INTERNATIONAL JOURNAL OF ADVANCE RESEARCH, IDEAS AND INNOVATIONS IN TECHNOLOGY

IJARIIT

ISSN: 2454-132X

Impact Factor: 6.078

(Volume 10, Issue 1 - V10I1-1298)

Available online at: https://www.ijariit.com

Smart glove for Malayalam sign language recognition and audio output

Muralikrishna K. S.
the13mk@gmail.com
Mar Athanasius College of Engineering, Kothamangalam, Kerala

Nikhitha Prakasan
nikhithaprakash2020@gmail.com
Mar Athanasius College of Engineering, Kothamangalam, Kerala

Haritha T. K.
harithatk521@gmail.com
Mar Athanasius College of Engineering, Kothamangalam, Kerala

Asha J. George
ashajgeorge@mace.ac.in
Mar Athanasius College of Engineering, Kothamangalam, Kerala

Rini T. Paul
rinitpaul@mace.ac.in
Mar Athanasius College of Engineering, Kothamangalam, Kerala

ABSTRACT

Sign language, which is a medium of communication for deaf and mute people, uses manual communication and body language to convey meaning, as opposed to using sound. In this system, we are proposing a hand wearable device by which the deaf and mute people can communicate with normal people through the glove. The National Institute of Speech and Hearing (NISH) recently introduced Signs for Malayalam Alphabets. The Malayalam Sign Language alphabet is distinguished by this wearable system combining five flex sensors, a three -axis MPU sensor, Arduino nano microcontroller and a WiFi--BT--BLE MCU module. The glove tracks hand and finger movements through sensors, sending data via Wi--Fi to a mobile app which converts it into text and audio. The RTOS enables concurrent task management with deterministic timing, ensuring efficient operation of multiple functions like sensor data acquisition, gesture interpretation, and communication. The system emphasizes dataset creation for this newly formed sign language, training a model using Support Vector Machine (SVM) to recognize these signs, and integrating the model into a flutter application for ease of access.

Keywords: *Malayalam Sign Language (MSL), Dataset, Support Vector Machine (SVM), Machine learning, Real time operating system, Sign language detection*

1. INTRODUCTION

Effective communication is crucial for a fulfilling life. However, for those who are deaf and mute, sign language is the only way to communicate. Sadly, many of them struggle with communication in their daily lives due to a lack of exposure to different sign languages used in different regions. According to the World Health Organization (WHO), there are over 460 million people globally with hearing disabilities, making up about 5 percentage of the world's population. In India alone, about 12.3 million people fall under this category. Engaging in conversation with this specially -abled population can be a challenge, as not everyone is familiar with sign language. In order to facilitate effective communication for people with disabilities, the Rights of Persons with Disabilities Act mandates the availability of trained Indian Sign Language interpreters in all government and public sector offices. While manual sign language translators are currently the most commonly used solution, this method can compromise the privacy of conversations. This poses a significant challenge for the estimated 1.8 to 7 million people in India who are deaf and mute, especially given that there are only 250 certified Sign Language Interpreters available in the country. This number is constantly decreasing due to reluctance or inadequate qualifications to pursue this profession. Just like any other sign language, the sign language used in Kerala incorporates facial expressions alongside hand gestures. However, this can differ in various regions of Kerala, highlighting the need for a standardized form of sign language specifically for the state. A major obstacle faced by teachers in schools for the hearing impaired is the lack of a sign language alphabet in Malayalam. While English and Hindi have their own established sign languages, schools currently rely primarily on lip movements to communicate. In situations where written words are necessary, they are either written onto the students' hands or traced in the air. Unfortunately, this method often leads to confusion.

The National Institute of Speech and Hearing (NISH) unveiled a groundbreaking development - uniform sign language characters in the Malayalam alphabet created through finger-spelling (Fig 1, Fig 2). Under the guidance of sign language experts, teachers and students at NISH] collaborated to produce this unique form of communication that encompasses both vowels and consonants. With this new tool, lip movement will no longer pose a barrier for teachers trying to convey messages to their hearing- impaired students. This breakthrough has the potential to significantly enhance the quality of life for those with hearing impairments. Through our work, we have developed a cutting- edge smart love equipped with advanced sensor technology, aimed at enhancing communication for deaf and mute individuals through sign language. The glove incorporates five flex- sensors and a three -axis MPU sensor, enabling precise detection of hand rotations and finger movements that correspond to sign language gestures. This wealth of sensor data is then transmitted via Wi-Fi to a mobile application, where it undergoes sophisticated machine learning processing. This sophisticated algorithm generates both textual and audio outputs, effectively b ridging the communication gap. Since this technology is specifically tailored for Malayalam Sign Language Alphabets, we saw the need to create a comprehensive dataset for accurate translation, providing a valuable contribution to the field.

