# Sign Language Detection and Translation using Smart Glove

**Sunila Maharjan*, Dr. Subeksha Shrestha, Dr. Sandra Fernando**

BE(Computer Engineering), MSc in Artificial Intelligence, BSc(Hons), MSc, AFHEA, PHD, Assistive Technology Research Group, SCDM, London Metropolitan University, 166-220 Holloway Rd, London N7 8DB, UK

sum0609@my.londonmet.ac.uk ; s.shrestha3@londonmet.ac.uk ; s.fernando@londonmet.ac.uk

---

**Abstract** Communication through sign language is essential for hard of hearing or deaf people. However, effective interaction and involvement in many facets of society are hampered by the communication gap that exists between sign language users and those who do not comprehend it. In order to solve this pressing problem, a smart glove-based system for real-time sign language detection and translation is being developed as part of this research project. The project's goals include designing and creating a smart glove prototype that has an MPU-6050 and five flex sensors. Intricate finger and hand movements, which are essential to sign language, are precisely captured by these sensors. To handle the sensor input, a Recurrent Neural Network (RNN) with Long Short-Term Memory (LSTM) is developed, which allows the recognition and categorization of sign language gestures. The system's first focus is on American Sign Language (ASL) gestures that correlate to the English letters A through E and the numerals 0 through 9. The real-time translation of sign language motions into spoken language and text is a significant innovation. A text-to-speech engine is integrated into the system, enabling simultaneous textual and audio outputs. Sign gestures made by users are translated into text on a screen instantaneously and used to communicate with the system. In order to facilitate communication with both sign language and non-sign language users, the system simultaneously translates the text into spoken language. The main focus of the research project is alphabet and number detection and translation in sign language. It skips over how to recognise convoluted sign language statements or how sign languages differ from one another. Due to time and budget constraints, evaluation focuses on accuracy and performance measures instead of in-depth user research.

**Keywords:** Artificial Intelligence, machine learning algorithms, RNN, LSTM, Arduino, python, TensorFlow, sensors, ASL, Assistive Technology

## 1. Introduction

Sign language is essential for individuals with hearing impairments, offering a unique mode of expression. According to WHO, over 5% of the world's population are deaf (Organization, 2023). However, the gap in communication between sign language users and those who don't understand it, presents challenges. Sign languages vary across different countries and regions, with each having its own unique signs and cultural nuances. According to National Geographic Society, there are more than 300 sign languages around the world (Society, 2022). It is not only convenient to bridge the communication gap between people who use sign language and non-users; it is also a critical first step in building an inclusive and just society. The foundation of social interactions, work, education, and healthcare is effective communication. With addition to limiting participation, difficulties communicating thoughts and feelings with sign language create a vicious cycle of exclusion and isolation for those who use it. The goal of this research is to develop a smart glove-based real-time sign language detection and translation system that will enable people with hearing loss to get high-quality education, succeed in the job, and access necessary services with dignity. This project is an investment in a society where everyone can enrich human connection, regardless of communication style, and it also demonstrates a dedication to inclusivity.

The study introduces a smart glove-based system to detect and translate sign language gestures in real-time. This system is designed to recognize American Sign Language (ASL) gestures representing English alphabets (A to E) and numerical digits (0 to 9). It leverages a smart glove equipped with five flex sensors and an MPU-6050, with sensor data being processed by Recurrent Neural Networks (RNN) with Long Short-Term Memory (LSTM). The innovation lies in the real-time translation of these gestures into both text and speech, facilitating communication between sign language users and those who do not understand sign language.

Figure 1 Sign Language Gesture for Numbers 0-9

Figure 2 Sign Language Gesture for Alphabet A-E

The research questions focus on the effectiveness of the smart glove in real time, suitable machine learning techniques, translation accuracy, and the challenges of this approach. This research holds significance in improving communication for individuals with hearing impairments, promoting inclusivity in education, employment, and social interactions.

Individuals with hearing impairments heavily rely on sign language as a primary means of communication. Yet, the existing communication gap between sign language users and those unfamiliar with this language creates profound challenges. This research project is motivated by the desire to bridge this communication divide and enable effective interactions in various aspects of society. Motivation is rooted in the need for inclusive communication in education, employment, and everyday life.

The remainder of the paper will delve into the development process, including the design and methodology, as well as the components used in this system, such as the hardware and software. It will discuss data collection, and processing. The machine learning models employed, particularly RNN and LSTM, will be detailed. The paper will also present results and evaluation of the system's performance. Finally, the paper emphasizes the significance of this research in fostering inclusivity, particularly in education and employment. This innovative approach aims to inspire further creative applications, making the world more accessible for individuals with hearing impairments.

## 2. Related Work

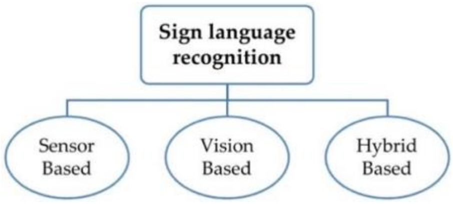


Figure 3 Sign Language Recognition Approaches (Ahmed, M.A., Zaidan, B.B., Zaidan, A.A., Salih, M.M. and Lakulu, M.M.B., 2018)

There are three main approaches exist: computer vision-based, sensor-based and hybrid-based. Computer vision relies on visual cues, while sensor-based uses wearable devices to capture movements and hybrid-based combines both sensor and vision-based approaches.

Vision-based techniques often employ hand tracking, feature extraction, and gesture recognition. Depth cameras, like the Kinect, can capture 3D sign language data (Amin, M.S., Rizvi, S.T.H. and Hossain, M.M., 2022). The article by Pentland, T. Starner and A., used single colour camera and Hidden Markov Models (HMM) (T. Starner and A. Pentland, 1995). Bantupalli, K. and Xie, Y. has used deep learning - Convolution Neural Network (CNN) and Recurrent Neural Network (RNN) and computer vision (Bantupalli, K. and Xie, Y., 2018; Ahmed, M.A., Zaidan, B.B., Zaidan, A.A., Salih, M.M. and Lakulu, M.M.B., 2018).

Sensor-based systems incorporate various wearable sensors to capture hand movements and orientation for recognition. Machine learning algorithms, such as CNNs and RNNs with LSTM, are commonly used for sign language recognition. There are various Sensor-based systems incorporate wearable sensors, such as flex sensors, tactile sensors (Amin, M.S., Rizvi, S.T.H. and Hossain, M.M., 2022), hall sensors (Choudhary, D.K., Singh, R. and Kamthania, D., 2021), pressure sensors (Ahmed, M.A., Zaidan, B.B., Zaidan, A.A., Salih, M.M. and Lakulu, M.M.B., 2018), IMU sensors, TOF sensors, proximity sensors, abduction sensors, data gloves, 3-axis gyroscope and accelerometer, (Ahmed, M.A., Zaidan, B.B., Zaidan, A.A., Salih, M.M. and Lakulu, M.M.B., 2018) to capture and analyse hand movements during sign language communication.

Sign language datasets consist of videos or motion capture data. Sign languages do not have a standardized written form, so large text corpora independent of videos are not available (Bragg, D., Koller, O., Bellard, M., Berke, L., Boudreault, P., Braffort, A., Caselli, N., Huenerfauth, M., Kacorri, H., Verhoef, T. and Vogler, C., 2019). Suri, A., Singh, S.K., Sharma, R., Sharma, P., Garg, N. and Upadhyaya, R. had created simple dataset for 5 fingers. When one of the fingers bent, its equivalent data is compared and recognised and translated into text and voice (Suri, A., Singh, S.K., Sharma, R., Sharma, P., Garg, N. and Upadhyaya, R., 2020). Salem, N., Alharbi, S., Khezendar, R. and Alshami, H. also created their own dataset by recording reading of each sensor using a voltage divider circuit (Salem, N., Alharbi, S., Khezendar, R. and Alshami, H., 2019). Ibrahim, N.B., Selim, M.M. and Zayed, H.H. used dataset that consist of 450 coloured ArSL videos which represented 30 Arabic words that were selected as daily used common words in school (Ibrahim, N.B., Selim, M.M. and Zayed, H.H., 2018).

