KULLIYYAH OF INFORMATION AND COMMUNICATION TECHNOLOGY

DEPARTMENT OF COMPUTER SCIENCE

FINAL YEAR PROGRESS REPORT

PROJECT ID

1103D

PROJECT TITLE

WEARABLE GLOVE FOR SIGN LANGUAGE TRANSLATION USING MACHINE LEARNING

STUDENT(S)

1. BAWAZIER NAJLA GEIS JUNAID (2017868)
2. ABDULLAH ABDULRAHMAN (1923111)

SUPERVISOR

ASSOC. PROF. TS. DR. AMELIA RITAHANI BT. ISMAIL

JANUARY 2024

SEMESTER 1 2023/2024

FINAL YEAR PROGRESS REPORT

PROJECT ID

1103D

PROJECT TITLE

WEARABLE GLOVE FOR SIGN LANGUAGE TRANSLATION USING MACHINE LEARNING

PROJECT CATEGORY

DEVELOPMENT

by

1. BAWAZIER NAJLA GEIS JUNAID (2017868)
2. ABDULLAH ABDULRAHMAN (1923111)

SUPERVISED BY

ASSOC. PROF. TS. DR. AMELIA RITAHANI BT. ISMAIL

In partial fulfillment of the requirement for the

Bachelor of Computer Science

Kuliyyah of Information and Communication Technology

International Islamic University Malaysia

ACKNOWLEDGEMENTS

Alhamdulillah, praise be to Allah the Almighty, the Most Gracious, and the Most Merciful. Firstly, we express gratitude to Allah for His blessings in the completion of this thesis. Subsequently, praises are also directed to our Prophet Muhammad SAW, who has guided us to a better life today.

Foremost, we owe an immense debt of gratitude to Assoc. Prof. Ts. Dr. Amelia Ritahani Bt. Ismail, whose steadfast guidance played a pivotal role in the success of this project. Her contributions, ideas, and ongoing support have profoundly influenced this project, and we genuinely appreciate having such a committed and knowledgeable supervisor guiding us. Dr. Ahmad Anwar bin Zainuddin deserves heartfelt thanks for his invaluable contributions. Our profound appreciation also goes to our parents for their unending support, which has been our pillar of strength.

Our gratitude extends to Hamed Hamed, Abdulaziz Alyazidi, Arina Abi, Abdulrahman Alaidaros, Aina Iffah, and others for their significant help and contributions. Special thanks to Aisha Arif and Alaa Bawazier for their exceptional contribution to the project's success. Last but not least, sincere thanks to Shaima, Huma, Areej, Tafany, 20 Maret, Meri Jaans, and those unnamed contributors who indirectly played a crucial role in this project. Your support and motivation are deeply appreciated and have significantly contributed to the success of this project.

ABSTRACT

To address communication challenges for the deaf community, this project introduces an innovative wearable glove designed to translate sign language into text. The primary objective is to bridge the communication gap between deaf individuals and those unfamiliar with sign language. Focusing on precision, the glove features flex sensors and a gyroscope to detect sign language gestures accurately. A concise yet impactful data collection phase involved participants performing ten sign language gestures, setting the foundation for a sophisticated K-Nearest Neighbors (KNN) machine learning model. The deployment of this technology was achieved through the Django web framework, resulting in a user-friendly web system for real-time translation. The model showcases an impressive accuracy of 92%, validated through thorough testing, confirming the system's effectiveness and successful operation in real-world scenarios. This achievement emphasizes the solution's technical adeptness and signifies notable progress toward broader inclusivity.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF APPENDICES

LIST OF ABBREVIATIONS

KNN	K Nearest Neighbor
IoT	Internet of Things
SVM	Support Vector Machine
HMM	Hidden Markov Model
CNN	Convolutional Neural Network
CHC	Cross generational elitist selection, Heterogeneous recombination, and Cataclysmic mutation
ORB	Oriented FAST and rotated BRIEF

CHAPTER ONE

INTRODUCTION

1.1 Project Overview

Communication barriers exist for the deaf community, making it challenging to interact with the hearing world. When interacting with individuals who do not understand it, the traditional manner of employing sign language is not always efficient, which can cause misunderstandings and social isolation. While there are currently existing systems that aim to facilitate communication between the deaf and hearing communities, they still fall short in terms of effectiveness and efficiency, often resulting in misinterpretations and communication breakdowns. Our new project tries to address this issue by developing a wearable glove that uses machine learning algorithms and IoT (Internet of Things) technology to detect specific sign language gestures for different alphabets and translate them into text. The gloves will be equipped with flex sensors that can recognize the movement and the bending of the fingers. The data read by the sensors will then be analyzed using machine learning algorithms to improve the accuracy of the translation. The project's primary focus will be on detecting gestures for specific alphabets, and the results will be tested for both accuracy and real-time translation capabilities. The wearable glove can make communication more efficient for members of the deaf community.

1.2 Problem Statement

Communication has long been a significant barrier for individuals within the deaf community, creating substantial challenges in their interactions with the broader world. According to the World Health Organization, over 5% of the world's population, including 432 million adults and 34 million children, require rehabilitation to address their disabling hearing loss. By the year 2050, it is expected that this number will exceed 700 million. For many who are hard of hearing, sign language serves as a crucial mode of communication, enabling them to connect with others who share this language. However, despite the widespread use of sign language within the deaf community, a significant communication divide persists between those who are deaf and those who do not comprehend sign language. This gap often leads to reduced social interactions and limited experiences for deaf individuals, as it necessitates the presence of a proficient sign language interpreter to facilitate communication.

Moreover, it is not easy for people to learn sign language in a short time, but as the world evolves, inclusivity is increasingly needed for those in the deaf community. This communication gap needs to be addressed. Although various solutions have been developed to bridge this gap, such as employing sign language interpreters or utilizing text-to-speech systems, these methods are not without their limitations. The introduction of this project, which focuses on providing a reliable and accessible communication tool for the deaf community, holds the potential to significantly enhance the quality of life for countless individuals around the globe who are affected by hearing loss. By offering a practical solution to this long-standing issue, the project aims to foster more inclusive and effective communication, thereby enriching the social and interactive experiences of those within the deaf community.

The challenge faced during this study primarily revolves around the physical design of the glove, particularly concerning the circuitry and sensor functionality. To optimize the system, numerous adjustments to the glove's circuit were necessary to establish the best connections. These modifications were critical in ensuring that the glove could consistently and accurately interpret sign language gestures. However, a notable issue arose with the sensors: due to frequent bending and extensive use, the sensors sometimes failed to work effectively. This mechanical stress led to the sensors occasionally producing incorrect value readings, which in turn could result in inaccurate translations of sign language gestures.

In addition to these technical hurdles, the glove's performance was also impacted by the subtle differences in sign language and the diversity of hand sizes among users. Sign language contains gestures that are similar in appearance but differ in meaning, posing a challenge for the glove's sensor system to accurately distinguish between these nuances. Moreover, the varying hand sizes of different individuals affected how the glove's sensors interpreted gestures. This variation could lead to inconsistencies in translation, as the glove might not be calibrated to accommodate the full range of hand sizes and movements. Consequently, while the glove demonstrated good accuracy overall, these factors highlighted the need for further refinement to enhance its reliability and versatility in translating sign language for a diverse user base.

1.3 Project Objectives

This project focuses on implementing the Internet of Things (IoT) and machine learning as a solution to the problem statements as stated previously, the main objectives are as follows:

1. To design a wearable glove that integrates flex sensors and IoT technology for real-time sign language translation.
2. To apply a suitable algorithm that can accurately identify and detect sign language gestures and improve the accuracy of real-time translation.
3. To validate the wearable glove for real-time sign language translation.

1.4 Significance of Project

This project stands out for its ability to significantly enhance communication and interactions between individuals who are deaf or hard of hearing and the broader community. At its core is an innovative wearable glove that expertly translates sign language into text, ensuring accuracy in conveying messages. This technological advancement promises to revolutionize accessibility in various environments such as public spaces, educational institutions, and business settings, making them more inclusive for those with hearing impairments. The glove's portability and ease of use offer a new level of independence to deaf individuals, empowering them to engage in conversations without the traditional barriers they often face. By bridging the communication gap in such a direct and effective manner, the project holds the potential to significantly improve the day-to-day experiences of the deaf community, fostering greater participation in social, educational, and professional activities.

Moreover, the project's impact extends beyond the deaf community. It has promising uses in other areas such as facilitating interactions between humans and robots or enhancing virtual assistant technology, highlighting the versatility and broad influence of the technology being developed. The project also makes a valuable contribution to the field of machine learning. Developing and testing a system that translates sign language in real-time, helps to refine and improve the accuracy of machine learning algorithms used for recognizing gestures. Overall, this project is significant because it not only helps increase access and inclusion for people with hearing disabilities but also pushes forward the development of assistive technology and contributes to creating more precise machine learning algorithms.

