Sign Language Interpreter System: An alternative system for machine learning

Salma A. Essam El-Din[1] and Mohamed A. Abd El-Ghany[2,3]
[1]*Communication Engineering Department, German University in Cairo, Cairo, Egypt*
[2]*Electronics Department, German University in Cairo, Cairo, Egypt*
[3]*Integrated Electronic Systems Lab, TU Darmstadt, Germany*
E-mails: {salma.essameldin, mohamed.abdel-ghany}@guc.edu.eg

***Abstract*—Losing the ability to speak exerts psychological and social impacts on the affected people due to the lack of proper communication. Thus, Sign Language (SL) is considered a boon to people with hearing and speech impairment. SL has developed as a handy mean of communication that form the core of local deaf cultures. It is a visual–spatial language based on positional and visual components, such as the shape of fingers and hands, their location and orientation as well as arm and body movements. The problem is that SL is not understood by everyone, forming a communication gap between the mute and the able people. Multiple and systematic scholarly interventions that vary according to context have been implemented to overcome disability-related difficulties. Sign language recognition (SLR) systems based on sensory gloves are significant innovations that aim to procure data on the shape or movement of the human hand to bridge this communication gap, as the proposed system. The proposed model is a glove equipped with five flex sensors, interfacing with a control unit fixed on the arm, translating American Sign Language (ASL) and Arabic Sign Language (ArSL) to both text and speech, displayed on a simple Graphical User Interface (GUI). The proposed system aims to provide an affordable and user friendly SL translator system, working on the basis of Machine Learning (ML). However, it adapts to each person's hand instead of using a generic data set. The system achieved 95% recognition rate with static gestures and up to 88% with dynamic gestures.**

***Index Terms*—American Sign Language (ASL), Arabic Sign Language (ArSL), Sign Language Recognition (SLR), sensor glove**

I. Introduction

The ability to speak or the power of speech is something we take for granted. It is the most effective and powerful way of sharing thoughts and emotions. It is the bridge that facilitates the communication with others. However, a large fraction of the world's population is not blessed with this ability. Hearing loss, also known as deafness, is one of the most common disabilities worldwide. Deafness is a partial or total inability to hear in one or both ears, it could be permanent or temporary. Deafness is defined as a degree of hearing loss such that a person is unable to understand or process speech even in the presence of amplification. According to the World Health Organization (WHO) statistics, around 466 million people worldwide are deaf, 34 million of those are children. The WHO also estimated that over 900 million people will have this disability by 2050 [1]. There are several causes for hearing loss: genetic causes, complications at birth, certain infectious diseases, chronic ear infections, the use of particular drugs, exposure to excessive noise and ageing. Loss of the ability to speak or hear exerts psychological and social impacts on the affected persons due to the lack of proper communication. This communication barrier adversely affects the lives and social relationships of deaf people. Sign Language (SL) is definitely a boon to deaf and mute people for communicating in daily life. Using any SL like American Sign Language (ASL), Arabic Sign Language (ArSL) or any other, a person can convey messages by the help of movement of the hands rather than sound patterns. SL involves simultaneously defined shapes, orientation and movement of the body parts. The main problem is that the majority of healthy people have little to no understanding of sign languages. Thus, effective communication is considered a challenge and an obstacle for the deaf and mute in their daily lives. This proposed work is an attempt to overcome the communication barrier and discomfort in the society by providing the deaf people with a method of communication so that they can share their thoughts and feelings effectively and independently without the need for a translator. If on one side a deaf person communicates using the SL that he is familiar and comfortable with, and then the system translates this SL into sound and visuals, which the able person can understand, this would be an effective solution bridging the gap between the two different means of communication. This proposed work aims to provide this solution with the least cost and the most effective user friendly way. The proposed model is a glove equipped with five flex sensors, interfacing with an Arduino Mega as the control unit, translating two different sign languages: ASL and ArSL to both text and speech on a simple Python Graphical User Interface (GUI). The system works similar to Machine learning (ML) but without the need of a generic data set.

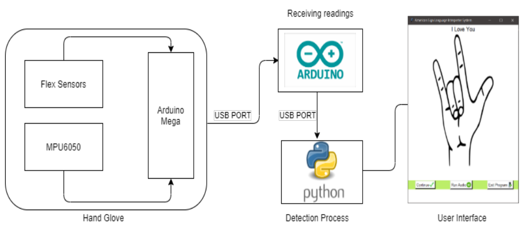

Fig. 1. Block diagram of the proposed system.

978-1-7281-8226-1/20/$31.00 ©2020 IEEE

This paper is organized as follows: Section II provides the previous related work. In section III, detailed information about all the hardware and the associated technologies used is discussed and all the steps taken during the implementation of the project is covered in order. Followed by, section IV that includes the testing process and the results. And finally Section V provides the conclusion and future suggested work.