Table 1: Comparison of Sign Language Recognition Approaches

| **Aspect** | **Vision-Based** | **Sensor-Based** | **Hybrid-Based** |
|---|---|---|---|
| **Data Source** | Video/Image Data | Wearable Sensors | Video/Image Data |
| **Key Advantage** | Non-invasive | Accurate Movements | Comprehensive |
| **Key Disadvantage** | Sensitive to Light | Device Dependency | Costly |
| **Primary Sensors** | Cameras | Flex, Tactile, Hall Sensors | Cameras, Wearable Sensors |
| **Machine Learning Algorithms** | CNN, RNN (LSTM) | Various | Advanced ML Algorithms |

| | | | |
|---|---|---|---|
| **Accuracy** | Variable | High | High |
| **Real-time Performance** | Challenging | Yes | Yes |
| **Gesture Segmentation** | Often Challenging | Yes | Yes |
| **User Comfort** | Comfortable | Device Dependency | Varies |

In sign language gesture recognition, different approaches have distinct characteristics. Vision-based methods utilize video and image data, offering a non-invasive solution. However, they can be sensitive to lighting conditions. Sensor-based approaches rely on wearable sensors for precise hand movement recognition, but this makes them device-dependent, potentially affecting daily communication without the device. Hybrid methods combine video data with sensor inputs, providing a comprehensive solution but can be complex and costly. Primary sensors used include cameras for vision-based systems, flex, tactile, and hall sensors for sensor-based solutions, and a combination of cameras and wearable sensors for hybrid approaches. Machine learning algorithms differ; vision-based methods employ CNNs and RNNs like LSTM, while sensor-based systems use various algorithms, and hybrid approaches leverage advanced ML techniques. Sensor-based and hybrid-based methods typically achieve high accuracy and real-time performance, whereas vision-based recognition can be challenging in real-time scenarios. Gesture segmentation is more reliable in sensor-based and hybrid-based methods, while user comfort varies, with vision-based being the most comfortable, sensor-based being device-dependent, and hybrid-based falling in between.

## 3. Materials and Methodology

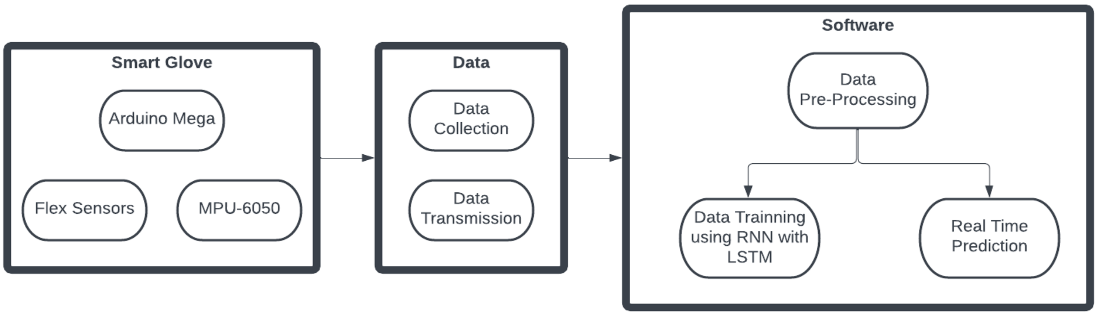


Figure 4 System Architecture

The proposed system architecture is based mostly on a custom-designed smart glove that incorporates several key hardware components. This glove acts as the primary interface for capturing hand orientation and finger movements since it has flex sensors and an MPU-6050. This hardware ensemble operates as a conduit, analysing and transmitting sensor data to a computer for in-depth analysis and prediction. It is connected to a microcontroller, more precisely the Arduino Mega 2560.

### 3.1 Flex Sensors

Smart gloves rely heavily on flex sensors, specialized devices that measure bending or flexing, to capture subtle movements. Among various types available, the project chose for resistive flex sensors due to their compatibility, adaptability to different signing styles, real-time responsiveness, robustness, accuracy, simplicity, and cost-effectiveness. The Spectra Symbol Resistive Flex Sensor 4.5, which is well-known for its ability to adapt to glove designs and quick reaction to finger bending, has been specifically incorporated into the project (spectrasymbol, 2014).

### 3.2 MPU 6050

The MPU-6050 is a multipurpose sensor that combines accelerometer and gyroscope capabilities. It is essential for recording hand orientations and movements. This project employs a particular variation, the "AZDelivery GY-521 MPU-6050 3 Axis Gyroscope and Accelerometer 6DOF Sensor Module," which is well-known for its precise motion detection and orientation data, to recognise sign language gestures (Amazon-MPU6050, 2019).

### 3.3 Arduino Mega 2560

The Arduino Mega 2560, which is well-known for having many I/O pins, is the preferred microcontroller for this project. This microcontroller was chosen because it can be seamlessly integrated into the system and has distinct pins for the MPU-6050 and can work with the necessary amount of ADC Analogue pins (store.arduino.cc, 2021).


Figure 5 Resistive Flex Sensor 4.5" (Amazon-Flex, 2011)

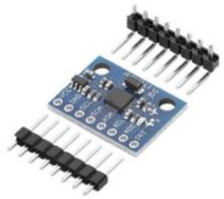
Figure 6 MPU-6050 (Amazon-MPU6050, 2019)

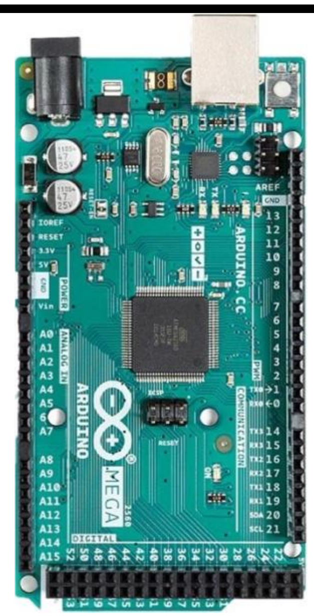

Figure 7 Arduino Mega 2560 (Amazon-Mega, 2011)


Figure 8 Jumper Wire Cable (Cable, 2017)

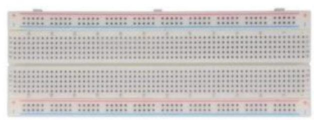
Figure 9 Breadboard (Amazon-Breadboard, 2019)

Data collection involved 18 participants who performed a range of sign language gestures, encompassing numbers from 0 to 9 and alphabets from A to E in American Sign Language. The data collected from the glove's sensors was methodically organized and stored in a structured CSV format for further analysis.

Data pre-processing was a pivotal step, which began with an initial check for null or missing data. Categorical gesture data was then subjected to one-hot encoding, facilitating compatibility with machine learning algorithms. Columns that didn't contribute significantly to gesture recognition were pruned, streamlining the dataset. The dataset was divided into training, testing, and validation subsets, a crucial aspect for precise model evaluation and the avoidance of overfitting.

Feature extraction and representation were integral to the project. While feature scaling techniques like min-max scaling and z-score normalization were considered, it was observed that the accuracy was higher without their implementation, leading to their omission in the final code.