1.5 Project Schedule

Figure 1. Gantt Chart of FYP1 SignWave System Development

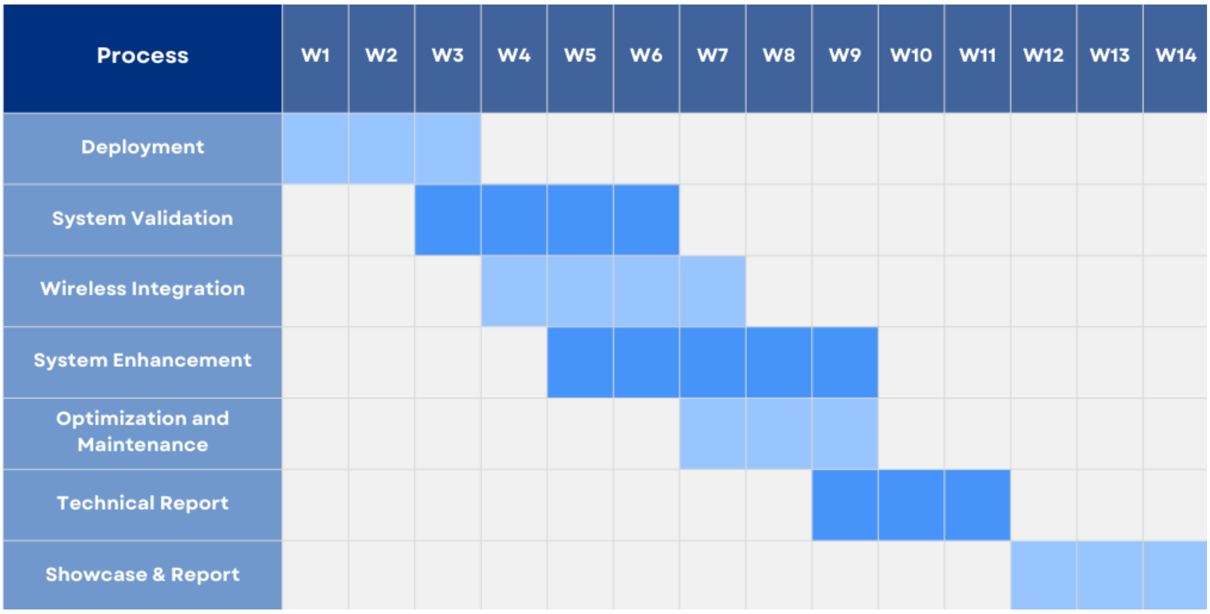


Figure 2. Gantt Chart of FYP2 SignWave System Development

CHAPTER TWO

REVIEW OF PREVIOUS WORK

2.1 IoT Sign Language Translation Approaches: Machine Learning vs. Non-Machine Learning

Heera et al. (2017) emphasized the importance of integrating machine learning techniques into gesture recognition systems. Their study focused on a Bluetooth-enabled glove and an Android smartphone system that converted detected gestures into artificial speech, improving communication for speech-impaired individuals. The system utilized a database of predefined alphabets and words, matching sensor values to generate meaningful text and speech output. Accuracy in hand gesture detection was prioritized through sensor integration, offering a user-friendly mobile solution. The authors also proposed extending the glove's functionality to gesture recognition in Virtual Reality and interacting with electronic devices via an IoT hub. Additionally, they explored the potential of integrating machine learning to enhance gesture understanding.

Choudhary et al. (2021) highlighted the role of machine learning algorithms in optimizing IoT systems and enhancing decision-making capabilities. They emphasized the importance of integrating machine learning models to improve the accuracy of the gesture classification system. Machine learning models have shown promise in sign language recognition, offering potential solutions to bridge communication gaps for individuals who are deaf or hard of hearing (Rajalingam et al., 2022).

2.2 Wearable Glove for Sign Language Translation

The primary emphasis of previous research has been on sign language translation systems that rely on visual input. Gesture recognition systems rely on two means to capture and record the shape and movement of the hand: they use a camera or a sensor. In gesture recognition systems, the use of a vision approach requires the adoption of at least one camera to capture hand images and interpret suitable gestures (Ahmed et al., 2018). A glove can be an assistive interpreter tool for hearing- and speech-impaired persons to communicate with non-disabled individuals who do not understand SL (Fu et. al, 2008). Gloves offer additional advantages in terms of mobility and comfort. Recent advancements in technology and advanced electronic circuits have allowed gloves to be independent of direct computer connection while still being capable of recognizing sign language.

Eshetu and Wolde (2019) conducted a study focusing on the real-time conversion of Ethiopian Sign Language to audio through the utilization of sensor-based technology. Their system incorporated IoT devices such as five flex sensors placed on the thumb, index, middle, ring, and pinky fingers, as well as a palm-mounted accelerometer and gyroscope (MPU6050). These sensors played a crucial role in capturing hand movements and gestures. However, it is important to note that although the system utilized IoT components, it did not rely on internet connectivity for its operation, thereby not meeting the full criteria of an IoT system. To facilitate communication, the system employed a Bluetooth module to establish a connection with a computer. The study reported promising results, achieving an average recognition rate of 88.56% for seven Ethiopian Sign Language gestures, with individual gesture recognition rates ranging from 80% to 90%. This research provides valuable insights into the development and performance analysis of a gesture recognition system specifically tailored for Ethiopian Sign Language, contributing to the advancement of communication technologies for individuals using this sign language.

2.2.1 Wearable Glove for Sign Language Translation Using IoT

The study conducted by Yudhana, Rahmawan, and Negara (2018) aimed to explore and evaluate the responses of flex sensors and MPU6050 sensors in a smart glove designed for sign language translation. The researchers proposed the use of flex sensors as transducers capable of measuring the degree of curvature to detect finger and hand movements. Specifically, they employed resistive flex sensors which produced an output resistance value proportional to the amount of flexibility of the sensor. To further enhance the glove's capabilities, MPU6050 sensors, consisting of a gyroscope and accelerometer, were integrated to measure earth gravity and provide information on hand orientation and movement. The sensor data was read using an Arduino Nano microcontroller. The study focused on assessing the effectiveness of the integrated sensor system in capturing and interpreting the relevant gestures and movements for sign language translation. The findings of the research shed light on the potential of flex sensors and MPU6050 sensors in facilitating accurate and real-time interpretation of sign language using smart gloves.

The "Smart Hand Glove" project, available on GitHub, introduces an accessible and cost-effective glove-based gesture recognition system that employs flex sensors, an accelerometer, and a gyroscope to capture hand movements and identify gestures. This system, which can be constructed using readily available hardware components, offers widespread accessibility for interested individuals. It is worth noting, however, that the current gesture recognition approach relies on a basic if-else statement, which may have limitations in accurately interpreting complex gestures. The captured gesture data is transmitted to an ESP32 microcontroller for further processing and then transferred to a smartphone or laptop via Bluetooth or Wi-Fi connectivity. Additionally, it is important to acknowledge that the current implementation focuses on a specific set of four distinct gestures, leaving room for future expansion and exploration of more advanced techniques.

Additionally, another paper presents a framework for the development of a sign language communication device using flex sensors, Arduino, a Bluetooth module, an accelerometer, a glove, and an Android application (Suri et al., 2020). The primary objective of the framework is to facilitate communication for individuals who are deaf and mute by recognizing their gestures through the integrated sensors and translating them into words displayed on the Android application. The research paper provides a comprehensive and well-documented approach to constructing a sign language recognition device utilizing flex sensors and other electronic components. The study highlights the significance of addressing the communication barrier faced by individuals with hearing and speech impairments and suggests potential enhancements by exploring additional technological options.

2.2.2 Wearable Glove for Sign Language Translation Using IoT and Machine Learning

Choudhary et al. (2021) developed a wearable device-based sign language recognition system, which employed flex sensors, an inertial measurement unit (IMU), and a Hall sensor to detect hand gestures and movements. They compared three machine learning algorithms, namely support vector machine (SVM), Naïve Bayes, and Decision Trees, for gesture classification. The SVM classifier achieved the highest accuracy of 90%, followed by Naïve Bayes (86%) and Decision Trees (88%). The integration of machine learning models significantly improved the accuracy of the system, enabling precise recognition of sign language. By incorporating IoT components such as Raspberry Pi, NodeMCU, and Firebase, the system facilitated two-way communication, speech-to-text and text-to-speech functionalities, and effective interaction for speech-disabled individuals. The study emphasized the role of machine learning algorithms in optimizing IoT systems and enhancing decision-making capabilities.