2. PROPOSED DESIGN

The National Institute of Speech and Hearing (NISH) made a significant achievement by aligning standardized sign language characters with the Malayalam alphabet using finger spelling (fig.1, fig.2) Inspired by this progress, we were motivated to create a communication system specifically for individuals who have hearing and speech impairments. With the integration of flex sensors, an IMU, an Arduino Nano board, an ESP microcontroller, and RTOS into a glove, we are able to capture hand and finger movements. These movements are then transmitted via Wi-Fi to a Flutter app, where they are processed using SVM algorithms to interpret sign language gestures into spoken words. This innovative approach greatly enhances accessibility for the deaf and mute community, in line with NISH's efforts to standardize sign language characters.

2.1 FreeRTOS

FreeRTOS acts as the foundation of the glove system, allowing it to seamlessly handle multiple tasks simultaneously and with reliable timing. This crucial functionality guarantees accurate synchronization of operations essential for real- time data collection and processing. Due to FreeRTOS, the glove excels in performing various functions such as acquiring sensor data, interpreting gestures, and seamlessly communicating with the mobile application, resulting in improved efficiency and reliability.

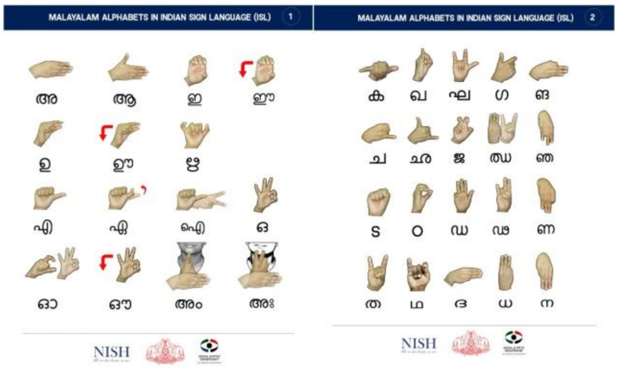


Figure 1. Malayalam alphabets in Indian sign language

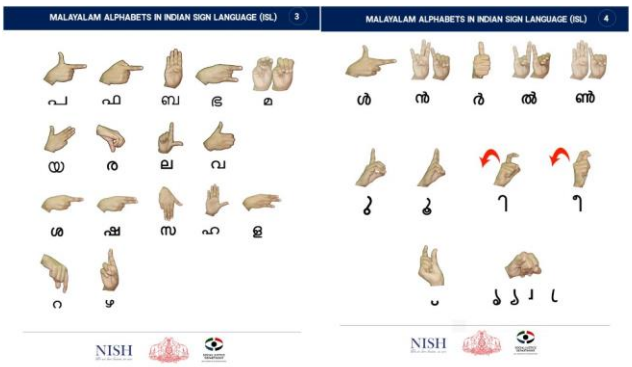


Figure 2. Malayalam alphabets in Indian sign language

2.2 Flex Sensors Integration

The glove design incorporates flex sensors to accurately capture the nuances of finger movements in Malayalam Sign Language (MSL). These sensors are strategically placed on the glove to correspond with specific hand gestures. They work by detecting changes in resistance as the fingers bend, generating analog voltage signals. The Arduino Nano's analog input pins (A0, A1, A2, A3, and A6) sample these signals. To ensure signal quality, each flex sensor is paired with a 10k resistor, reducing noise and ensuring consistent readings. The Arduino Nano, using the FreeRTOS library, processes these analog signals and transmits them to the ESP32 microcontroller via UART protocol. The ESP32 then accurately interprets these signals in real-time, enabling seamless communication through MSL gestures.

2.3 MPU9250 Integration

One essential component of the glove is the MPU9250 sensor, which significantly enhances the gesture recognition process by providing precise orientation and motion data. With an accelerometer, gyroscope, and magnetometer, this sensor offers an impressive nine degrees of freedom (DOF) for capturing three- dimensional (3D) hand movements. The accelerometer detects acceleration along three axes, while the gyroscope tracks angular velocity and the magnetometer detects the magnetic field. By combining the information from these sensors, the glove can accurately measure roll, pitch, and yaw (head) angles. This sophisticated integration, using an Attitude and Heading Reference System (AHRS) algorithm, enables the glove to interpret hand and finger gestures with remarkable precision.