II. Literature Review

Extensive research has gone into the development of technology to achieve gesture to speech conversion which can be done using one of three approaches: vision based, sensor-based or hybrid-based which is a mixture between vision-based and glove-based systems but it is not widely used. These systems are adopted to capture hand configurations needed for the Sign Language Recognition (SLR) [2]. Sensors-based systems are basically just sensors mounted on certain type of gloves used to collect input data for the system to start the recognition process. Various types of sensors are used to measure the bend angles for fingers, the abduction between fingers and the orientation of the wrist. The major advantage of glove-based systems over vision-based systems is that gloves are more straight forward. They eliminate the need to process raw data into meaningful values by directly reporting relevant and required input data in terms of voltage values or resistances as in flex sensors' case to the computing device. But on the other hand, vision-based systems need to apply specific tracking and feature extraction algorithms to raw video streams, increasing the computational overhead [2]. Finger bending detection sensors are claimed to be the dominant category of sensors as the main movement in SL related to the fingers is bending [2]. Flex sensors are the most used by researchers and developers according to the previous papers. A flex sensor determines the amount of finger curvature, based on resistive carbon elements. Many previous works used flex sensors with hand orientation sensors to convert SL to text and speech. For example in [3], the proposed system translates the Pakistan Sign Language (PSL) to speech with the help of flex sensors, accelerometer, gyroscope and contact sensors, achieving an efficiency of 93.4% with alphabets and numbers. A similar model was proposed in [4], a hand glove along with some sensors and circuitry placed on the arm of the deaf person, aiming to detect the change in gestures and convert them into text or speech. In [5], the authors used the same glove model with ML to detect ASL, then display the gesture on an android application with accuracy 94.23%. Last but not least, authors in [6] proposed a simple model of five flex sensors in combination with the gyroscope leading to a successful and accurate translation of ASL to speech. In addition to using Python to build a reliable and accurate system for the messaging service to make the system user friendly. Some of the other used sensors in SLR are: Optical sensors which are electronic detectors that convert light, or a change in light into an electronic signal, they are used to measure the angle of the finger curvature in order to determine its shape by the amount of light passing through the channel which depends on the optic technology [2], so that when the finger is straight, the density of the received light will be very significant. The advantage of this sensor is that it is suitable for handicapped individuals whose fingers can barely perform even very small motions. Tactile sensor is a robust polymer-thick film device whose resistance changes when a force is applied, so by calculating the amount of force placed on the finger, it allows us to determine whether the finger is curved or straight [7]. Capacitive touch sensors have emerged as one of the preferred ways for detecting SL. It is composed of two terminals that function as emitting and receiving electrodes. In [8], the proposed model consisted of eight sensors, on the tip of each finger and between the finger joints in order to recognize the posture of individual finger segments, to have more precise reading and with a Raspberry Pi as the processing unit. This model benefits from the touch sensor's digital advantage to translate all of the 26 alphabets and 10 digits of ASL with an accuracy over 92%. Vision-based hand gesture recognition is an area of active current research in computer vision and ML. It requires fast and extremely robust hand detection and gesture recognition in real time [9]. Those systems use cameras as primary tools to obtain the necessary input data, MATLAB has been commonly used to perform image processing by using image segmentation and with the help of feature extraction algorithms, the system recognizes the image of the hand gesture captured by the camera and converts it to speech from a list of pre-recorded tracks. After image processing, ML and Template Matching can be used to get the desired text to speech conversion. Image processing is an umbrella term for many functions that analyze images or convert one representation of an image into another. Nowadays image processing is done in the digital domain replacing the analog processing used in the past. In 2007, an automatic vision-based system with a customized colour glove was presented, where the colour eases the extraction process [10]. At the end, an Artificial Neural Network (ANN) is used for classifying and translating gestures having an overall recognition rate over 90%. In 2008, [11] used another approach based on the skin colour, indicating that it can be used in the segmentation process for separating the human from a cluttered background. There is plenty of previous work that used this same approach but they only differ in the final recognition accuracy. Although vision-based systems have a nearly perfect recognition accuracy, their limitations cannot be dismissed. The need of high specification camera, limited field of view of the capturing device, high computational costs and the need for multiple cameras to obtain robust results are the main limitations to such systems [2].

III. Methodology

A glove-based recognition system is composed of three main units: input, processing and output units. The MPU6050 and the flex sensors represent the input unit, giving all the required input data to the processing unit. This data is then processed using the Arduino Mega micro-controller board. Upon reaching the output unit, it displays the detected gestures on a GUI using Python3. The implementation process includes three steps: hardware design, communication between Arduino and Python and finally detecting and displaying the gestures. Figure 2 shows the schematic view of the proposed model.