Rosero-Montalvo et al. (2018) proposed an intelligent electronic glove system designed to automate communication between deaf-mute individuals and others by detecting sign language gestures and translating them into oral language. The system incorporates flex sensors in each finger of a glove to collect hand movement data. The authors applied a methodology involving various stages, including data balancing using the Kennard-Stone (KS) algorithm, prototype selection using the CHC evolutionary algorithm or the Decremental Reduction Optimization Procedure 3 (DROP3), and implementation of the K-Nearest Neighbors (KNN) classifier. The results showed a significant reduction of data storage within the system by 98% and a classification performance of 85% using the CHC evolutionary algorithm. The study highlights the role of machine learning algorithms in enabling electronic systems to make decisions based on experience. Furthermore, the authors emphasized the importance of prototype selection algorithms, identifying CHC as the most suitable for sensor data due to its training set reduction capacity and classifier performance. While the proposed system shows promise, it is important to note that the article does not provide information about any user testing of the system. Additionally, the limitations of the LilyPad suggest that the proposed system may not be recommended for practical use.

2.3 Machine Learning for Sign Language Translation

Rajalingam et al. (2022) conducted a study on finger spelling classification in sign language using several machine learning algorithms. The study utilized Support Vector Machine (SVM) and K-nearest neighbors (KNN) algorithms for classification. SVM creates a hyperplane to separate classes in an n-dimensional space, while KNN assigns items based on majority voting among nearest neighbors. Convolutional Neural Networks (CNNs) were also employed, involving operations such as convolution, non-linearity (Relu) layers, pooling, and fully connected layers. ORB feature extraction was explored, providing a condensed yet meaningful feature vector capturing visual qualities of hand movements and achieving improved accuracy compared to previous systems. ORB offers advantages such as lower training costs and a focus on overall picture visual characteristics.

Li et al. (2018) proposed a dynamic gesture recognition approach based on Hidden Markov Models (HMM) and D-S evidence theory for human-computer interaction in the Internet of

Things (IoT) context. The study focused on improving the recognition rate of dynamic gestures, which is crucial for non-contact human-computer interaction. The authors extracted features from complex motion gestures, including gesture changes and palm motion trajectories, and used the Forward-backward algorithm, Viterbi algorithm, and Baum-Welch algorithm for classification and training of HMM models. The integration of gesture change features and hand motion trajectory features through D-S evidence theory allowed for dynamic gesture recognition. The experimental results demonstrated the effectiveness of the proposed approach in achieving accurate dynamic gesture classification and recognition. The study contributes to the field of human-computer interaction under IoT technology and has implications for applications such as smart cars, smart homes, and robots.

CHAPTER THREE

METHODOLOGY

3.1 Design

3.1.1 Physical Components

The circuit for the wearable sign language translation glove incorporates various components to enable precise gesture detection and communication, the primary components are as follows:

1. *Flex Sensors:*

A flex sensor is an electronic component that utilizes a strip of plastic coated with carbon to function as a variable resistor. Its resistance varies in response to bending. When the sensor is flexed in one direction, the resistance increases proportionally to the degree of bending, altering the electrical input readings. This characteristic makes flex sensors ideal for detecting and measuring the amount of flexion in applications such as motion tracking and gesture recognition.

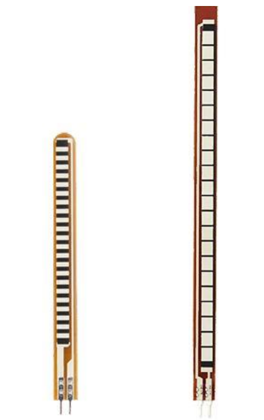


Figure 3. Flex Sensors

To ensure precise gesture recognition and optimal ergonomic comfort, this glove is outfitted with a specialized configuration of sensors, each designed for the unique requirements of different fingers:

- Index, Middle, and Ring Fingers (3 sensors of 4.5 inches): These sensors are strategically placed on the index, middle, and ring fingers to detect their bending and flexing movements. This design is crucial for accurately tracking the nuanced gestures associated with sign language expression.
- Pinky and Thumb (2 sensors of 2.2 inches): The pinky and thumb are equipped

with smaller sensors to ensure a precise and comfortable fit. This consideration not only enhances the accuracy of gesture recognition but also accounts for the anatomical variations in finger sizes, leading to a more tailored and comfortable experience for the user.

```
// get the reading
 int flexValueP = analogRead(flexPinP);
 int flexValueR = analogRead(flexPinR);
 int flexValueM = analogRead(flexPinM);
 int flexValueI = analogRead(flexPinI);
 int flexValueT = analogRead(flexPinT);
// map the inputs
 String t_value = String(map(flexValueT, 0, 130, 0, 5000)) ;
 String i_value = String(map(flexValueI, 0, 520 , 0, 5000)) ;
 String m_value = String(map(flexValueM, 0, 520, 0, 5000));
 String r_value = String(map(flexValueR, 0, 520, 0, 5000)) ;
 String p_value = String(map(flexValueP, 0, 130, 0, 5000)) ;
```

Figure 4. Code of The Flex Sensors Mapping

This code reads analog values from flex sensors and scales them using the "*map*" function to a standardized range (0-5000). The specific details of the mapping are:

a. For 2.2 inches sensors (pinky and thumb):
   - map(flexValueT, 0, 130, 0, 5000)
   - map(flexValueP, 0, 130, 0, 5000)

   Raw values are expected to fall in the range of 0-130.
b. For 4.5 inches sensors (index, middle, and ring):
   - map(flexValueI, 0, 520, 0, 5000)
   - map(flexValueM, 0, 520, 0, 5000)
   - map(flexValueR, 0, 520, 0, 5000)

   Raw values are expected to fall in the range of 0-520.

This scaling ensures that all sensor readings, regardless of their original ranges, are normalized to a common scale of 0 to 5000, facilitating consistent and comparable interpretation of the flex sensor data. This normalization ensures consistent and comparable readings across different sensors, accounting for the variations in sensor output based on their sizes (2.2 inches and 4.5 inches).

2. *ESP32 Microcontroller*

The ESP32 is a highly integrated, low-cost system-on-a-chip (SoC) microcontroller developed by Espressif Systems. It combines a dual-core processor, Wi-Fi connectivity, Bluetooth, and a rich set of peripheral interfaces, making it well-suited for a wide range of applications in the Internet of Things (IoT), embedded systems, and wireless communication projects.

Figure 5. ESP32 Microcontroller

The ESP32 uses its numerous analog inputs to connect with flex sensors, reading their analog values via the analogRead function in Arduino IDE. It also communicates with the MPU6050, a motion-tracking device with a gyroscope and accelerometer, through I2C. The integrated Wi-Fi is configured for wireless connectivity, facilitating data exchange. In real-time, the microcontroller processes analog data from flex sensors and motion data from MPU6050, employing algorithms to interpret sign language gestures and movements.

3. *MPU 6050 Gyroscope*

MPU-6050 is a motion-tracking device that integrates a three-axis gyroscope and a three-axis accelerometer on a single chip. The accelerometer plays a critical role in detecting hand movements, which is essential for discerning nuances between words that share identical finger representations but exhibit distinct hand positions. By harnessing x and y accelerometer data, the system can compute changes in hand rotation over time, enhancing the precision of differentiation. This integration of orientation and motion data empowers the microcontroller to effectively recognize intricate hand movements and variations in sign language gestures, leading to a heightened level of accuracy in the translation process.

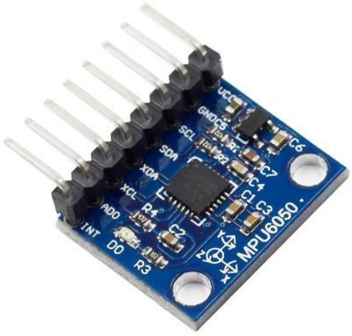


Figure 6. MPU6050 Gyroscope

4. Jumper Wires and Resistors

In the design of the glove, the integration of flex sensors plays a pivotal role. These sensors are soldered to create a secure connection with the ESP32 microcontroller, which is crucial for reliable communication. The flex sensors are adeptly stitched onto the back of each finger, enabling accurate tracking of finger movements. Additionally, the MPU6050, a critical component for motion detection, is stitched onto the back of the hand. This placement is strategic, allowing for comprehensive motion analysis. To further enhance the functionality and streamline the wearable design, the ESP32 is soldered onto a Printed Circuit Board (PCB). This step not only improves connectivity but also simplifies the integration of different components.

Figure 7. Jumper Wires

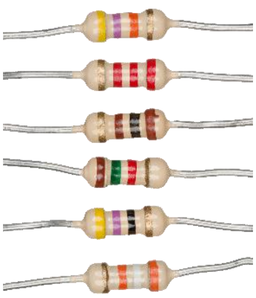


Figure 8. Resistors

The second key aspect of the glove's design involves the meticulous arrangement of jumper wires and power supply. Jumper wires are used to intricately connect various components of the circuit, effectively linking the sensors to the ESP32. This arrangement facilitates the smooth flow of electrical signals and establishes connections between different points in the circuit. Furthermore, the ESP32 is connected to a compact power bank, ensuring a stable power supply for the glove. This power source is critical for activating the glove and enabling the wireless transmission of sensor readings. The entire assembled unit is then securely affixed to a wristband. This careful configuration of the glove optimizes its functionality, making it a seamless and wearable solution for translating sign language.