2.4 Arduino Nano Integration

The Arduino Nano board is used to enhance the capabilities of the system. It helps integrate the FreeRTOS library, which provides real-time operating system features. The Nano acts as a middleman, connecting the flex sensors, IMU (Inertial Measurement Unit), and the ESP32 microcontroller. The flex sensors are connected to the Nano's analog input pins (A0, A1, A2, A3, A6), and the IMU is connected to pins A4 and A5. The Nano processes the analog values generated by these sensors. Through the I2C protocol, the Nano communicates with the IMU to read the nine -axis values, providing detailed motion data. Finally, the Arduino Nano transmits both the flex sensor and IMU values to the ESP32 microcontroller using the UART protocol for serial communication. The communication is vital as the Nano acts as a bridge between the sensors and the ESP32. The ESP32 lacks the ability to precisely detect analog values from the flex sensors. By creating a serial connection between the Nano and ESP32, with the Rx (receiver) and Tx (transmitter) pins properly connected, smooth data transmission is achieved. This allows the ESP32 to effectively process and interpret the sensor data. The UART (Universal Asynchronous Receiver -Transmitter) protocol facilitates serial communication between two devices, typically microcontrollers or sensors, enabling them to exchange data bit by bit. It operates asynchronously, meaning that there is no shared clock signal between the devices. Instead, data is transmitted in packets, with each packet containing a start bit, data bits, optional parity bits for error detection, and stop bits to signal the end of transmission. UART communication is commonly used in embedded systems due to its simplicity and versatility. In the context of the described system, UART enables the Arduino Nano to transmit data from the flex sensors and IMU to the ESP32 microcontroller, allowing for seamless integration and accurate transfer of sensor information despite the ESP32's limited analog input capabilities.

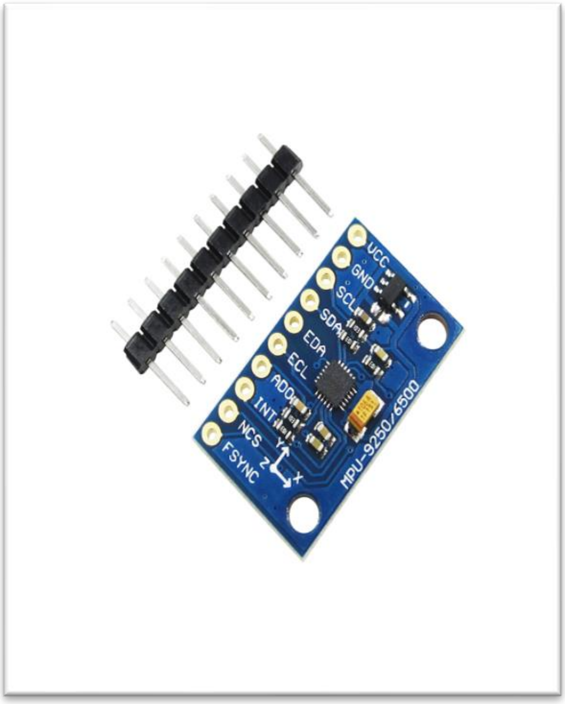


Figure 3. MPU-9250 Sensor

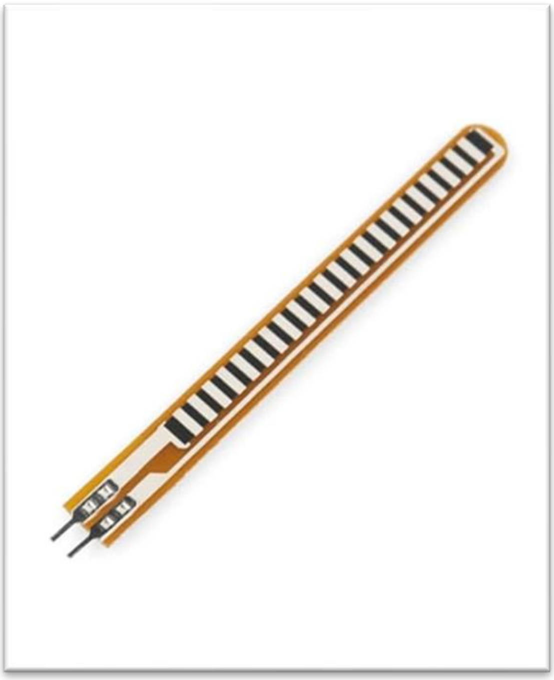


Figure 4. Flex Sensor

2.5 ESP32 MPU Data Transmission to Firebase

The ESP32 microcontroller uses Wi--Fi to connect wirelessly with the application. The Arduino Nano collects data from the flex sensor and IMU, and sends it to the ESP32 via the UART protocol. The ESP32 then sends this combined data to Google Firebase to create a dataset. The data is transmitted in the widely- used JSON format, which helps organize and convey the information efficiently. By using Google Firebase's features, the system enhances gesture recognition and communication within the application, improving the user experience and enabling accurate and reliable interpretation of gestures.

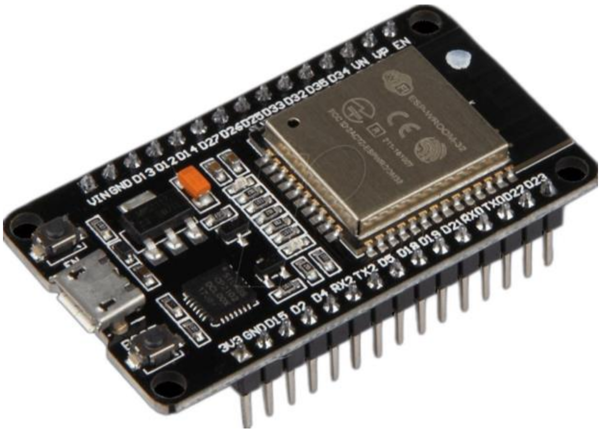


Figure 5. ESP-32 Microcontroller

Figure 6. Aurdino nano

2.6 Support Vector Machine (SVM) Model Training and Inference

The SVM model is an incredibly useful supervised learning algorithm, specifically designed for gesture recognition within the glove system. Through the training process, this sophisticated model is able to effectively classify and identify a vast array of Malayalam Sign Language (MSL) gestures, utilizing the valuable sensor data acquired during the demonstration of said gestures. Each sample within the training dataset contains labeled sensor data, accurately representing a specific MSL gesture. Sub section Training Process.