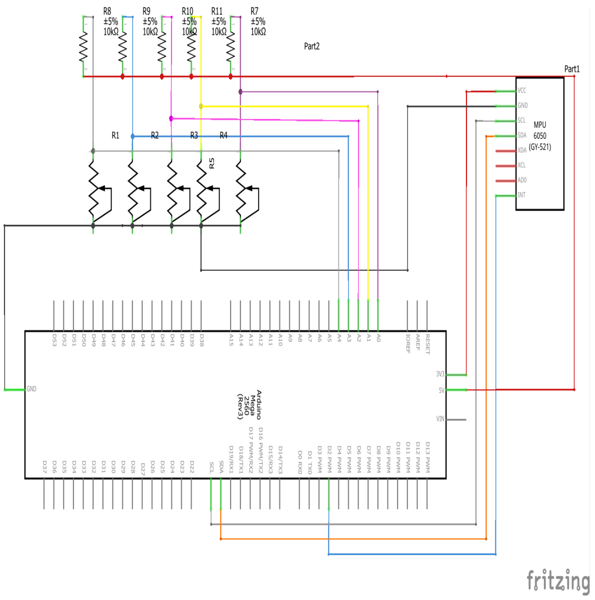

Fig. 2. A schematic diagram of the proposed glove.

A. Flex Sensor

Flex sensor is a type of bending sensors used to measure the amount of defection and bending. There is a direct relation between the resistance of the flex sensor and the amount of bending so it can also be employed like a potentiometer [2]. It is a two-terminal device, that does not contain any polarized terminals similar to diodes or capacitors, which means there is no positive and negative terminals. The first terminal is generally connected to the positive terminal of the power supply and the second is connected to the GND pin of the power source. The proposed model includes five 2.2-inch flex-sensors for each finger, mounted on the glove. The sensor's working bending strip principle is straight forward, whenever the strip is twisted then its resistance will be changed and this can be measured easily with the help of a micro-controller. Its operating voltage ranges from 0V to 5V and it can function on low-voltages. Moreover, the flat resistance is 25K ohm, with 30 percent tolerance and resistance bend range from 45K -125K ohm (0-1023). It was chosen to be the main player in this model because it is affordable, easy to integrate, does not require complicated signal processing and can be easily attached to fabric without adding volume. However, flex sensors tend to have few drawbacks, including time-dependent creep and nonlinear relationship between curvature and resistance change which prevents them from measuring large curvatures and the absolute angles of objects. They are also limited in length and cannot measure a bend change for a large area.

B. MPU6050

MPU6050 is a Micro-Electro-Mechanical System (MEMS) consisting of a three-axis accelerometer and a three-axis gyroscope. It is used to measure velocity, orientation, acceleration, displacement and other motion related features. It ranks first among the world's six dimension motion tracking devices [2]. It has the ability to process nine-axis algorithms as it captures motion in the three axes: X, Y and Z at the same time due to the availability of a 16-bit Analog to Digital Converter (ADC). It also includes dynamic multi-pathing (DMP) which has the ability to solve complex calculations. The proposed model includes an MPU6050 which is mounted on the bottom left of the glove, near the wrist. Its operating voltage ranges between 3V and 5V. SCL is the pin responsible for clock pulse synchronization for I2C communication while SDA is used for the transferring of data through I2C communication. It uses the I2C communication protocol while interfacing with Arduino which is simply defined as a bidirectional two-wire bus, used to send data between integrated circuits.

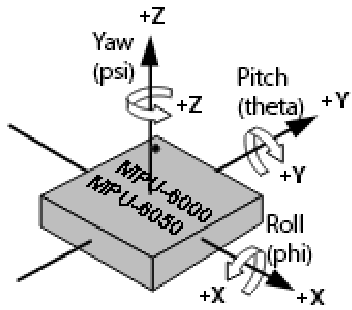

Fig. 3. Illustration for yaw, pitch and roll.

C. Arduino Mega

Arduino Mega 2560 is a controller board based on the Atmega 2560 micro-controller. It has more memory compared to other Arduino boards, enough to store the required code. The provided USB port is used to upload the code from Arduino IDE to the board and also in the communication between Arduino IDE and Python. Arduino Mega is one of the most common and used micro-controllers in the SLR field. Figure 2 shows the circuit implementation. The two pins supporting the I2C communication between the board and the MPU are: pin number 20 (SDA) which is used for holding the data and pin number 21 (SCL) that is used for providing data synchronization between the devices. Arduino Mega is the mastermind that processes the input data from the gloves' sensors to the interpreter system and it is fixed on the arm of the user using an elastic band. Figure **??** shows all the used components during the implementation of the hardware, their price, their quantity and the final cost of the prototype.