Machine learning models, Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM), were selected for gesture recognition.

### 3.1 Recurrent Neural Network (RNN):

Recurrent Neural Network (RNN) is a specialized architecture tailored for processing sequential and temporal data. RNNs differ from traditional feedforward neural networks due to the fact they have loops that enable information persistence between steps. RNNs are used in many different applications, including natural language processing, time series analysis, and sign language recognition (RNN, 2022-

2023). They are particularly effective at modelling dependencies in sequential data. Its ability to handle sequential data, recurrence that allows information transfer between steps, maintenance of a hidden state that acts as memory, time unfolding for information flow visualisation, and challenge with the vanishing gradient problem are some of its key features. Traditional RNNs struggle to capture long-term dependencies, despite their potency. This problem is solved by more sophisticated variations, such as Long Short-Term Memory (LSTM), which was selected for this project.

### 3.2 Long Short-Term Memory (LSTM):

Long Short-Term Memory (LSTM) stands out for its efficacy in simulating sequential data, especially beneficial for tasks involving time series and sensor data. Long-term interdependence in sequences is recorded by LSTMs, which have memory cells with the capacity to store information for an extended period of time (LSTM, 2022-2023). The input, forget, and output gates are integrated into the architecture to regulate the flow of information. The input gate governs the inclusion of new information, the forget gate decides what to discard, and the output gate controls the information outputted to the next layer or as the final prediction. Memory cells and gates collaborate to enable LSTMs to efficiently acquire and retain important information over long sequences, which makes them very useful for tasks requiring an essential understanding of long-term dependencies.

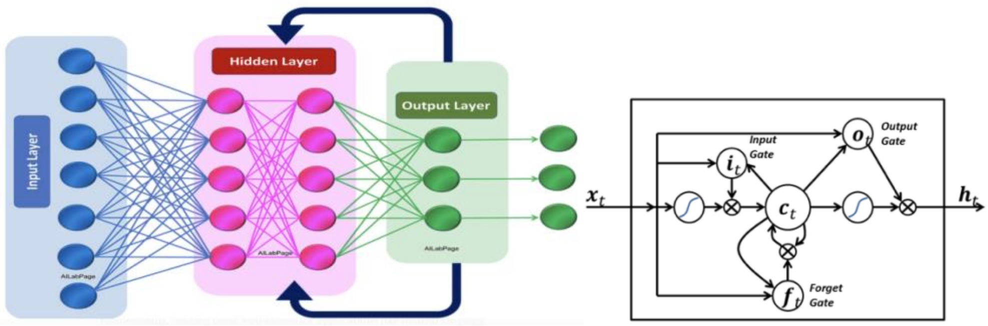


Figure 10 Recurrent Neural Network (kondreddy, 2023) and Long Short-Term Memory (Commons, 2015)

The hardware components, including flex sensors, MPU-6050, Arduino Mega 2560, resistors, a breadboard, and jumper cables, were meticulously connected following a step-by-step procedure. These components underwent functionality tests to ensure their proper operation within the glove.

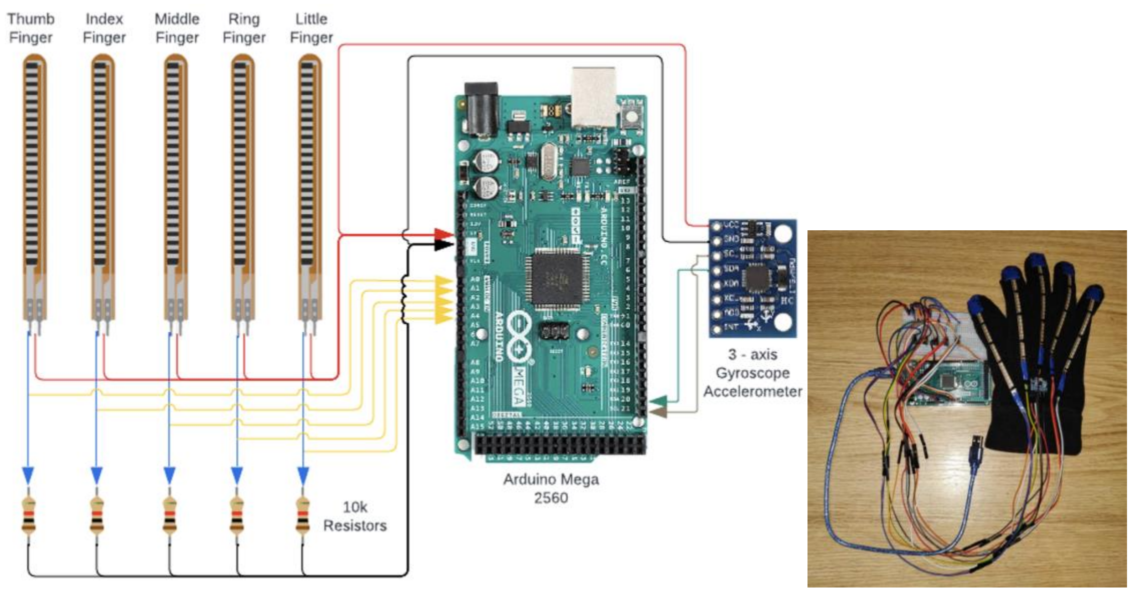


Figure 11 Circuit Connection and Prototype

In terms of software tools, Arduino IDE was utilized for code development. Jupyter Notebook served as the web-based interactive environment for code creation, enabling the incorporation of live code, equations, visualizations, and narrative text.

Incorporation of various libraries was essential for data manipulation, visualization, and machine learning. These libraries included pandas for data handling, NumPy for numerical and mathematical operations, seaborn for data visualization, matplotlib for creating various types of plots and graphs, TensorFlow for machine learning, and scikit-learn for diverse tasks related to data pre-processing, feature selection, model training, and evaluation.

This comprehensive methodology provided a systematic approach that initiated with hardware setup and data collection and proceeded through data pre-processing, feature extraction, machine learning, testing, and the intricate process of circuit connection. It laid the foundation for an accurate sign language recognition system through the smart glove.

## 4. Data Collection and Processing

The project relies on collecting data about American Sign Language (ASL) gestures, primarily focusing on numbers and alphabets. This is accomplished using a smart glove prototype equipped with flex sensors and an MPU-6050 to capture hand gestures and orientation. The project gathered data from a diverse group of participants, totalling approximately 18 individuals, including us. The data was meticulously captured and organized, with each participant performing various sign language gestures. Gestures included numbers from 0 to 9 and alphabets from A to E in American Sign Language (ASL). The collected data was then stored in a structured CSV format to facilitate efficient handling and further analysis. Two separate datasets were created for Numbers and Alphabets as there were similar gestures for both numbers and alphabets.