3.1.2 Glove Design and Circuitry

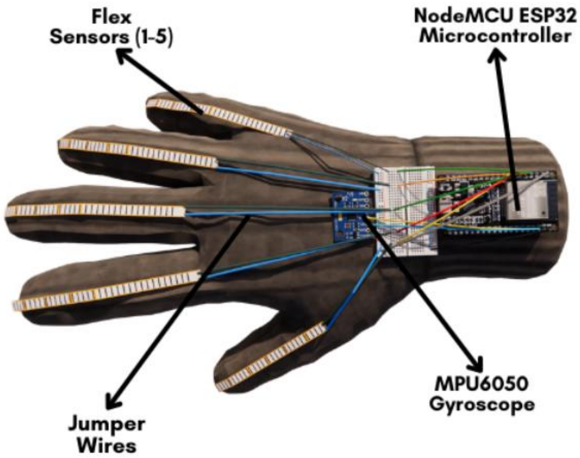


Figure 9. The Wearable Glove Design

The glove is meticulously engineered, with each component strategically stitched or attached to ensure optimal functionality and comfort. Flex sensors are placed on each finger to accurately track their movements, while the MPU6050 is positioned near the wrist, essential for reading motion values and orientation. Following this careful placement, each component is then connected to specific pins of the microcontroller, forming a cohesive and efficient circuitry system. The connections are as follows:

1. *Flex Sensors:*
   a. Pinky Finger (2.2 inches):
      - VCC: ESP32 3V3
      - Signal: ESP32 Pin 39 (VN)
      - Resistor: Signal to GND

   b. Ring Finger (4.5 inches):

- VCC: ESP32 3V3
- Signal: ESP32 Pin 34
- Resistor: Signal to GND

c. Middle Finger (4.5 inches):
- VCC: ESP32 3V3
- Signal: ESP32 Pin 35
- Resistor: Signal to GND

d. Index Finger (4.5 inches):
- VCC: ESP32 3V3
- Signal: ESP32 Pin 32
- Resistor: Signal to GND

e. Thumb Finger (2.2 inches):
- VCC: ESP32 3V3
- Signal: ESP32 Pin 33
- Resistor: Signal to GND

2. *MPU6050 Gyro:*
- VCC: ESP32 3V3
- GND: ESP32 GND
- SCL (Signal): ESP32 Pin 22
- SDA (Signal): ESP32 Pin 21

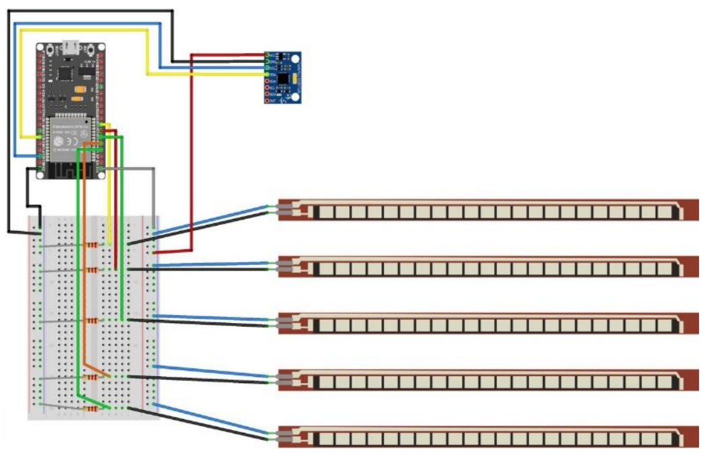


Figure 10. Circuit Design

3.2 System Architecture

At the core of this sophisticated system architecture is a specialized glove, outfitted with an array of sensors including flex sensors and a gyroscope. This wearable technology interfaces seamlessly with an Arduino microcontroller, functioning as an extension of the user's hand. Its dual role encompasses the thorough capture and storage of nuanced hand movements and gestures in a structured CSV format, while concurrently managing real-time data transmission to a Firebase database—a pivotal cloud-based repository that ensures synchronization with the user's gestures.

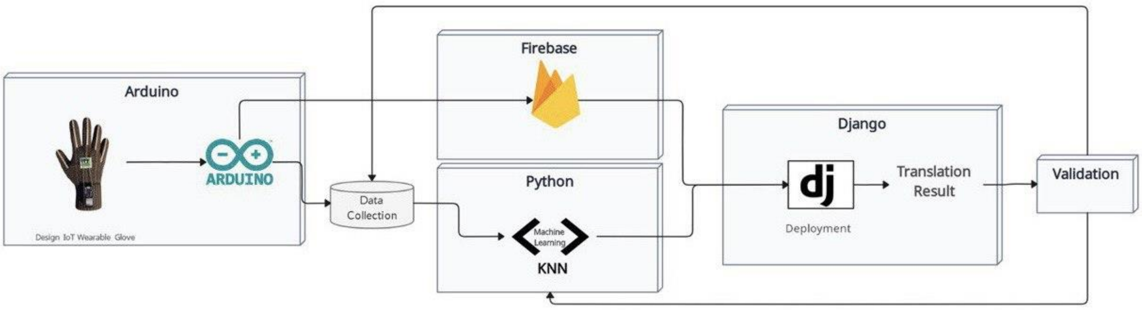


Figure 11. System Architecture

As shown in the system architecture illustration above, the Arduino setup, thus, serves as a bridge between the physical realm of gesture capture and the virtual storage infrastructure. The cloud-based Firebase database emerges as a crucial component, not merely for data storage, but for facilitating real-time updates, assuring the accuracy of the collected information to the user's expressions.

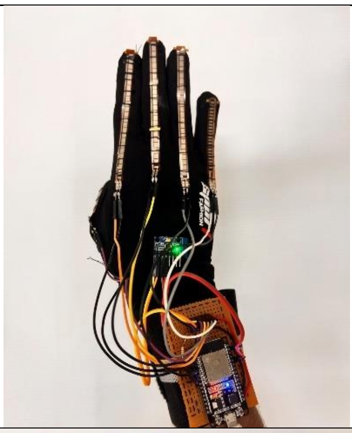
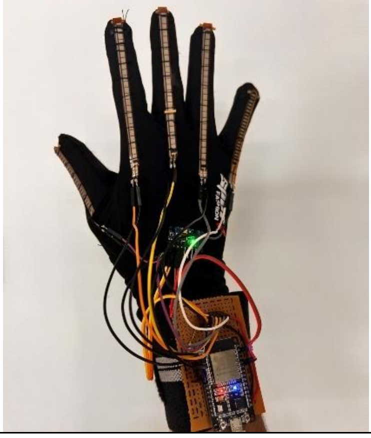
Enhancing the system's capabilities is the machine learning component—a sophisticated K-Nearest Neighbors (KNN) classifier. Trained on the detailed collected gesture data, this intelligent module transforms intricate hand movements into coherent sign language elements. This training process endows the model with the ability to interpret a spectrum of sign language gestures, laying the groundwork for seamless communication between users.

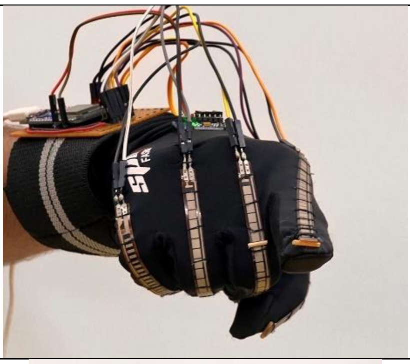
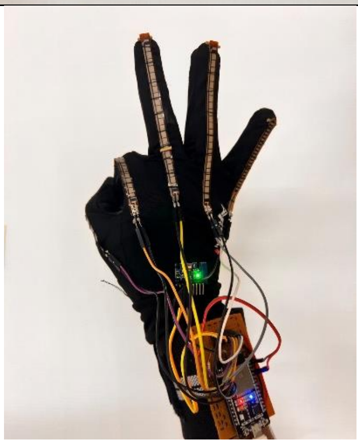
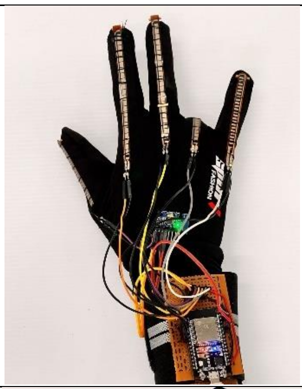
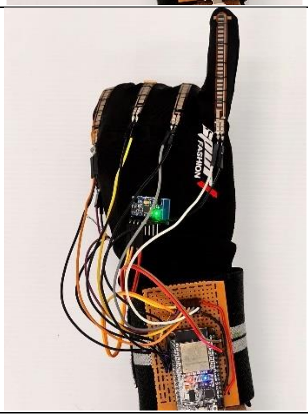
Upon successful training and validation for accuracy and reliability, the deployment phase takes center stage. The implementation is facilitated through the Django web framework—an adept platform that seamlessly integrates the trained model into the operational system. In this live state, the Django application establishes a continuous communication loop with the Firebase database, fetching the latest gesture data in real-time.