1) Training Process:

- Data Collection: We use flex sensors and an MPU9250 sensor to capture the intricacies of hand and finger movements during gesture demonstrations.
- Feature Extraction: This includes measuring the bending angles ofthe fingers using the flex sensors and calculating orientation angles (roll, pitch, yaw) using the AHRS algorithm from the MPU9250 sensor.
- Labeling: we label each sample in our dataset with the corresponding MSL gesture. This provides the necessary ground truth for training our SVM model.
- Training: The SVM model is honed through exposure to a labeled dataset. It uses this data to uncover the most effective hyperplane for distinguishing between various gesture types within the feature space. This model is driven to achieve its ultimate goal of maximizing the distance between different classes while minimizing any classification mistakes.

2.7 Attitude and Heading Reference System (AHRS)

The use of Madgwick's IMU and AHRS algorithms successfully employs the AHRS algorithm to precisely gauge the orientation and motion data from the accelerometer, gyroscope, and magnetometer of the MPU9250 sensor. This dynamic algorithm accurately computes roll, pitch, and yaw angles, offering a multi-faceted view of hand and finger motions in a three- dimensional realm

1) Data Fusion:

- Sensor Data Acquisition: The MPU9250 sensor is a powerful tool for collecting raw data, providing detailed readings of acceleration, angular velocity, and magnetic fields through its accelerometer, gyroscope, and magnetometer. With this data acquisition, precise measurements can be obtained for further analysis.
- Sensor Fusion: Through the use of Madgwick's algorithm, Sensor Fusion integrates data from various sensors to determine the orientation of the glove system in three dimensional space. This innovative approach takes into account biases and noise from each sensor, resulting in highly precise and steady orientation readings.

2) Orientation Calculation:

- Attitude Estimation: Accurately determining the attitude of a glove system in three- dimensional space is made possible with AHRS. By calculating the relative orientation of the system to a reference frame, it provides essential roll, pitch, and yaw angles.
- Heading Estimation: Improving Directional Orientation: Thanks to the incorporation of magnetometer readings, AHRS can now estimate the magnetic heading of the glove system, resulting in precise determination of its directional orientation.

The glove system of Malayalam Sign Language gestures leverages the combination of AHRS for orientation estimation and Support Vector Machine (SVM) for gesture recognition. This integration yields a highly effective and precise interpretation of gestures, greatly benefitting the deaf and mute community by enabling seamless and real- time communication.

2.8 Integration of Firebase and Google Colab in the Gesture Translation System

The Gesture Translation System relies on Firebase as its backbone, ensuring smooth management and storage of sensor data in real -time. Using a Wi--Fi connection, the ESP32 microcontroller wirelessly transmits these sensor values to Firebase. Once received, Firebase swiftly creates a cloud based real- time database, guaranteeing instant access and synchronization across all linked devices. Firebase caters to various needs of the system, offering essential services.

Authentication services ensure secure access control to the application, safeguarding user privacy. Meanwhile, storage services efficiently handle the storage of sensor data, allowing seamless retrieval and manipulation whenever needed. Firebase isn't working alone. It partners up with Google Colab, a collaborative platform where we train our gesture recognition models. Using machine learning techniques like Support Vector Machines (SVM), analyze the sensor data stored in Firebase to train and refine our models. Colab's interactive environment makes the process smooth and iterative, ensuring our models are accurate and up-to-date with real-world data. By bringing together Firebase and Google Colab, The Gesture Translation System gets a significant boost in functionality and performance. Firebase keeps our data synced and managed in real-time, while Colab empowers us to build powerful gesture recognition models. The seamless integration means the system is not just accurate and reliable but also userfriendly, ensuring a smooth and satisfying experience for all users.

2.9 TensorFlow Lite for Gesture Translation

By leveraging the power o fTensor Flow Lite, the application can seamlessly transform MSL gestures into written and audible outputs. Utilizing trained SVM models within TensorFlow Lite enables the swift and accurate detection of gestures in real- time. The application then further processes the output from the SVM model, generating both text and computerized audio descriptions of the gestures. This seamless integration of Tensocommunication system accessibility, but also caters to the diverse needs of its users.