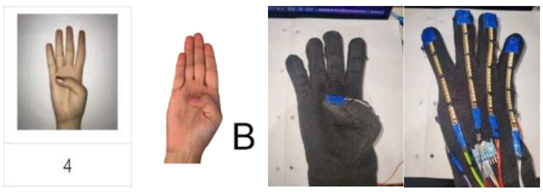


Figure 12 Gesture for number 4 and alphabet B is similar

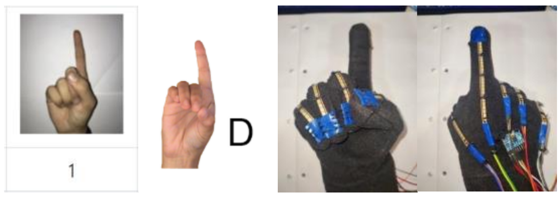


Figure 13 Gesture for number 1 and alphabet D is similar

Table 2 Data Collection Sample

| | A | B | C | D | E | F | G | H | I | J | K | L | M |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | Thumb | IndexFinger | MiddleFinge | RingFinger | LittleFinger | acceleromet | acceleromet | acceleromet | temperature | gyro_x | gyro_y | gyro_z | Gesture |
| 2 | 387 | 403 | 399 | 402 | 388 | 1348 | 780 | 17504 | 25.09 | -689 | -229 | -32 | 0 |
| 3 | 387 | 341 | 338 | 336 | 343 | 1484 | 812 | 17500 | 25.19 | -674 | -231 | -22 | 0 |
| 4 | 388 | 386 | 387 | 387 | 375 | 1404 | 708 | 17484 | 25.19 | -666 | -223 | -6 | 0 |
| 5 | 368 | 306 | 299 | 292 | 309 | 15372 | -6588 | 6264 | 25.19 | 2762 | -2544 | 6863 | 0 |
| 6 | 367 | 167 | 166 | 165 | 225 | 11468 | -520 | 12980 | 25.24 | 7639 | 2582 | 4205 | 0 |
| 7 | 370 | 309 | 307 | 305 | 308 | 16720 | -748 | 8556 | 25.33 | -6898 | -5946 | -7595 | 1 |
| 8 | 365 | 206 | 204 | 204 | 249 | 12160 | -2216 | 10836 | 25.33 | 4148 | 3573 | 8744 | 1 |
| 9 | 346 | 280 | 280 | 280 | 292 | 9576 | 6380 | 7868 | 25.19 | 31658 | 8237 | 1079 | 1 |
| 10 | 353 | 142 | 138 | 135 | 201 | 9288 | 8312 | 2036 | 25.24 | 7330 | -7707 | 2744 | 1 |
| 11 | 407 | 977 | 981 | 984 | 751 | 8696 | 14928 | 1612 | 25.24 | -3446 | -5461 | -3496 | 1 |
| 12 | 382 | 543 | 540 | 536 | 484 | 4552 | -2500 | 19660 | 25.38 | -14416 | 5685 | -7532 | 2 |
| 13 | 380 | 102 | 101 | 102 | 187 | 5308 | 260 | 16824 | 25.28 | -701 | -459 | -169 | 2 |
| 14 | 387 | 124 | 115 | 114 | 182 | 14752 | 5616 | 8588 | 25.33 | 709 | 87 | 1938 | 2 |
| 15 | 396 | 113 | 108 | 105 | 174 | 15716 | 3628 | 6768 | 25.52 | 72 | -1833 | -386 | 2 |
| 16 | 396 | 218 | 217 | 216 | 243 | 15600 | 3720 | 7116 | 25.47 | -430 | -55 | 175 | 2 |
| 17 | 396 | 199 | 195 | 188 | 231 | 15448 | 3668 | 7576 | 25.47 | -767 | -158 | -35 | 3 |
| 18 | 405 | 219 | 221 | 223 | 255 | -3248 | -2184 | 15108 | 25.42 | -1940 | 7331 | 1293 | 3 |
| 19 | 381 | 733 | 730 | 727 | 592 | -3928 | -4152 | 15492 | 25.47 | -4201 | 1263 | 635 | 3 |
| 20 | 376 | 525 | 525 | 524 | 488 | -2724 | -3260 | 16788 | 25.52 | -525 | -74 | -19 | 3 |
| 21 | 386 | 472 | 470 | 468 | 445 | -2956 | -3212 | 16592 | 25.47 | -447 | -238 | 57 | 3 |
| 22 | 387 | 348 | 347 | 345 | 361 | -2876 | -3184 | 16772 | 25.57 | -661 | -222 | -31 | 4 |
| 23 | 387 | 359 | 359 | 359 | 362 | -2780 | -3100 | 16852 | 25.52 | -525 | -165 | 5 | 4 |
| 24 | 387 | 334 | 332 | 330 | 340 | -2840 | -3172 | 16776 | 25.61 | -638 | -156 | 41 | 4 |
| 25 | 372 | 307 | 307 | 307 | 323 | -2816 | -2888 | 16708 | 25.52 | -637 | -182 | 3 | 4 |
| 26 | 384 | 365 | 363 | 362 | 355 | -2996 | -3024 | 16724 | 25.66 | -703 | -200 | -26 | 4 |
| 27 | 386 | 310 | 309 | 309 | 323 | -2832 | -3092 | 16760 | 25.71 | -672 | -222 | -33 | 5 |
| 28 | 384 | 370 | 370 | 369 | 359 | -3024 | -3132 | 16696 | 25.66 | -452 | -86 | 142 | 5 |
| 29 | 387 | 327 | 325 | 323 | 331 | -2976 | -3084 | 16848 | 25.8 | -629 | -331 | -93 | 5 |

Created CSV file has 13 columns. The columns represent different features, such as Thumb, IndexFinger, MiddleFinger, RingFinger, LittleFinger, accelerometer_x, accelerometer_y, accelerometer_z, temperature, gyro_x, gyro_y, gyro_z, and Gesture.

- The Thumb, IndexFinger, MiddleFinger, RingFinger, and LittleFinger columns contain numerical data, which represents readings from the respective flex sensors.
- The accelerometer_x, accelerometer_y, accelerometer_z, temperature, gyro_x, gyro_y, and gyro_z columns also contain numerical data, representing readings from the MPU-6050.
- The Gesture column represents the categorical variable, which indicates the gesture associated with the flex sensor readings. For Numbers, Gesture column will have 0-9 values for 0-9 numbers and for Alphabets, it will have 0-4 values for A-E alphabets.
- The data processing phase is a pivotal step in this project, encompassing several crucial tasks to prepare data for analysis and machine learning model training.

Table 3 Gesture Columns for Numbers and Alphabets

| Gesture | Gesture for |
|---:|---|
| 0 | Number 0 |
| 1 | Number 1 |
| 2 | Number 2 |
| 3 | Number 3 |
| 4 | Number 4 |
| 5 | Number 5 |
| 6 | Number 6 |
| 7 | Number 7 |
| 8 | Number 8 |
| 9 | Number 9 |

| Gesture | Gesture for |
|---:|---|
| 0 | Alphabet A |
| 1 | Alphabet B |
| 2 | Alphabet C |
| 3 | Alphabet D |
| 4 | Alphabet E |

**Data Understanding and Visualization:** The collected data is visualized using various techniques, including Seaborn, Matplotlib's Pyplot, and animation to reveal patterns and insights.

