in the Turkish alphabet, classification could be made. In addition, while creating the CSV files, the test and training data sets were adjusted as desired with the phyton code algorithm. In our case, 1000 training sets and 200 test sets were used. Each separate file was made ready for loading edge impulse on the interface.

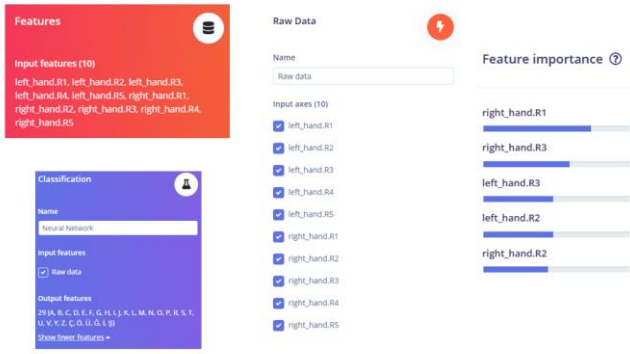

Figure 3.8. Settings configured in the "Create Impulse" section of the Edge Impulse interface

In the Neural Network settings section, 100 cycles were used for training and 0.001 for learning rate. For the neural network architecture, Excluding the input and output layers, 30 layers were used: 20 in the second row and 10 in the third row as shown in figure 3.9. As a result of all the settings, the training data model was trained with machine learning and an accuracy rate of 99.4% was achieved. The fact that there is a unique combination for each letter has a great effect on obtaining this high rate. Achieving a high accuracy rate in the training dataset is crucial for accurately predicting the outcomes of the test data during the model testing phase. It is evident that if the model fails to learn from the training dataset, it will inevitably produce low accuracy in the model testing phase.

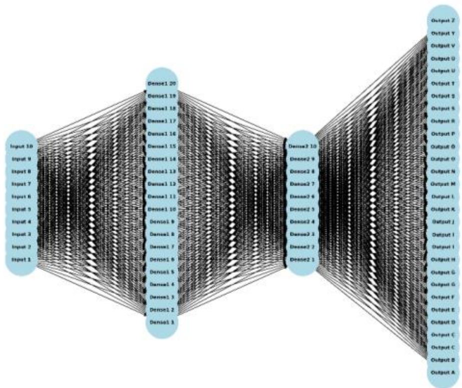
Figure 3.9. Neural network structure with Turkish Alphabet

Finally, the test data in the previously prepared "data acquisition" section was estimated to assess the effectiveness of the model's training. As a result of this estimate, an accuracy rate of 96% was obtained. As can be seen in figure table 1, even though every letter can be estimated correctly by the model, the rate of some letters is lower than others. The biggest reason for this is that only 1 out of 10 data received for some letters creates a difference. This situation can be improved by making the data received from the sensor more specific and defining it with narrower intervals. Nevertheless, a rate of 96% was considered sufficient in our system and this model was taken from the IDE library edge impulse to be used in ESP32. While using the library taken from edge impulse, the system was considered as a basic

block operation. 10 data on which TinyML was trained were thrown into the library, considering their order, and it was expected that TinyML would give an output because of this data and make a prediction. The outputs and predictions from the library are labels as previously configured, and in our system, the label for each letter is directly set as the letter itself. Based on the obtained predictions, these labels, i.e., the predicted letters, were sent to the dashboard to enable writing.

To conclude the methodology, the combination of hardware and software elements, enhanced by the addition of TinyML for on-device motion recognition, establishes a strong basis to reach the project's objectives. This extensive design guarantees real-time performance, energy efficiency, and scalability, rendering the system flexible and suitable for various practical applications. The suggested system effortlessly integrates hardware and software features, creating a novel solution that links accessibility, cost-effectiveness, and technological progress.

IV. RESULT & DISCUSSION

In this chapter, the test results of the designed systems in the hardware and software parts will be presented. As mentioned in the hardware design part, the maximum resistance of potentiometer is selected as 50kΩ. Because resistance-to-time-to digital converter operates in this resistance range based on our tests.

To test the circuit, fixed resistors with various resistance values were estimated by the system. The primary goal of this test was to determine the resistance range where the system exhibited the least error. The system was tested with fixed resistors ranging from a minimum of 100 Ω to a maximum of 500 kΩ. (The values of R_A, R_B, and the capacitor were optimally selected based on the magnitude of the tested resistance.) As shown in Figure 4.1, the error rate within the range of 1 kΩ to 50 kΩ was significantly lower than the average error rate which is 8.40%. Observations during glove usage further indicated that the potentiometer resistance, based on finger position, typically falls within this range (1k Ω-50k Ω) or at 0 Ω. Based on the test results, the range of 0-50 kΩ was selected for the potentiometer to be used.

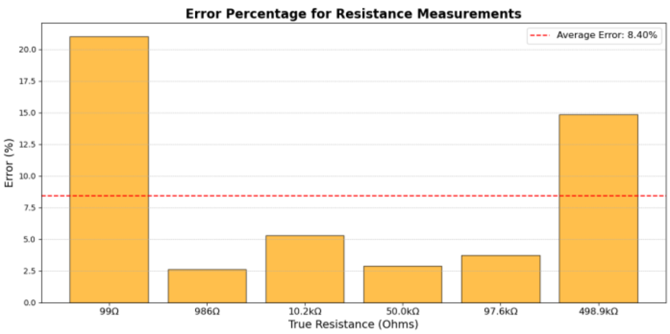

Figure 4.1. SCI circuit resistance measurement test results

The table below displays the precision, F1-score, and recall values of the TinyML model for each letter. These metrics illustrate the model's effectiveness in correctly predicting letters, with precision indicating the accuracy of positive predictions, recall assessing the model's capacity to identify true positives, and F1-score balancing precision and recall. The findings demonstrate the system's effectiveness with various letters, emphasizing its strengths and aspects that need enhancement.