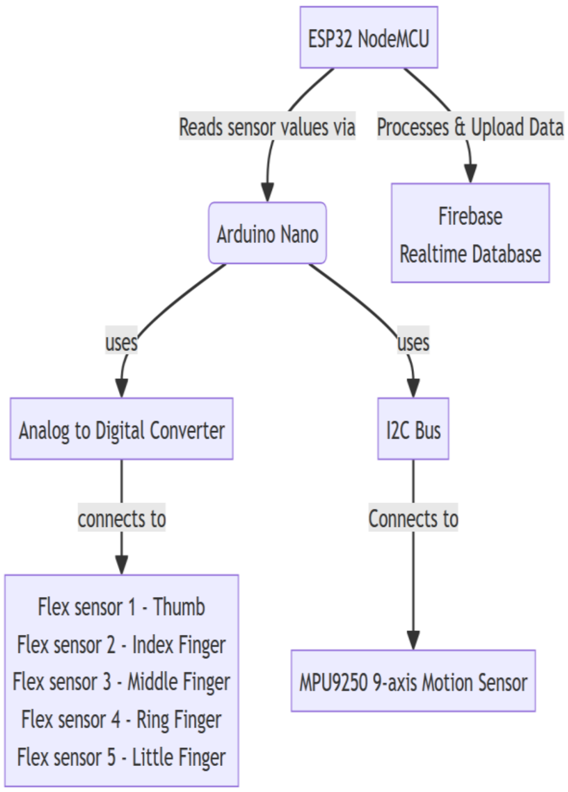


Figure 7. Data module

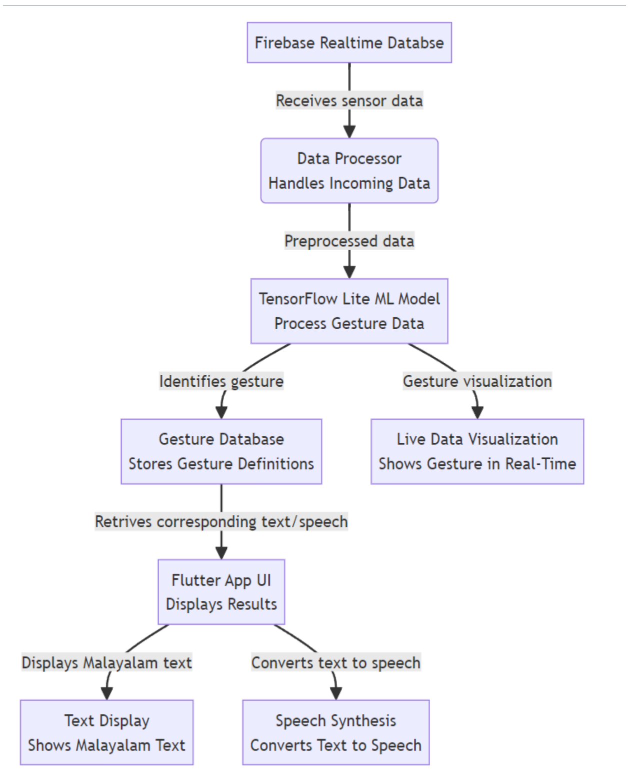


Figure 8. Application module

2.10 Gesture Recognition Workflow in the Flutter Application

After training the gesture recognition models using Google Colab, they are integrated into the Flutter application. When a user performs a gesture, the ESP32 captures the sensor data and wirelessly transmits it to the Flutter app. Inside the app, the received sensor values are processed and fed into the deployed gesture recognition models. These models work in real-time, analyzing the sensor data to predict the corresponding sign or symbol based on the patterns and classifications learned during the training phase. Once the model identifies a gesture, the Flutter app interprets the prediction and links it to the corresponding text and audio output. This allows for instant feedback the recognized gesture is displayed as text while generating audio to improve accessibility. In summary, the trained gesture recognition models within the Flutter app analyze incoming sensor data, enabling accurate identification of performed gestures. This process enables the app to provide immediate and precise text and audio feedback in real -time.

3. EXPERIMENT ANALYSIS AND RESULT

3.1. Dataset Creation and Training

The dataset used in this study consisted 44,000 images, sourced from Firebase. Each image represented one of the 44 Malayalam vowels, with 1,000 images allocated per vowel. This ensured a comprehensive and balanced dataset across all vowel categories. To facilitate model development and evaluation, the dataset was split into 80 percentage for training and

20 percentage for testing. Training of the Support Vector Machine (SVM) model was conducted on Google Colab, leveraging its collaborative environment for machine learning tasks.

3.2 Model Training

A Support Vector Machine (SVM) classifier was chosen for its suitability in handling multidimensional input data and binary classification tasks. Feature extraction techniques, such as capturing finger bending angles from flex sensors and orientation data from the IMU, were employed to convert raw sensor data into feature vectors suitable for SVM training. Hyperparameter tuning was performed to optimize the SVM model's performance, including kernel selection, regularization parameters, and margin optimization.