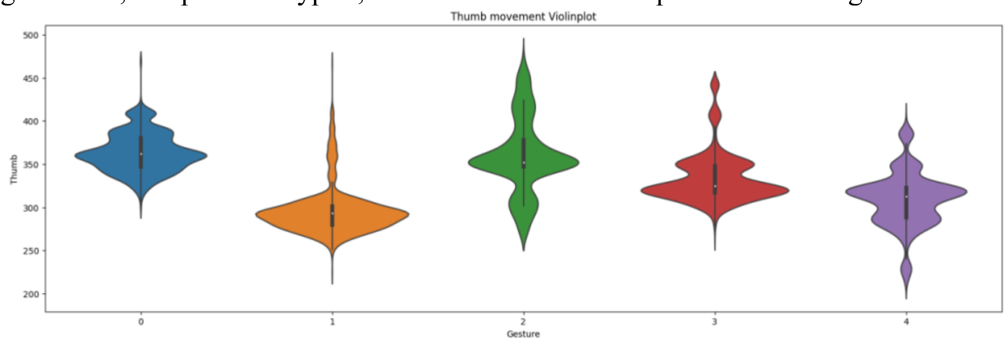


Figure 14 Violon Plot using Seaborn

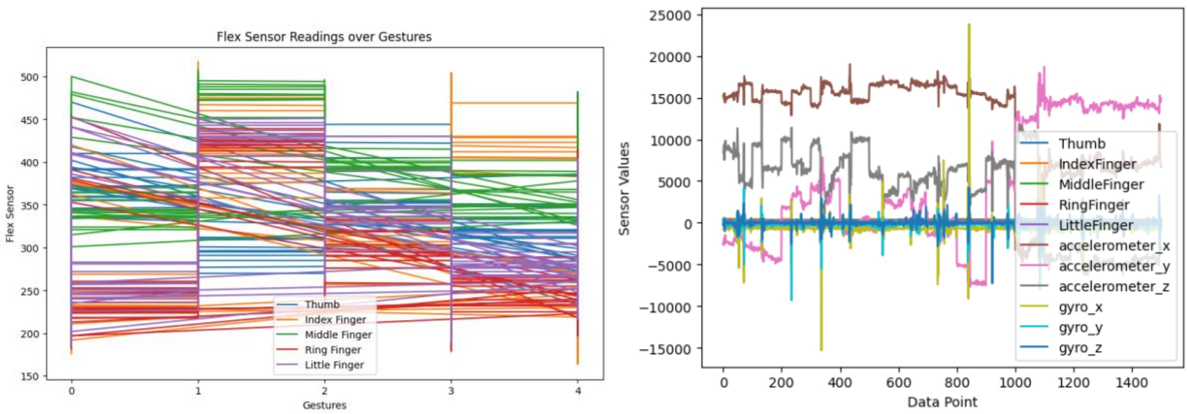


Figure 15 Line plot for flex sensor reading and All sensor data using Animation Module

Violin plots are used to show how thumb movement varies across different gestures, while line plots illustrate flex sensor for different fingers and gestures. An animation plot provides a dynamic visualization of how sensor values vary as different gestures are performed.

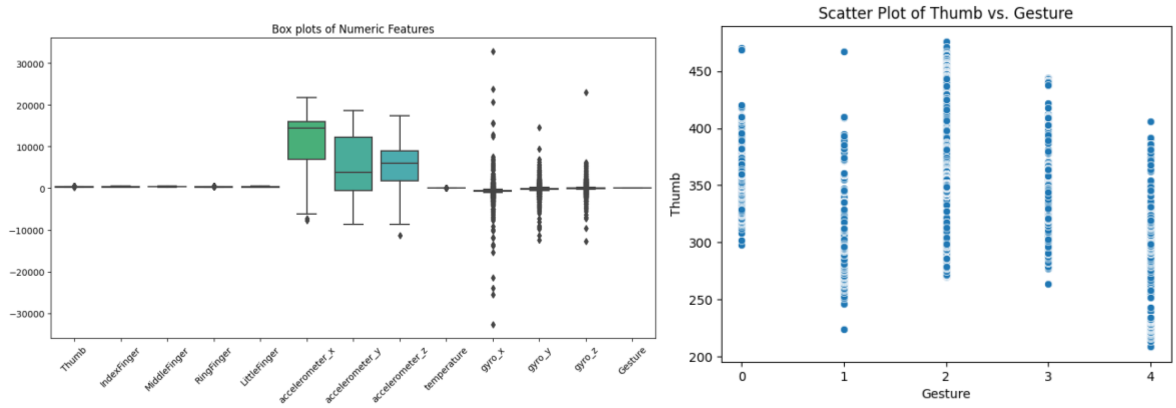


Figure 16 Box plot and Scatter plot using Seaborn

A box plot summarizes key statistics such as the median, quartiles, and potential outliers. A box plot displays these statistics through a box and "whiskers and scatter plot helps in gaining insights into how the flex sensors behave during different hand movements

**Data Pre-processing:** This phase includes data cleaning to identify and correct errors, ensuring there are no missing values. Data transformation techniques, like one-hot encoding, are applied to make data suitable for analysis. Unnecessary or redundant information are removed to streamline the dataset. Data splitting is performed to create training and testing subsets for model training. Data reshaping is done to ensure compatibility with machine learning algorithms.

**Model Training:** An RNN with LSTM layers is used for time series data, with specific hyperparameters like epochs, batch size, and dropout rates. Early stopping is introduced to prevent overfitting. The trained model is saved in HDF5 format for later use.

```
model.summary()
Model: "sequential"
 Layer (type)                Output Shape              Param #
=================================================================
 lstm (LSTM)                 (None, None, 1024)        4218880

 lstm_1 (LSTM)               (None, None, 512)         3147776

 dropout (Dropout)           (None, None, 512)         0

 lstm_2 (LSTM)               (None, None, 512)         2099200

 lstm_3 (LSTM)               (None, 512)               2099200

 dropout_1 (Dropout)         (None, 512)               0

 dense (Dense)               (None, 32)                16416

 dropout_2 (Dropout)         (None, 32)                0

 dense_1 (Dense)             (None, 10)                330

=================================================================
Total params: 11581802 (44.18 MB)
Trainable params: 11581802 (44.18 MB)
Non-trainable params: 0 (0.00 Byte)
```

Figure 17 Model Summary

**Model Evaluation:** Model performance is evaluated on both the testing set and predictions, and metrics like loss and accuracy are calculated. Classification reports and confusion matrices provide further insights into the model's performance. After careful evaluation, model evaluation is done in real time prediction focusing on accuracy and performance.

**Real-Time Prediction with Text-To-Speech Translation:** This phase involves two key stages: development and implementation. The trained model is deployed for real-time prediction, and data collection for these predictions mirrors the data collection process.