As new gesture data arrives, the application employs the pre-trained KNN model to interpret these movements, generating predictions that effortlessly translate gestures into written text words. This transformative process ensures that users, particularly those within the deaf and hard-of-hearing community, receive immediate and accurate sign language translations. The result is an elevated communication experience that empowers users to express themselves with clarity and precision. The strategic integration of Arduino with Firebase, followed by the adept utilization of Django for deploying the machine learning model, showcases a harmonious blend of hardware, cloud computing, and web technologies.

3.3 Data Collection

The project's data collection involved gathering inputs from 16 participants—comprising 8 males and 8 females. This intentional mix aimed to ensure a diverse range of hand sizes, encompassing small, medium, and large, to attain varied data. This strategic selection was crucial for perfecting the model, enabling the system to be effective for different individuals. The comprehensive dataset, totaling 3087 entries, was crafted meticulously, with each data set incorporating readings from flex sensors on the thumb, index, middle, ring, and pinky fingers, along with X and Y axis values from the gyroscope. This approach was meticulously designed to capture the intricate nuances of sign language, focusing specifically on 10 key American Sign Language (ASL) gestures: "Hello," "Goodbye," "You," "OK," "I Hate You," "I am/I," "Fine," "Yes," "No," and "I Love You."

Sign Language	**Label**
	Hello
	Goodbye

	You
	Ok
	I Hate You
	I/ I am

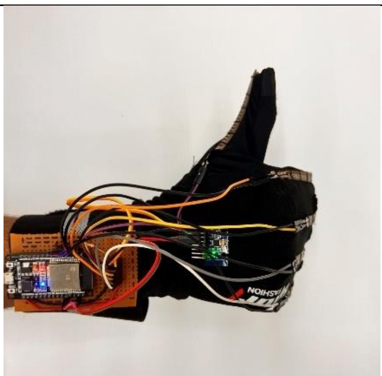
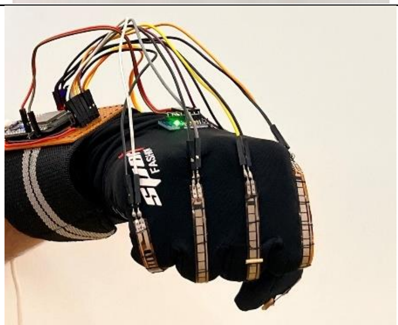
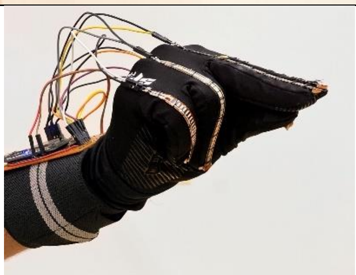
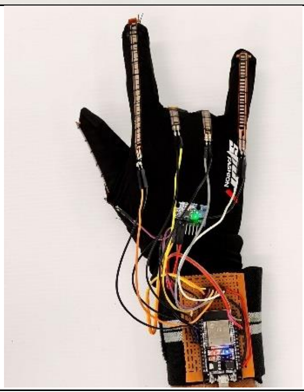
	Fine
	Yes
	No
	I Love You

Table 1. 10 ASL Gestures Used

Participants were instructed to perform these gestures while wearing the glove equipped with sensors, enabling real-time tracking of precise hand movements and orientations. The Arduino setup facilitated seamless recording and immediate storage of data in either CSV or table format, ensuring efficiency and organization in data handling.

Following the data collection phase, the dataset underwent thorough cleaning and preprocessing. This refining process reduced the total number of usable datasets to 2895, a crucial step in enhancing dataset quality. Standardizing the data was a key aspect, ensuring consistency across all measurements, especially for the selected ASL gestures. This meticulous preparation of the dataset laid the groundwork for the subsequent stage—building a machine learning model responsible for accurately interpreting and translating sign language gestures based on the collected data. The varied hand sizes represented in the dataset were pivotal for the model's effectiveness, allowing it to seamlessly translate gestures from individuals with different hand sizes and ensuring its applicability to a diverse user base.

3.4 The K-Nearest Neighbor Model

The K-Nearest Neighbors (KNN) algorithm stands as a central and pivotal component in the successful execution of this project, serving as a versatile and fundamental machine-learning technique specifically well-suited for classification tasks. Its fundamental principle involves classifying new data points by assessing the majority class among their closest neighbors in each feature space, as illustrated in Figure 4. This proximity-based classification relies on the calculation of a distance metric, often utilizing the Euclidean distance measure, which quantifies the dissimilarity or similarity between data points. A critical parameter in the KNN algorithm is 'k,' representing the number of neighboring data points considered during classification. The selection of an appropriate 'k' value is paramount, as it significantly influences the model's sensitivity to the local structure of the data.

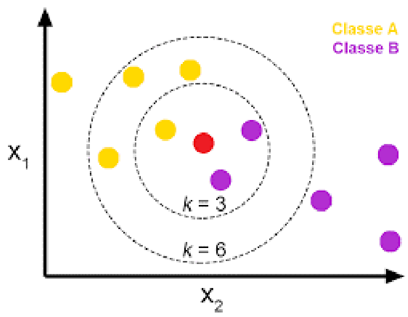


Figure 12. K-Nearest Neighbor Algorithm

In this project, the KNN algorithm assumed a central role as a foundational machine-learning method for classification tasks. It operated by classifying new data points based on the majority class among their nearest neighbors, relying on a distance metric, commonly the Euclidean distance. For the specifics of this project, the 'k' parameter, representing the number of neighbors considered, was set to 8. The dataset, comprised of hand gesture data collected from flex sensors and gyroscopes, underwent preprocessing steps. These steps included the removal of outliers using Z-scores and the standard division into training and test sets. Subsequently, the KNN classifier was trained on the prepared dataset, enabling it to accurately classify hand gestures based on sensor readings.

In the general context of KNN, the algorithm involves several steps:

1. *Receive Unclassified Data:*
   Begin by acquiring a set of unclassified data, representing a new observation that needs classification.
2. *Distance Measurement:*
   Evaluate the distance between the new data point and all other data points in the dataset, utilizing a chosen distance metric such as Euclidean, Manhattan, Minkowski, or Weighted distance.
3. *Selecting K Smallest Distances:*
   Choose the 'K' smallest distance from the calculated distances. Here, 'K' is a user-defined parameter representing the number of neighbors considered during classification.
4. *Class Enumeration:*
   Examine the list of classes associated with the 'K' smallest distances and tally the occurrences of each class.
5. *Majority Class Determination:*
   Identify the class that appears most frequently in the list as the majority class.
6. *Classification Decision:*
   Assign the majority class as the correct classification for the new data point. This ensures that the new data point is classified based on the class that predominates among its 'K' nearest neighbors.

3.5 SignWave: System Deployment

The deployment of the system centered on prioritizing real-time functionality and optimizing the user experience, resulting in the creation of the SignWave web application using Django—a Python-based web framework. This application was specifically designed to efficiently handle real-time data processing and user interactions. Leveraging Django's inherent strengths in managing database operations and user interfaces formed a robust foundation for seamlessly integrating complex functionalities. Furthermore, Django's compatibility with Python facilitated the smooth integration of the machine learning model, enabling efficient data processing and real-time analytics. At the core of the system's functionality lies the K-Nearest Neighbors (KNN) model, seamlessly embedded within the Django framework, allowing direct communication with Firebase—a cloud database renowned for its real-time data handling capabilities.