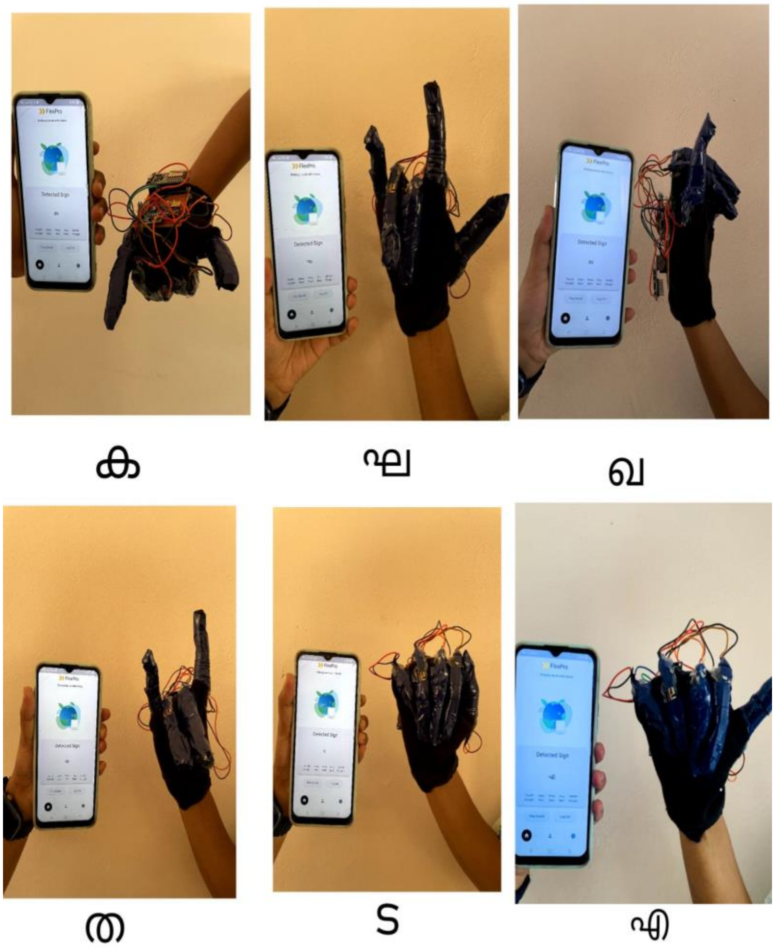


Figure 9. Sample output of detected sign languages

3.3 Performance Evaluation Metrics

The main metrics used to assess the performance of the SVM model were precision, recall, F1score, and accuracy. Precision measured how accurate the positive predictions were, recall looked at the system's ability to correctly identify positive cases, and the F1score balanced precision and recall. Additionally, the overall accuracy of the model on the test set provided a comprehensive gauge of its performance.

3.4 Result Analysis

The SVM model exhibited an accuracy of 83 percentage on the test set, indicating its proficiency in classifying Malayalam vowels. A detailed analysis of precision, recall, and F1 score revealed varying performance levels across different vowels. Notably, high accuracy was observed in vowels such as 'അ', 'ആ', 'ഇ', 'ഈ', and 'ഉ', boasting precision scores ranging from 0.94 to 1.00. These vowels exhibited robust performance, with recall and F1scores above 0.90, indicating accurate classification and prediction. In the study, the accuracy for some vowels like 'ക', 'ഖ', 'ഗ', and 'ഘ' was moderate, with precision scores ranging from 0.82 to 0.86. While slightly lower in precision, they achieved good recall and F1scores above 0.80, indicating reliable classification. However, the vowel 'ഡ' had lower accuracy, with a precision score of 0.84, suggesting a higher rate of misclassification. Additionally, the vowel 'ഠ്' exhibited lower precision and recall values, highlighting challenges in its accurate identification and classification.

3.5 Model Evaluation Summary

The model's performance was impressive, with the following metrics, Accuracy: 83 percentage Macro Average Precision, Recall, F1score: 0.85, 0.83, 0.84 Weighted Average Precision, Recall, F1score: 0.85, 0.83, 0.84. These stats provide a thorough picture of how well the model handled the Malayalam vowel classification task. The high scores indicate the model was quite effective at this challenging job.

Alphabet	Precision	Recall	F1-Score	Support
അ	1	0.99	0.99	1000
ആ	0.99	0.95	0.97	1000
ഇ	0.94	0.9	0.92	1000
ഈ	0.89	0.86	0.88	1000
ഉ	0.86	0.84	0.85	1000
ഊ	0.84	0.82	0.83	1000
ഋ	0.83	0.82	0.83	1000
എ	0.87	0.83	0.85	1000
ഒ	0.86	0.81	0.83	1000
ഔ	0.84	0.81	0.82	1000
ക	0.85	0.83	0.84	1000
ഖ	0.83	0.82	0.82	1000
ഗ	0.83	0.81	0.82	1000
ഘ	0.86	0.8	0.83	1000
ങ	0.85	0.81	0.83	1000
ച	0.85	0.81	0.83	1000
ഛ	0.85	0.82	0.83	1000
ജ	0.82	0.82	0.82	1000
ഞ	0.82	0.83	0.83	1000
ട	0.86	0.82	0.84	1000
ഠ	0.86	0.84	0.85	1000
ഡ	0.84	0.8	0.82	1000
ഢ	0.88	0.82	0.85	1000
ണ	0.84	0.81	0.83	1000
ത	0.84	0.79	0.81	1000
ഥ	0.84	0.82	0.83	1000
ദ	0.85	0.82	0.83	1000
ധ	0.86	0.82	0.84	1000
ന	0.85	0.83	0.84	1000
പ	0.82	0.84	0.83	1000
ഫ	0.84	0.8	0.82	1000