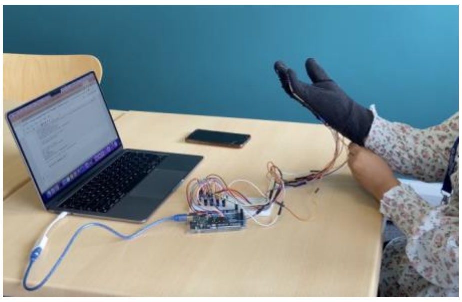

Figure 18 Prototype in Real time Usage

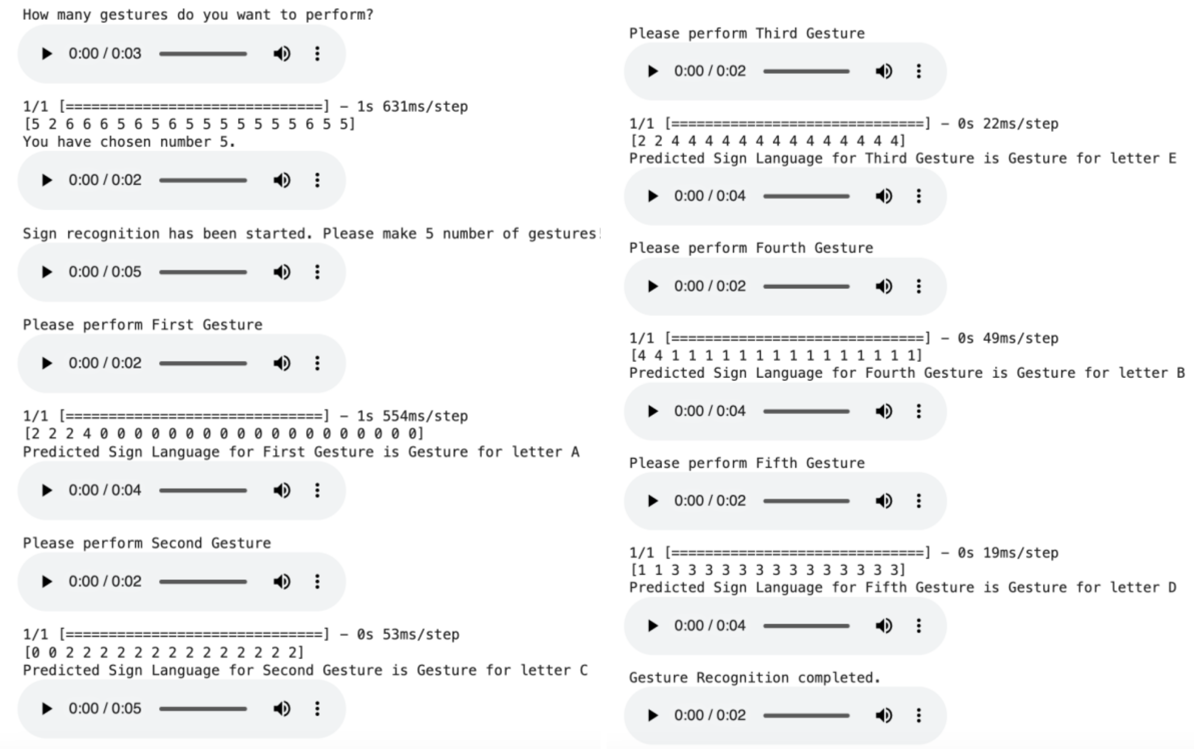


Figure 19 Real time prediction

Users perform sign language gestures in real-time, and the prototype interprets these gestures using the deployed machine learning model. The interpreted gestures are converted into text and then into speech using a text-to-speech engine, enabling real-time communication for both sign language and non-sign language users.

## 5. Results and discussion

Focus Questions of the study:

- How effectively can the smart glove capture and represent hand movements using the collected sensor data?
- What machine learning techniques and algorithms are most suitable for recognizing and classifying sign language gestures based on the sensor data?
- How accurately can the system translate detected sign language gestures into spoken language or text?

- What are the limitations and challenges of using a smart glove-based approach for sign language detection and translation?
- What is the new method that is used in this paper that make different from the existing methodologies?

The evaluation metrics and performance measures employed in this project provided a comprehensive understanding of the LSTM model's capabilities in sign language gesture prediction.

```
Epoch 67/1000
38/38 [==============================] - 2s 53ms/step - loss: 0.4098 - accuracy: 0.8539 - val_loss: 0.4093 - val_accu
racy: 0.8642
Epoch 68/1000
38/38 [==============================] - 2s 52ms/step - loss: 0.4230 - accuracy: 0.8523 - val_loss: 0.4352 - val_accu
racy: 0.8417
Epoch 69/1000
38/38 [==============================] - 2s 53ms/step - loss: 0.4172 - accuracy: 0.8564 - val_loss: 0.4205 - val_accu
racy: 0.8433
Epoch 70/1000
38/38 [==============================] - 2s 53ms/step - loss: 0.4162 - accuracy: 0.8521 - val_loss: 0.4158 - val_accu
racy: 0.8592
Epoch 71/1000
37/38 [============================>.] - ETA: 0s - loss: 0.4175 - accuracy: 0.8545Restoring model weights from the en
d of the best epoch: 51.
38/38 [==============================] - 2s 54ms/step - loss: 0.4173 - accuracy: 0.8548 - val_loss: 0.4092 - val_accu
racy: 0.8425
Epoch 71: early stopping
Overall execution time: 0 hours, 2 minutes, and 21 seconds
```

Figure 20 Model Evaluation

```python
# Generate a classification report by comparing the true class labels (y_test) with the
# predicted class labels (y_pred_labels).
report = classification_report(y_test, y_pred_labels)

# Print the classification report
print(report)
```

```
47/47 [==============================] - 1s 13ms/step
              precision    recall  f1-score   support

           0       0.91      0.97      0.94       307
           1       0.88      0.92      0.90       300
           2       0.79      0.87      0.82       290
           3       0.86      0.61      0.72       285
           4       0.85      0.90      0.88       318

    accuracy                           0.86      1500
   macro avg       0.86      0.86      0.85      1500
weighted avg       0.86      0.86      0.85      1500
```

Figure 21: Classification Report

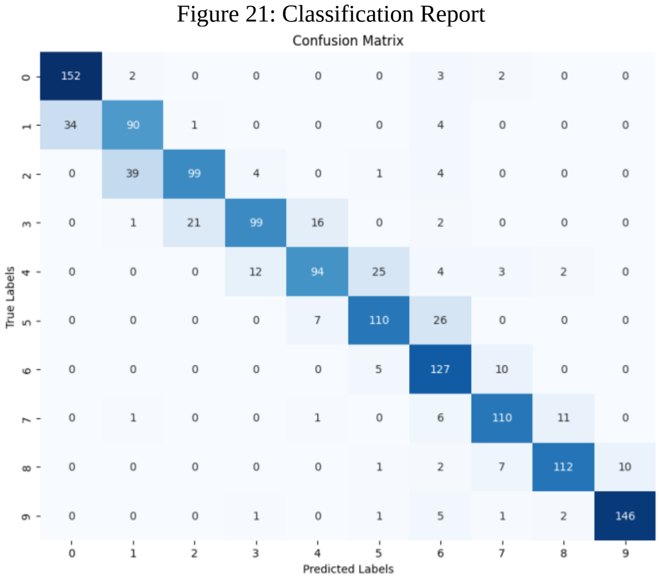


Figure 22: Confusion Matrix

Notably, the model consistently demonstrated remarkable accuracy rates, surpassing 85% in real-time sign detection. These exceptional results underscore the model's practicality and reliability in translating sign language. The evaluation of the model encompasses a thorough analysis on both the testing set and predictions. Essential metrics, including loss and accuracy, are meticulously calculated to gauge the model's overall performance. Beyond conventional measures, classification reports and confusion matrices are employed, offering deeper insights into the nuances of the model's predictive capabilities.

The evaluation process extends to real-time predictions, emphasizing not only accuracy but also real-world performance. This real-time assessment is crucial in ensuring that the model not only performs well in controlled testing scenarios but also demonstrates reliability and effectiveness in dynamic, real-world applications. The focus on accuracy and performance underscores the commitment to creating a robust and dependable system for sign language detection and translation.

Another key finding is the model's resilience to external environmental factors, setting it apart from traditional sign language recognition methods. It remains largely unaffected by variations in lighting conditions, background interference, or differences in skin colour. This robustness enhances its real-world applicability, ensuring seamless operation regardless of the user's surroundings. Additionally, the technology's adaptability for anytime, anywhere use emerged as a pivotal discovery, further expanding its potential applications. Users can confidently rely on its accuracy and performance indoors, outdoors, or in varying lighting conditions.

A particularly ground-breaking feature is the model's personalized adaptation to individual user styles and nuances, facilitated through the personalized training phase. This adaptability enhances not only the system's accuracy for each user but also fosters a highly personalized and intuitive user experience. Users can interact with the system in a manner that aligns with their unique signing styles and preferences. The user-centric design, marked by adaptability and robust performance, is expected to drive wider adoption among diverse user groups, mitigating potential barriers to usage and enhancing overall accessibility. This project's results and discussions collectively underscore the transformative potential of the smart glove system in the realm of sign language detection and translation.