In practice, user interactions with the glove, equipped with sensors for gesture recognition, trigger instantaneous data transmission to Firebase. Subsequently, the Django application continuously retrieves this data, feeding it into the machine learning model for the real-time translation of gestures into text. The immediacy of this process is pivotal for the system's effectiveness, ensuring users experience minimal delay between their gestures and the corresponding translations. To enhance this functionality, the system's front end was meticulously crafted with HTML and CSS, prioritizing ease of use and visual appeal. This design decision not only increased user engagement but also guaranteed that the interface was accessible, contributing to an enriched overall user experience.

```python
#to initialize Firebase and retrieve data
def initialize_firebase_and_get_data():
    # Check if Firebase app is already initialized
    if not firebase_admin._apps:
        # Initialize Firebase with service account credentials
        cred = credentials.Certificate('signlanguage/credentials/esp32glovedataset-firebase-adminsdk-fevrw-f03886b
        firebase_admin.initialize_app(cred, {
            'databaseURL': 'https://esp32glovedataset-default-rtdb.asia-southeast1.firebasedatabase.app/'
        })
    return db.reference('test')

#Load KNN classifier
knn_classifier = joblib.load('signlanguage/knn_model/model.pkl')
```

Figure 13. Views.py Code to Initialize The Firebase

```python
def predict_view(request):
    #Initialize Firebase and retrieve data
    ref = initialize_firebase_and_get_data()
    #Fetch data from Firebase
    json_data = ref.get()

    #Check if data was successfully retrieved
    if json_data:
            # Remove the "Word" key from the JSON data
        if 'Label' in json_data:
            del json_data['Label']

        #Load the pre-trained scaler
        scaler = joblib.load('signlanguage/knn_model/trained_scaler.pkl')
        X = json_data

        # Convert the JSON data to a NumPy array
        X_array = np.array(list(X.values())).reshape(1, -1)
        X_scaled = scaler.transform(X_array)

        # Make predictions using the model
        prediction = knn_classifier.predict(X_scaled)
        # Pass the prediction to the html web
        context = {'prediction': prediction[0]}

        return render(request, 'prediction.html', context)
    else:
        return JsonResponse({'error': 'Failed to retrieve data from Firebase.'}, status=400)
```

Figure 14. View.py Code for Generating The Translation

The SignWave web application comprises four distinct pages. The Homepage serves as a welcome page and provides directions to other sections. The Translation Page is the central hub where the system displays translation results in real-time, fetched from Firebase data. The Tutorial Page offers insights into all recorded gestures in SignWave, serving as an educational resource. Finally, the Design Page showcases the physical glove design, explains the components, and features the circuit design, offering users additional information. Through the combination of Django's robust back-end capabilities and a carefully designed front-end, the system attains a harmonious balance between technical efficiency and user-centered design.

Figure 15. SIgnWave Welcome Page

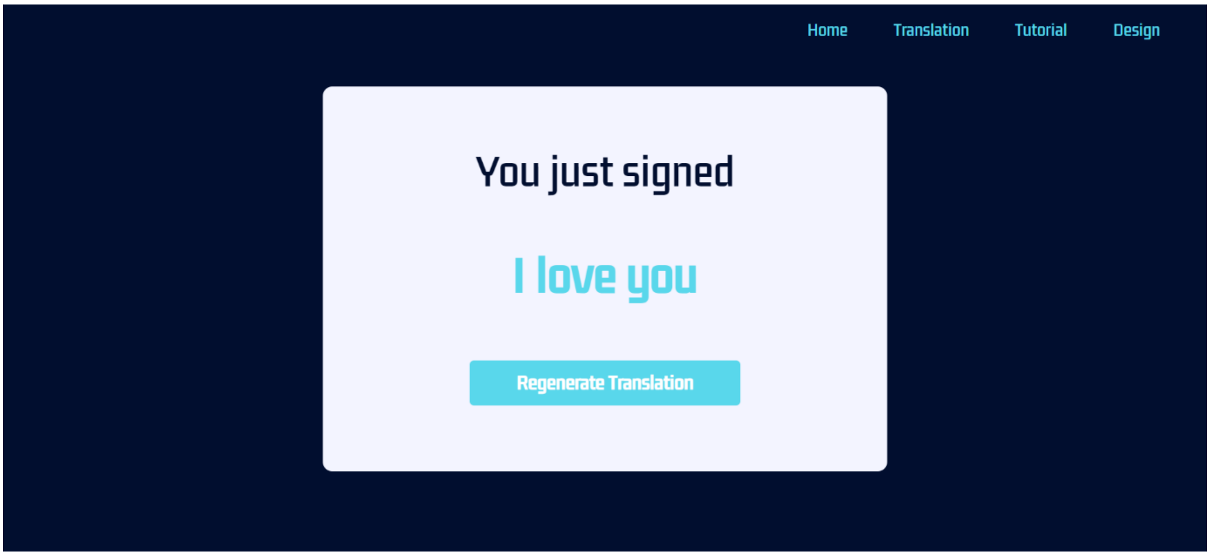


Figure 16. SignWave Translation Page

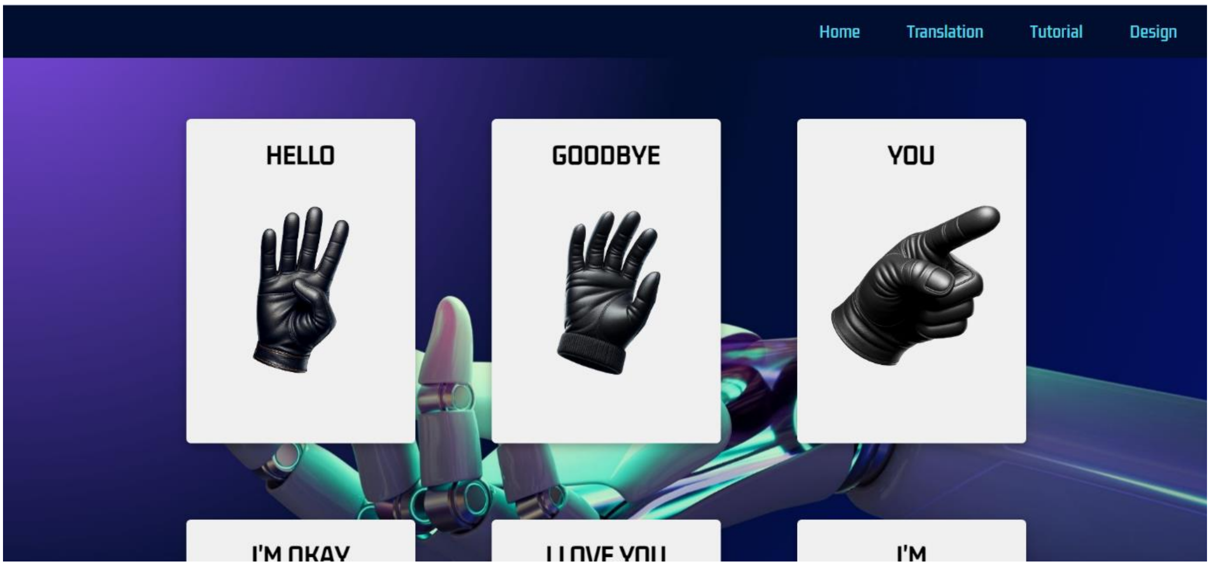


Figure 17. SignWave Tutorial Page

Figure 18. SignWave Design page

In conclusion, the SignWave system embodies a seamless fusion of advanced technological functionalities and user-friendly design, exemplifying a commitment to providing a transformative experience for users within the realm of assistive technology.

CHAPTER FOUR

ANALYSIS OF RESULTS AND OUTCOMES

4.1 System Evaluation

The wearable glove, incorporating flex sensors and a gyroscope, is a critical component in the system's operational framework. These strategically positioned sensors have accurately captured the bending of hand movements and gestures and contributed to the successful recording of user sign language gestures. However, achieving this functionality was not without its challenges. Establishing a reliable connection proved to be a considerable challenge, requiring careful adjustments. The jumper wires posed a persistent challenge as they needed to be reattached securely multiple times. Additionally, occasional inaccuracies in sensor readings and the need for periodic resets of the ESP32 microcontroller added complexity to maintaining a seamless connection.

Simultaneously, the physical appearance of the glove emphasizes an ergonomic design, prioritizing user-friendliness and comfort, especially for individuals with hearing impairments. Despite the challenges encountered, the physical glove has successfully demonstrated its functional prowess in capturing intricate gestures with precision.

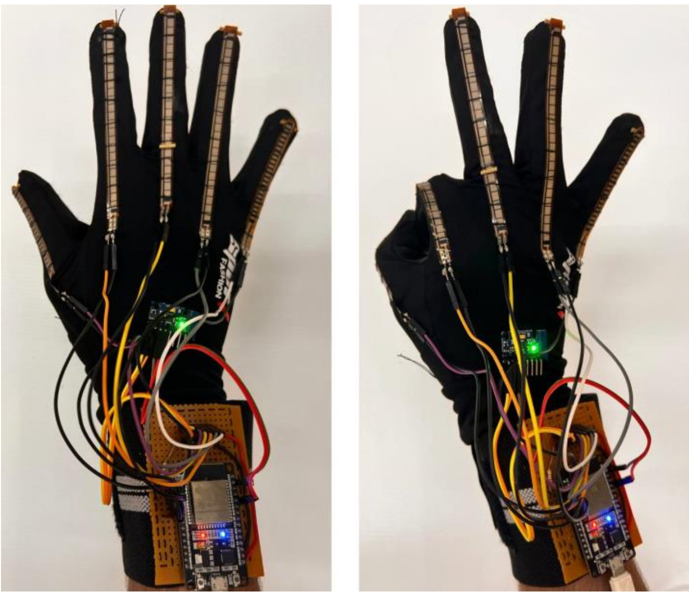


Figure 19. The Wearable Glove Being Used for Translation

On the other front, the web application interface serves as a gateway to real-time gesture translation. Thoughtfully designed for user interaction, the interface ensures an intuitive and accessible platform, prioritizing ease of use and visual appeal. Integrating a pre-trained machine learning model, the web application dynamically fetches sensor data from Firebase, generating instant translations presented to the user in real-time. This seamless interaction represents the culmination of successful web application design, translating complex data into user-friendly outputs.