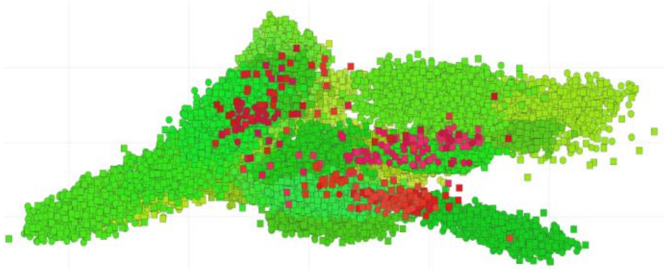

Figure 4.2. All training and test datasets are represented. Squares indicate test data, while circles represent training data.

In Figure 4.2, all the data used for training and testing is shown. In this figure, the green and red sections within the square blocks represent the correct and incorrect predictions of the test data uploaded during data acquisition. The circular symbols, which represent the training data, indicate the correct and incorrect classifications that occurred while training the model to classify the letters. As can be observed from the graph, the majority is green, indicating that the model operates with high accuracy.

TABLE 1. F1-Score, precision and recall for each letter

Letters	F1-Score	Precision	Recall
A	0,98	0,98	0,97
B	0,91	0,94	0,88
C	0,88	0,91	0,86
D	0,99	0,99	0,98
E	0,99	0,99	0,99
F	0,99	0,99	0,98
G	0,93	0,91	0,94
H	1	1	1
I	0,99	0,99	1
J	0,95	0,59	0,91
K	0,99	0,97	1
L	1	1	1
M	0,97	0,97	0,97
N	0,99	0,99	0,99
O	0,99	0,99	0,99
P	0,99	0,98	0,99
R	0,93	0,92	0,94
S	0,99	1	0,99
T	1	1	1
U	0,84	0,84	0,83
V	0,99	1	0,97
Y	0,99	1	0,97
Z	1	1	1
Ç	0,88	0,99	0,96
Ö	0,99	0,99	0,99
Ü	1	1	0,99
Ğ	0,86	0,87	0,85
İ	0,95	0,94	0,95
Ş	0,94	0,96	0,92

The inclusion of TinyML in the project presented a number of challenges. At first, the system could have been developed with traditional ML methods, determining resistance values for both half-bend and full-bend positions and using event-driven programming with if-else statements. This method could have realized total accuracy. However, considering that the product will be used by a variety of people with different hand and finger sizes, TinyML was chosen for its flexibility and learning potential.

Using TinyML greatly reduced the complexity of the receiver's code. However, the letter prediction accuracy dropped to 96%. This drop led to occasional mispredictions during letter transitions. Nevertheless, TinyML enabled the device to successfully transmit the intended words to the receiver and effectively fulfill its purpose. Moreover, the device provides the ability to manage your home, as detailed in the software section. This demonstrates that gestures can be used as a communication tool as well as to interact remotely with smart environments, increasing the versatility of the project.

This initiative lays a strong foundation for gesture-driven communication and control systems and offers promising opportunities for future advances. The inclusion of 3D motion tracking sensors can improve the accuracy of gesture recognition, and AR-enabled smart glasses can offer immediate visual feedback for the user and receiver, making the system more user-friendly and approachable. Custom TinyML training interfaces will enable users to modify the device to suit their own hand and finger measurements, increasing its usability among various groups. Expanding the dataset to include various sign languages could further increase its global reach. Furthermore, improving compatibility with IoT devices for effortless smart home management and maximizing energy efficiency for better portability could turn this device into a versatile tool for everyday use.

V. Conclusion

This project effectively incorporates TinyML into a wearable device for gesture-driven communication and control, using a novel resistance-time-to-digital converter (RTD) method for fast and energy-efficient data acquisition. The RTD mechanism enables the system to monitor resistance changes in real-time without relying on traditional ADC techniques, greatly reducing complexity and energy use. When paired with a TinyML model, the system achieves a letter prediction accuracy of 96% and efficiently converts gestures into words while remaining flexible for different hand sizes. Furthermore, the device's connectivity with smart home systems demonstrates its adaptability, providing opportunities for seamless management of IoT devices. Future developments such as AR-enabled smart glasses, 3D motion detection and larger sign language databases could further enhance their functionality. This initiative represents a significant advance in energy-saving assistive technologies, eliminating communication barriers and facilitating natural interaction with the environment.

References

[1] Alzubaidi, M. A., Otoom, M., & Abu Rwaq, A. M. (2023). A novel assistive glove to convert arabic sign language into speech. ACM Transactions on Asian and Low-Resource Language Information Processing, 22(2), 1-16.

[2] J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68-73.

[3] I.S. Jacobs and C.P. Bean, “Fine particles, thin films and exchange anisotropy,” in Magnetism, vol. III, G.T. Rado and H. Suhl, Eds. New York: Academic, 1963, pp. 271-350.

[4] K. Elissa, “Title of paper if known,” unpublished.

[5] R. Nicole, “Title of paper with only first word capitalized,” J. Name Stand. Abbrev., in press.

[6] Y. Yorozu, M. Hirano, K. Oka, and Y. Tagawa, “Electron spectroscopy studies on magneto-optical media and plastic substrate interface,” IEEE Transl. J. Magn. Japan, vol. 2, pp. 740-741, August 1987 [Digests 9th Annual Conf. Magnetics Japan, p. 301, 1982].

[7] M. Young, The Technical Writer’s Handbook. Mill Valley, CA: University Science, 1989.

V.