This research introduces a novel approach based on machine learning integration, which sets it apart from existing sensor-based sign language recognition and translation systems. By using a Recurrent Neural Network (RNN) with Long Short-Term Memory (LSTM) instead of the traditional methods that rely on static stored data for comparisons (Sapkota, B., Gurung, M.K., Mali, P. and Gupta, R., 2018), the suggested method introduces a system that is both dynamic and adaptive. Notably, the study recognises inherent variability in sign language expressions by gathering and using unique data from individuals, departing from conventional datasets. When processing sequential data, the RNN with LSTM architecture is very helpful as it fits in well with the complexities of sign language gestures. This method is more direct and user-centric as it uses acquired data patterns from users directly, instead of converting sensor data to point clouds for comparison. Importantly, unlike previous one-size-fits-all methods, the emphasis on user-specific training improves flexibility to individual signing styles. To put it briefly, this paper's innovative methodology combines personalised training, machine learning, and user-centric design to improve the accuracy and versatility of sign language translation and detection.

## 5.1 Challenges

This study faced challenges across its five project phases, including prototype development, software development, data collection, model training, and real-time prediction.

In the **Prototype Development** phase, selecting appropriate components like flex sensors and Arduino boards was challenging due to factors like compatibility, cost, and robustness. The power supply management was essential to ensure stable and continuous power for reliable data collection.

In the **Software Development** phase, unwanted warning messages during model training were addressed to reduce distractions. Important warnings related to optimizer performance were resolved by adopting the Legacy Adam optimizer.

The **Data Collection** phase presented challenges in recruiting willing participants who could commit time, participant unfamiliarity with sign language, and the need for comprehensive training to ensure accurate and consistent gestures. Expanding the dataset was vital for robustness.

**Model Training and Evaluation** involved fine-tuning the model to enhance accuracy. Initial training phases had low accuracy, but through optimization, accuracy improved significantly. Data collected from the MPU-6050 was not used due to a lack of intricate hand movements.

**Real-Time Prediction** was the ultimate objective, where the technology's effectiveness was tested in real-world scenarios. Iterative refinement helped troubleshoot issues. Transitioning from sign language to text and speech expanded the technology's capabilities for a more inclusive communication experience.

### 5.2 Limitations

This project, while achieving significant milestones, is not without limitations, which highlight areas for potential future research and development.

- **Cross Validation**: The prototype lacks a cross-validation process, which could enhance the model's robustness by assessing its performance across different data subsets.
- **Internet Connection**: Real-time prediction relies on internet connectivity, which may limit usability in areas with poor connectivity.
- **Unilateral Right-Hand Prototype**: The system is optimized for detecting sign gestures performed solely by the right hand, excluding left hand or two-handed gestures. A more comprehensive solution accommodating both hands is needed.
- **Restricted to Finger Movements**: The current configuration primarily captures finger movements and may not detect gestures involving broader hand movements. Expanding its capabilities to encompass varied hand configurations is necessary.
- **Physical Discomfort**: The prototype's wires and components may cause discomfort during use. Efforts to minimize physical inconvenience through design and wireless connectivity should be explored.
- **Additional Devices for Varied Ages**: Users of different age groups may require different devices due to hand size differences. A tailored approach, such as adjustable components or various glove sizes, could address this issue.
- **Prototype Aesthetic and Social Acceptance**: The prototype's appearance may not align with user preferences, potentially affecting adoption, especially among younger users.
- **Lack of Two-way Communication**: The technology currently enables one-way communication from hearing-impaired individuals to others but doesn't facilitate two-way communication. Future research could explore ways to bridge this gap, allowing for more interactive dialogue.

## 6. Conclusion

This research introduces an innovative strategy centred on certain system goals, which represents a substantial advancement in improving communication for people with hearing impairment. Recognition of complex hand and finger movements, recognition, and translation of American Sign Language (ASL) gestures in real time with primary focus on alphabet from A to E and numbers from 0 to 9 were the objectives of the project which was achieved successfully. The system's unique feature is its simultaneous translation of sign language into spoken language and text, which is further enhanced by the addition of a text-to-speech engine for smooth output.

The prototype development phase involved creating a smart glove using resistive flex sensors and an Arduino Mega 2560 microcontroller. It faced challenges in component selection and power supply management. The software development phase focused on selecting an RNN model with LSTM architecture for sign language recognition. Challenges included managing warning messages and model optimization. Data collection was pivotal, requiring transparent communication with participants and their training. Collaboration with university staff expanded the dataset. Model training and evaluation involved continuous refinement, resulting in an accuracy rate exceeding 80%. Real-time prediction marked a significant milestone, though challenges persisted in aligning training accuracy with real-time performance.

The study revealed the system's adaptability to individual signing styles and its potential to function in diverse real-world environments. The implications and applications are extensive, including enhanced communication accessibility, improved education, contributions to human-computer interaction, assistive technology, societal integration and inclusion, and technological innovation.

Future research and development opportunities include comprehensive hand gesture detection, capturing whole-hand movements, designing a wireless and wearable smart glove, multilingual support, real-time two-way communication, improved performance in various situations, integration with communication devices, accessibility for special needs education, and addressing ethical and social considerations.

## 7. REFERENCES

Ahmed, M.A., Zaidan, B.B., Zaidan, A.A., Salih, M.M. and Lakulu, M.M.B., 2018. A review on systems-based sensory gloves for sign language recognition state of the art between 2007 and 2017. *MDPI,* p. 2208.

Ahmed, M.A., Zaidan, B.B., Zaidan, A.A., Salih, M.M., Al-Qaysi, Z.T. and Alamoodi, A.H., 2021. Based on wearable sensory device in 3D-printed humanoid: A new real-time sign language recognition system. *Measurement,* Volume 168, p. 108431.

Amazon-Breadboard, 2019. *Amazon.* [Online]
Available at: https://www.amazon.co.uk/dp/B07YJTFVB2?psc=1&ref=ppx_yo2ov_dt_b_product_details
[Accessed 23 Jun 2023].

Amazon-Flex, 2011. *Amazon.* [Online]
Available at: https://www.amazon.co.uk/Generic-Flex-Sensor-4-5/dp/B004TAA2PU/ref=sr_1_2?crid=12CIGJQUWJHZB&keywords=Flex+Sensor+by+Spectra+Symbol&qid=1686771565&sprefix=flex+sensor+by+spectra+symbol%2Caps%2C114&sr=8-2
[Accessed 17 June 2023].

Amazon-Mega, 2011. *Amazon.* [Online]
Available at: https://www.amazon.co.uk/dp/B0046AMGW0?psc=1&ref=ppx_yo2ov_dt_b_product_details
[Accessed 9 July 2023].

Amazon-MPU6050, 2019. *Amazon.* [Online]
Available at: https://www.amazon.co.uk/AZDelivery-MPU-6050-Gyroscope-Accelerometer-including/dp/B07TKLYBD6/ref=pd_day0fbt_img_sccl_1/258-6528343-1122901?pd_rd_w=siEQC&content-id=amzn1.sym.bfe06781-69ca-4a3a-9842-e535d2f2ae0e&pf_rd_p=bfe06781-69ca-4a3a-9842-e535d2f2ae0e
[Accessed 17 June 2023].

Amin, M.S., Amin, M.T., Latif, M.Y., Jathol, A.A., Ahmed, N. and Tarar, M.I.N., 2020. Alphabetical gesture recognition of American sign language using e-voice smart glove. *IEEE 23rd International Multitopic Conference (INMIC) ,* pp. 1-6.

Amin, M.S., Rizvi, S.T.H. and Hossain, M.M., 2022. A Comparative Review on Applications of Different Sensors for Sign Language Recognition. *Journal of Imaging,* p. 98.