Together, the wearable glove and the web application form a cohesive system. The challenges faced in establishing a reliable physical connection with the glove are counterbalanced by the system's overall success in capturing intricate gestures accurately. Simultaneously, the web interface smoothly translates this data into real-time user-friendly outputs, overcoming barriers for individuals with hearing impairments.

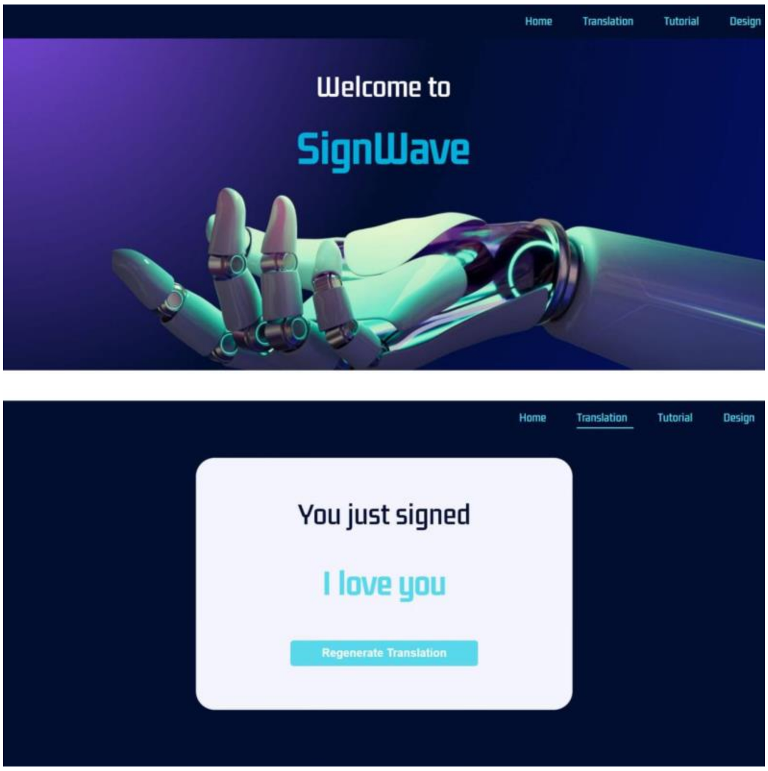


Figure 20. The Web Successfully Displays Translation

4.2 Model Evaluation and Performances

The classification report presented in Table 2 plays a pivotal role in the comprehensive evaluation of the K-Nearest Neighbors (KNN) model, especially in the context of a classification task involving distinct categories or gestures. This meticulously structured report provides essential metrics such as Precision, Recall, and F1-score, accompanied by the Support for each category. These metrics are critical for gaining insights into the model's performance across individual classes. Specifically, in the classification report for the KNN model, the 'Hello' and 'Goodbye' gestures exhibit lower precision (0.87 and 0.91, respectively) and recall

(0.89 and 0.84, respectively). This discrepancy is likely attributed to their physical resemblance, resulting in frequent misclassifications, and underscoring the challenge of distinguishing visually or contextually similar gestures.

	Precision	**Recall**	**F1-score**	**Support**
Hello	0.87	0.89	0.93	55
Goodbye	0.91	0.84	0.88	51
You	0.93	0.98	0.96	56
Ok	0.96	0.96	0.96	50
I Hate You	0.98	0.89	0.88	51
I am	0.92	0.92	0.92	60
Fine	0.91	0.94	0.92	32
Yes	0.93	0.88	0.90	65
No	0.85	0.98	0.91	45
I Love You	0.93	0.95	0.94	56
Model Accuracy			0.92	516

Table 2. Model Performance Evaluation

On closer examination, the 'No' gesture stands out as an interesting case, displaying a high recall of 0.98. This suggests that the model effectively captures a substantial proportion of actual 'No' instances. However, the lower precision of 0.85 indicates a tendency to over-identify, possibly influenced by the dynamic nature of the gesture involving motion. This intricate balance between precision and recall highlights the need for careful consideration when handling gestures with inherent motion characteristics.

Similarly, the 'I Hate You' and 'I Love You' gestures, despite conveying opposing meanings, exhibit a nuanced pattern in their precision and recall metrics. Both gestures showcase high precision (0.98 and 0.93, respectively), indicating that when the model predicts these gestures, it is often correct. However, the corresponding recall metrics (0.89 and 0.95) suggest occasional instances of missed identifications. This ambiguity in predictions underscores the importance of meticulous training in gesture recognition models, particularly when confronted with gestures that share subtle physical similarities or differences. Fine-tuning the model to discern these nuances and augmenting the dataset with diverse examples can further refine its ability to accurately interpret and translate gestures with precision.

In summary, the table implies that while the overall model performance is commendable (as indicated by the overall accuracy of 0.92), there are specific instances where gesture similarity impacts precision and recall. This underscores the importance of training the model with a diverse set of examples for each gesture and potentially fine-tuning it to enhance its ability to distinguish between gestures that share similarities in appearance or context.

4.3 Translation Validation

The extensive validation of the translation system's performance, involving a diverse group of 10 individuals comprising both males and females, unveils its robust functionality in real-world scenarios. The variation in hand sizes, denoted as large ('3'), medium ('2'), and small ('1'), is a pivotal aspect influencing the system's operational success. This diversity plays a crucial role as it significantly impacts the flex sensor values in the glove, introducing variations in the degree of bending. The intricate relationship between hand size and flex sensor readings is pivotal, directly influencing the system's accuracy in interpreting gestures with precision during real-time translation scenarios.

Person	A	B	C	D	E	F	G	H	I	J	**Accuracy**
Hand Size	3	3	3	3	2	1	1	1	1	2	
Hello	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100%
Goodbye	✓	✓	✓	✓	✓	✓	✗	✗	✓	✓	80%
You	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100%
Ok	✓	✓	✓	✓	✓		✓	✓	✓	✓	100%
I Hate You	✗	✓	✓	✗	✓	✓	✗	✓	✓	✗	60%
I am/ I	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100%
Fine	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100%
Yes	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100%
No	✓	✓	✓	✓	✓	✗	✓	✓	✗	✗	70%
I Love You	✓	✓	✓	✓	✗	✓	✓	✓	✓	✓	90%

Table 3. Translation Validation

Table 3 provides an insightful breakdown of the system's performance across different hand sizes, highlighting its adaptability and accuracy in the face of varying physical characteristics among users. While the system adeptly addresses the challenges introduced by diverse hand sizes, it's noteworthy that smaller hand sizes exhibit slightly lower accuracy. This discrepancy may be attributed to the smaller fingers in individuals with small hand sizes, limiting their ability to reach the maximum length of the sensor. As a result, the flex sensor values may not reach the optimal range, impacting the system's overall accuracy in interpreting gestures.

Moreover, as the analysis delves into gestures with subtle differences, such as the phrases "I hate you" and "I love you," the system faces unique challenges, exhibiting a lower accuracy rate of 60% for "I hate you." This underscores the importance of meticulous training to distinguish between closely resembling gestures. Additionally, the gesture for "No" experiences a reduction in accuracy (70%), primarily due to its visual resemblance to the gesture for "You." These complexities emphasize the need for a targeted data collection approach, focusing on diverse examples, especially those with close resemblances. This iterative process ensures the continual enhancement of the system's accuracy and reliability, addressing challenges posed by various hand sizes and sign language intricacies.

CHAPTER FIVE

CONCLUSION AND FUTURE WORK

In conclusion, the development of the wearable glove for sign language translation using machine learning (SignWave) represents a significant breakthrough in addressing communication barriers faced by the deaf and hard-of hearing community. The system's remarkable accuracy rate of 92.05% in translating sign language gestures into text underscores its effectiveness and potential to revolutionize the way individuals with hearing impairments interact with the world. Through the integration of flex sensors, gyroscope technology, and the K-Nearest Neighbors (KNN) machine learning model, this innovative solution offers a practical and efficient means of bridging the communication gap between the deaf community and those unfamiliar with sign language. Notably, the system's real-world performance evaluations, involving a diverse group of participants, have demonstrated its capability to accurately interpret a range of sign language gestures. While challenges exist in distinguishing between closely resembling gestures, particularly "I hate you" and "I love you," the system's overall performance is auspicious. With further data collection and training, the system is expected to continue to improve in recognizing subtleties and nuances in sign language. In essence, this project signifies a significant step towards inclusivity and accessibility, offering a potential lifeline to over 5% of the global population with disabling hearing loss and paving the way for a more inclusive future where communication barriers are a thing of the past.