Figure 10. classification report

ബ	0.85	0.82	0.83	1000
ഭ	0.86	0.83	0.85	1000
ര	0.86	0.83	0.85	1000
റ	0.84	0.85	0.84	1000
ല	0.86	0.82	0.84	1000
ള	0.86	0.81	0.84	1000
ഴ	0.83	0.83	0.83	1000
വ	0.84	0.81	0.83	1000
ശ	0.82	0.82	0.82	1000
ഷ	0.85	0.83	0.84	1000
സ	0.88	0.8	0.84	1000
ഹ	0.85	0.81	0.83	1000
ർ	0.42	1	0.59	1000

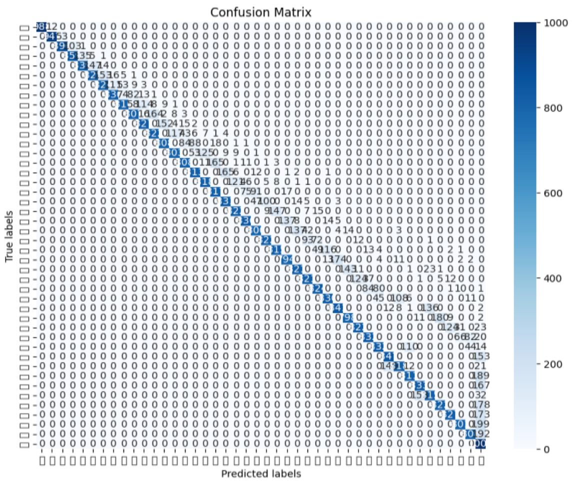
Figure 11. Classification Report

3.6. Conclusion and Implications

The experimental results underscored the effectiveness of the SVM based Malayalam vowel recognition model, with high accuracy achieved for the majority of vowels. While certain vowels posed challenges in classification, the overall system performance remained robust, providing a reliable foundation for further research and development in gesture recognition. The findings of this study hold implications for the advancement of language processing technologies, particularly in the context of Malayalam language applications.

3.7. Experimental Data Presentation

The experimental data, including precision, recall, F1score, and accuracy for each Malayalam vowel, is presented in Figure 9 and Figure 10. Additionally, a confusion matrix shows the classification performance of the SVM model, along with a corresponding matrix comparison graph for visual analysis.



12. confusion Matrix

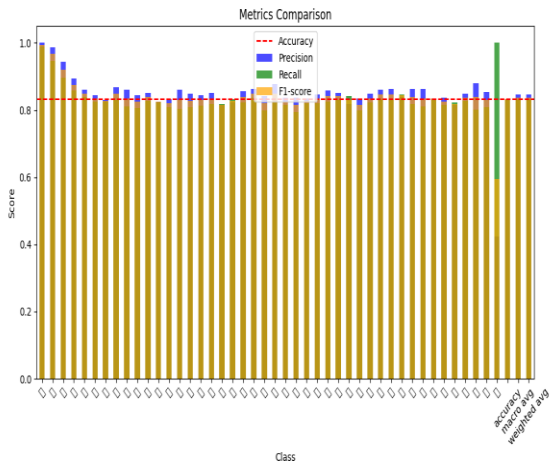


Figure 13. Metrics Comparison

4. FUTURE SCOPE

In future developments, interaction with the system could occur seamlessly through a screen or a small speaker attached to the glove, eliminating the need for a standalone application. Integration of multiple input modalities, such as hand gestures with voice commands or eye movements, could significantly enrich user interaction. Realtime feedback mechanisms could offer immediate guidance based on gestures, assisting users in refining movements effectively. Furthermore, integrating gesture recognition with augmented reality (AR) and virtual reality (VR) environments holds the potential to revolutionize immersive experiences, empowering users to manipulate virtual objects or navigate digital interfaces with unprecedented ease and efficiency.