Anupama, H. U. B. M. S. V. V. a. K. Y., 2021. Automated sign language interpreter using data gloves. *international conference on artificial intelligence and smart systems (ICAIS),* pp. 472-476.
Bantupalli, K. and Xie, Y., 2018. American sign language recognition using deep learning and computer vision. *IEEE International Conference on Big Data (Big Data),* pp. 4896-4899.
Bragg, D., Koller, O., Bellard, M., Berke, L., Boudreault, P., Braffort, A., Caselli, N., Huenerfauth, M., Kacorri, H., Verhoef, T. and Vogler, C., 2019. Sign language recognition, generation, and translation: An interdisciplinary perspective. *Proceedings of the 21st International ACM SIGACCESS Conference on Computers and Accessibility,* pp. 16-31.
Bukhari, J., Rehman, M., Malik, S.I., Kamboh, A.M. and Salman, A., 2015. American sign language translation through sensory glove; signspeak. *International Journal of u-and e-Service, Science and Technology,* pp. 131-142.
Cable, A.-J. W., 2017. *Amazon.* [Online]
Available at: https://www.amazon.co.uk/dp/B074P726ZR?ref=ppx_yo2ov_dt_b_product_details&th=1
[Accessed 17 June 2023].
Choudhary, D.K., Singh, R. and Kamthania, D., 2021. Sign language recognition system. *Proceedings of the International Conference on Innovative Computing & Communication (ICICC).*
Commons, W., 2015. *Wikimedia Commons.* [Online]
Available at: https://commons.wikimedia.org/wiki/File:Long_Short_Term_Memory.png
[Accessed 25 July 2023].
Dayal, A.; Paluru, N.; Cenkeramaddi, L.R.; J., S.; Yalavarthy, P.K. , 2021. Design and Implementation of Deep Learning Based Contactless Authentication System Using Hand Gestures. *https://doi.org/10.3390/electronics10020182.*
El-Din, S.A.E. and Abd El-Ghany, M.A., 2020. Sign Language Interpreter System: An alternative system for machine learning. *Novel Intelligent and Leading Emerging Sciences Conference (NILES),* pp. 332-337.
Farooq, U., Asmat, A., Rahim, M.S.B.M., Khan, N.S. and Abid, A., 2019. A comparison of hardware based approaches for sign language gesture recognition systems. *International Conference on Innovative Computing (ICIC) ,* pp. 1-6.
Ibrahim, N.B., Selim, M.M. and Zayed, H.H., 2018. An automatic Arabic sign language recognition system (ArSLRS). *Journal of King Saud University-Computer and Information Sciences,* pp. 470-477.
JIMBLOM, BBOYHO, 2016. *Sparkfun Start Something.* [Online]
Available at: https://learn.sparkfun.com/tutorials/flex-sensor-hookup-guide#example-program
[Accessed 21 June 2023].
kondreddy, I., 2023. *medium.com.* [Online]
Available at: https://medium.com/@issackondreddy/chapter-3-recurrent-neural-networks-unlocking-the-secrets-of-sequential-data-5905ab186730#id_token=eyJhbGciOiJSUzI1NiIsImtpZCI6ImMzYWZlN2E5YmRhNDZiYWU2ZWY5N2U0NmM5NWNkYTQ4OTEyZTU5NzkiLCJ0eXAiOiJKV1QifQ.eyJpc3MiOiJodHRwcz
[Accessed 22 July 2023].
LSTM, W. -., 2022-2023. *Wikipedia.* [Online]
Available at: https://en.wikipedia.org/wiki/Long_short-term_memory
[Accessed 24 July 2023].
Nimisha, K.P. and Jacob, A., 2020. A brief review of the recent trends in sign language recognition. *International Conference on Communication and Signal Processing (ICCSP) ,* pp. 186-190.
NV5 Geospatial - Calculate Confusion Matrices, 2023. *www.nv5geospatialsoftware.com.* [Online]
Available at: https://www.nv5geospatialsoftware.com/docs/CalculatingConfusionMatrices.html
[Accessed 20 July 2023].
Organization, W. H., 2023. *Deafness and hearing loss.* [Online]
Available at: https://www.who.int/news-room/fact-sheets/detail/deafness-and-hearing-loss#:~:text=Overview,will%20have%20disabling%20hearing%20loss.
Parton, B.S., 2006. Sign language recognition and translation: A multidisciplined approach from the field of artificial intelligence. *Journal of deaf studies and deaf education,* Volume 11, pp. 94-101.
Rastgoo, R., Kiani, K. and Escalera, S., 2021. Sign language recognition: A deep survey. *Expert Systems with Applications,* Volume 164, pp. 113-794.
Rizwan, S.B., Khan, M.S.Z. and Imran, M., 2019. American sign language translation via smart wearable glove technology. *International Symposium on Recent Advances in Electrical Engineering (RAEE),* Volume 4, pp. 1-6.
RNN, W. -., 2022-2023. *Wikipedia.* [Online]
Available at: https://en.wikipedia.org/wiki/Recurrent_neural_network
[Accessed 20 July 2023].

Rosero-Montalvo, P.D., Godoy-Trujillo, P., Flores-Bosmediano, E., Carrascal-García, J., Otero-Potosi, S., Benitez-Pereira, H. and Peluffo-Ordonez, D.H., 2018. Sign language recognition based on intelligent glove using machine learning techniques. *IEEE Third Ecuador Technical Chapters Meeting (ETCM),* pp. 1-5.
Saleh, N., Farghaly, M., Elshaaer, E. and Mousa, 2020. Smart glove-based gestures recognition system for Arabic sign language. *International Conference on Innovative Trends in Communication and Computer Engineering (ITCE) ,* pp. 303-307.
Salem, N., Alharbi, S., Khezendar, R. and Alshami, H., 2019. Real-time glove and android application for visual and audible Arabic sign language translation. *Procedia Computer Science,* Volume 163, pp. 450-459.
Sapkota, B., Gurung, M.K., Mali, P. and Gupta, R., 2018. Smart glove for sign language translation using Arduino. *1st KEC Conference Proceedings,* Volume 1, pp. 5-11.
scikit-learn.org, 2007-2019. *scikit-learn.org.* [Online]
Available at: https://scikit-learn.org/0.21/user_guide.html
[Accessed 14 July 2023].
Society, N. G., 2022. *National Geographic Education.* [Online]
Available at: https://education.nationalgeographic.org/resource/sign-language/#
Sparkfun, 2016. *Sparkfun.* [Online]
Available at: https://learn.sparkfun.com/tutorials/flex-sensor-hookup-guide
[Accessed 13 June 2023].
spectrasymbol, 2014. Flex Sensor. *https://cdn.sparkfun.com/datasheets/Sensors/ForceFlex/FLEX%20SENSOR%20DATA%20SHEET%202014.pdf.*
StartASL.com, 2008. *StartASL.com.* [Online]
Available at: https://www.startasl.com/american-sign-language-alphabet/
[Accessed 10 June 2023].
Suri, A., Singh, S.K., Sharma, R., Sharma, P., Garg, N. and Upadhyaya, R., 2020. Development of sign language using flex sensors. *international conference on smart electronics and communication (ICOSEC),* pp. 102-106.
T. Starner and A. Pentland, 1995. Real-time American Sign Language recognition from video using hidden Markov models. *Proceedings of International Symposium on Computer Vision - ISCV, Coral Gables, FL, USA,* Issue 10.1109/ISCV.1995.477012., pp. 265-270.
WIKIBOOKS, 2021. *WIKIBOOKS.* [Online]
Available at: https://en.wikibooks.org/wiki/Sign_language