Upgrading the SignWave web application to enable user-contributed sign language gestures is a critical enhancement. This improvement aims to build a larger and more varied database, significantly boosting the system's accuracy and scope. By embracing a broader spectrum of sign language expressions, the system will become more inclusive and adaptable to diverse user needs. The personalization of the system is a major focus for future development. This feature will enable users to have their real-time database, which will be particularly useful for capturing and reading sensor values from the wearable glove. This level of customization not only enhances the accuracy of translations but also ensures greater usability and convenience in various environments. Another area of future work involves improving the system’s adaptability to different hand sizes and shapes. This enhancement will focus on refining the glove’s sensor sensitivity and the model’s ability to interpret a wider range of hand movements. By doing so, the system will become more flexible and accommodating to individual physical differences, which is crucial for accurately translating sign language gestures for a diverse user base. This improvement will ensure that the system remains effective and reliable, regardless of variations in users' hand characteristics. The final aspect of future enhancements involves upgrading the database architecture to support more complex functionalities. A robust and advanced database is critical for efficiently managing larger data volumes and enabling the integration of sophisticated machine-learning algorithms. This improvement is vital for scaling the system, enhancing its performance, and expanding its capabilities to meet future technological demands and broader application areas.

REFERENCES

Ahmed, M. A., Zaidan, B. B., Zaidan, A. A., Salih, M. M., & Lakulu, M. M. B. (2018). A Review on Systems-Based Sensory Gloves for Sign Language Recognition State of the Art between 2007 and 2017. Sensors (Basel, Switzerland), 18(7), 2208. https://doi.org/10.3390/s18072208

Choudhary, D. K., Singh, R., & Kamthania, D. (2021, April). Sign language recognition system. In Proceedings of the International Conference on Innovative Computing & Communication (ICICC).

Côté-Allard, U., Fall, C. L., Campeau-Lecours, A., Gosselin, C., Laviolette, F., & Gosselin, B. (2017, October). Transfer learning for sEMG hand gestures recognition using convolutional neural networks. In 2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC) (pp. 1663-1668). IEEE.

Eshetu, Y., & Wolde, E. (2019). A Real-Time Ethiopian Sign Language to Audio Converter. INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH & TECHNOLOGY (IJERT), 8(08).

Fu, Y. F., & Ho, C. S. (2008). Development of a programmable digital glove. *Smart materials and structures*, *17*(2), 025031.

Heera, S. Y., Murthy, M. K., Sravanti, V. S., & Salvi, S. (2017, February). Talking hands—An Indian sign language to speech translating gloves. In *2017 International conference on innovative mechanisms for industry applications (ICIMIA)* (pp. 746-751). IEEE.

I. José, "KNN (K-Nearest Neighbors) #1," Medium, Jun. 26, 2019. https://towardsdatascience.com/knn-k-nearest-neighbors-1-a4707b24bd1d

Li, G., Wu, H., Jiang, G., Xu, S., & Liu, H. (2018). Dynamic gesture recognition in the Internet of Things. IEEE Access, 7, 23713-23724.

Paul Sabu. (2020). Smart hand Glove. GitHub. https://github.com/pscretn/smart-hand-glove

Rajalingam, B., Kumar, R. S., Deepan, P., Patra, P. S. K., & Bavankumar, S. (2022, December). A Smart System for Sign Language Recognition using Machine Learning Models. In 2022 4th International Conference on Advances in Computing, Communication Control and Networking (ICAC3N) (pp. 1125-1131). IEEE.

Rosero-Montalvo, P. D., Godoy-Trujillo, P. E., Flores-Bosmediano, E., Carrascal-García, J., Otero-Potosi, S., Benitez-Pereira, H., & Peluffo-Ordóñez, D. H. (2018). Sign Language Recognition Based on Intelligent Glove Using Machine Learning Techniques. 2018 IEEE Third Ecuador Technical Chapters Meeting (ETCM), 1-5.

Suri, A., Singh, S. K., Sharma, R., Sharma, P., Garg, N., & Upadhyaya, R. (2020, September). Development of sign language using flex sensors. In 2020 international conference on Smart Electronics and communication (ICOSEC) (pp. 102-106). IEEE.

World Health Organization, “Deafness and hearing loss,” WHO, 2023. https://www.who.int/news-room/fact-sheets/detail/deafness-and-hearing-loss

Yudhana, A., Rahmawan, J., & Negara, C. U. P. (2018, September). Flex sensors and MPU6050 sensors responses on the smart glove for sign language translation. In IOP conference series: materials science and engineering (Vol. 403, No. 1, p. 012032). IOP Publishing.

APPENDICES

A. GANTT CHART

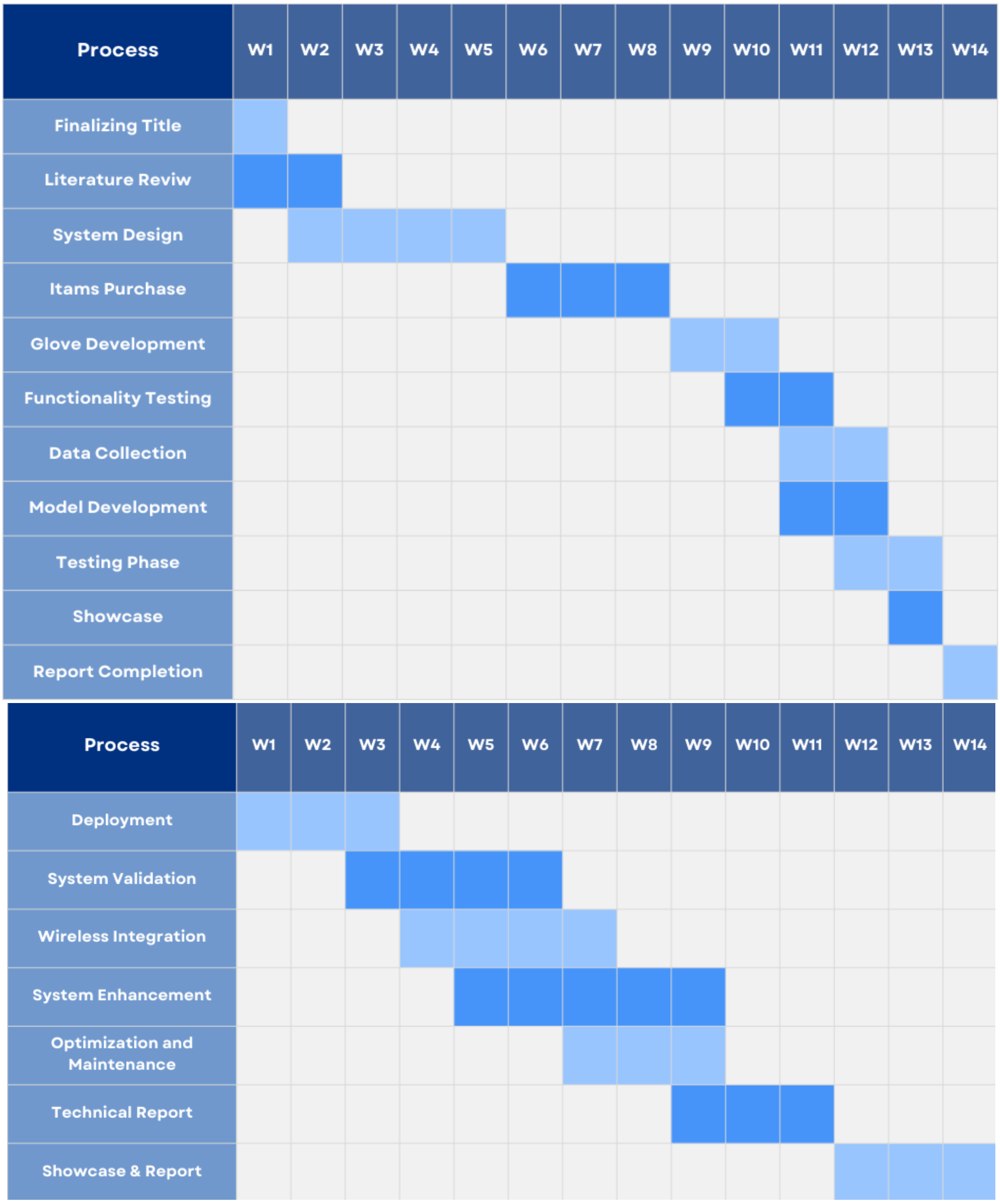


Figure 21. Gantt Chart SignWave System Development

B. INTERFACE

The interfaces are in Chapter 3 as part of the Deployment.

C. PRODUCT VIDEO

https://youtu.be/LsWQlNrGWuk

D. PROJECT LOGBOOK