5. CONCLUSION

The development of a hand wearable device presents an innovative approach to enhancing communication accessibility for deaf and mute individuals, fostering inclusivity within society. By leveraging modern technology, this system effectively addresses communication barriers by integrating various components such as flex sensors, an MPU sensor, an Arduino nano, and a WiFi-BT-BLE MCU module to track hand gestures. Paired with a mobile app, the glove translates these gestures into both text and audio formats, enabling seamless communication between users. The implementation of a real time operating system further optimizes task management, enhancing the system's overall performance and reliability. Additionally, the creation of a dedicated dataset for Malayalam Sign Language, coupled with SVM model training, signifies significant progress in accurately recognizing and interpreting sign language gestures. With an impressive accuracy rate of 83 percentage on the test set, the SVM model demonstrates its efficacy in facilitating communication by accurately identifying Malayalam vowels. This milestone underscores the transformative potential of such technology in bridging communication disparities and fostering meaningful interactions among individuals with varying communication abilities.

4. REFERENCES

[1] Accuracy and precision - wikipedia. https://en.wikipedia.org/wiki/ Accuracy_and_precision, 2024. Accessed: 2024-04-01.
[2] ahrs algorithm - google search. https://shorturl.at/yDORU, 2024. Accessed: 2024-04-01.
[3] Arduino nano: Accelerometer gyroscope compass mpu9250 i2c sensor with visuino : 11 steps - instructables. https://www.instructables.com/ Arduino-Nano-Accelerometer-Gyroscope-Compass-MPU92/, 2024. Accessed: 2024-04-01.
[4] Confusion matrix in machine learning - geeksforgeeks. https:// www.geeksforgeeks.org/confusion-matrix-machine-learning/, 2024. Accessed: 2024-04-01.
[5] F-score - wikipedia. https://en.wikipedia.org/wiki/F-score, 2024. Accessed: 2024-04-01.
[6] Freertos - market leading rtos (real time operating system) for embedded systems with internet of things extensions. https://www.freertos.org/, 2024. Accessed: 2024-02-15.
[7] Hand gesture recognition using flex sensor and machine learning algorithms. https://ieeexplore.ieee.org/document/9398789, 2024. Accessed: 2024-04-01.
[8] Indian sign language. https://islrtc.nic.in/sites/default/files/Interpreter% 20Directory.pdf, 2024. Accessed: 2024-02-01.
[9] Indian sign language interpreter for deaf and mute people. https://ijcrt. org/papers/IJCRT2103178.pdf, 2024. Accessed: 2024-02-01.
[10] Madgwick orientation filter — ahrs 0.3.1 documentation. https://ahrs. readthedocs.io/en/latest/filters/madgwick.html, 2024. Accessed: 202404-01.
[11] Mpu-9250 via spi · issue 62 · kriswiner/mpu9250 · github. https://github. com/kriswiner/MPU9250/issues/62, 2024. Accessed: 2024-02-15.
[12] Nish - indian sign language alphabet in malayalam - released. https: //nish.ac.in/index.php/others/news/882-indian-sign-language, 2024. Accessed: 2024-02-01.
[13] Nish - national institute of speech hearing. https://www.nish.ac.in/, 2024. Accessed: 2024-02-01.
[14] Precision and recall - wikipedia. https://en.wikipedia.org/wiki/Precision_ and_recall, 2024. Accessed: 2024-04-01.
[15] A prototype malayalam to sign language automatic translator. https: //arxiv.org/pdf/1412.7415.pdf, 2024. Accessed: 2024-02-01.
[16] Roll, pitch, and yaw | how things fly, 2024. Accessed: 2024-04-01. [17] Sign language interpreter | iiitb. https://cognitive.iiitb.ac.in/isli/, 2024. Accessed: 2024-02-01.
[18] sin.png (680×390). https://malabarinews.com/wp-content/uploads/2021/ 09/sin.png, 2024. Accessed: 2024-04-02.
[19] The rights of persons with disabilities act, 2016.//www.indiacode.nic.in/bitstream/123456789/15939/1/the_rights_of_persons_with_disabilities_act%2C_2016.pdf, 2024.2024-02-01.
[20] Admin. Flex sensor arduino interfacing - measure bend/resistance. https: //how2electronics.com/interfacing-flex-sensor-with-arduino/, 2024. Accessed: 2024-04-01.
[21] Scott Campbell. Basics of the i2c communication protocol. https: //www.circuitbasics.com/basics-of-the-i2c-communication-protocol/, 2024. Accessed: 2024-04-01.
[22] Chih-Chung Chang and Chih-Jen Lin. Libsvm: A library for support vector machines. ACM Transactions on Intelligent Systems and Technology, 2, 07 2007.
[23] Pankaj Chougale, Vaibhav Yadav, Anil Gaikwad, Bharati Student, and Vidyapeeth. Firebase -overview and usage, 08 2022.
[24] Deepali Mali, Nitin Limkar, and Satish Mali. Indian sign language recognition using svm classifier. SSRN Electronic Journal, 2019.
[25] Kun Xia, Weiwei Lu, Hongliang Fan, and Qiang Zhao. A sign language recognition system applied to deaf-mute medical consultation. Sensors, 22(23):9107, November 2022.
[26] Tong Yu and Hong Zhu. Hyper-parameter optimization: A review of algorithms and applications. 2020.