Product	Price	Quantity	Total
Flex Sensor (2.2-inches)	EGP 225.00	5	EGP 1,125.00
MPU6050	EGP 95.00	1	EGP 95.00
Arduino Mega (Atmega 2560)	EGP 260.00	1	EGP 260.00
iGlove	EGP 150.00	1	EGP 150.00
Female to female (40 jumpers set)	EGP 22.00	1	EGP 22.00
Female to male (40 jumpers set)	EGP 22.00	1	EGP 22.00
Male to male (40 jumpers set)	EGP 22.00	1	EGP 22.00
			EGP 1,696.00

Fig. 4. A table representing the cost analysis of the proposed prototype

D. Arduino Software

After connecting the hardware, Arduino IDE was used to program the board. It is an open source software that is used for writing and compiling the code into the Arduino module. The first section of the code includes declaration of the necessary sensors variables, the required libraries to the program and also declaration of the used analog pins for the flex sensors. The MPU6050 is a little more complicated, different variables should be declared to run the program such as: control or status variables and orientation and motion variables. Followed by the setup section which includes statements that lay the foundation for actions that happen later on in the program like: joining the I2C bus and setting the I2C clock, initializing the serial communication between Arduino and Python as well as choosing the baud-rate. The baud-rate is chosen based on the requirements of the project, verifying the connection, loading and configuring the DMP, setting the gyroscope x, y and z offsets to 220, 76 and -85 respectively. These offsets are chosen as they are the common values scaled for minimum sensitivity. Also the accelometer Z offset is set to the default value 1788. These values were founded on [12] and tested on the module before starting the project. Whenever the calibration of the MPU is done, the DMP is enabled to give a sign that it is ready to be used. In the proposed work, baud-rate 115200 is chosen as it suits working with the flex sensors. The code consists of two parts: part responsible for the MPU and one for the flex sensors. This section goes as following: First, checking if the programming of the setup failed or if the DMP is not ready, reading a packet from the FIFO buffer if it is not empty, the buffer is then checked to be able to display the following MPU output readings: quaternion values in easy matrix form (w, x, y and z) and Euler angles in degrees. Euler angles are used to describe the orientation of a rigid body with respect to a fixed coordinate system. Moreover, the most important outputs for the SLR glove are the yaw, pitch and roll. Their definition is commonly explained using the Aircraft Principal Axis [12]. Vehicles that are free to operate in three dimensions can change their attitude and rotation about the three orthogonal axes: the longitudinal, vertical, and horizontal axes, centered at the vehicle's center of gravity. Roll is defined as the motion around the longitudinal axis. While the motion about the perpendicular axes is called yaw. And finally pitch is the motion about the lateral axis which measures how far an object is tilted up or down. Figure 3 illustrates their definitions. The detection of the dynamic gestures is based on these directions. Flex sensors' values are then read and displayed on the serial monitor after running the program. At last, the delay of this program is set to 1000ms which is used to stop the program from moving to the next line of code when it encounters this function until the delay time is finished.

E. Python

While working on this model, it was necessary to send the output data from the Arduino to a processor that can interpret and display the data as desired. A high level programming language like Python was needed to build a GUI through which the user would interact with the system and be able to listen to the words or letters. Platforms like Arduino work well with Python, especially for applications that require integration with sensors and other physical devices. Python is a versatile, easy to learn and easy to use scripting language, its huge library of user-created modules makes it an ideal language for a wide verity of computer side tasks [13]. Basically, Arduino sends the required data to Python through serial communication using the USB port. A simple code is written on both sides of to commence the communication. The serial communication was already initialized in the setup section in Arduino IDE, with 115200 baud-rate. On Python's side, few steps should be done to start the communication: Installing "PySerial" which is a library that provides support for serial communication and importing the serial module in Arduino. The baud-rate used by the Arduino board should be the same as the one Python uses to initiate the connection. After that, the data is received, cleaned up and printed. The data is first read and stored as a string variable then stored in suitable integer variables and printed. As for the output, Python offers multiple options for developing GUI. Tkinter is the most commonly used out of all the Python GUI packages. It is a standard Python interface to the Tk GUI toolkit shipped with Python. Combining Python with Tkinter provides a fast and easy way to create GUI applications as it provides a powerful object-oriented interface to the Tk GUI toolkit. It was chosen to be the GUI package used in this work to build a user interface that works as an ML alternative system. The system works on any Personal Computer (PC) running any operating system like Windows, Linux and MacOS, with no hardware limitations except for the presence of a USB port and Python3 installed. The proposed system is capable of translating the 26 alphabets of ASL and the 28 alphabets of ArSL to both text and speech, in addition to few words and sentences such as: Okay, good job, stop, I wish you a long life, I really love you, no, هذا رهيب, أحبك, etc. The provided voice in this proposed work is a collection of pre-defined and pre-recorded audio clips downloaded from https://ttsmp3.com/. Figure 5 show some of the available letters in ASL and ArSL.

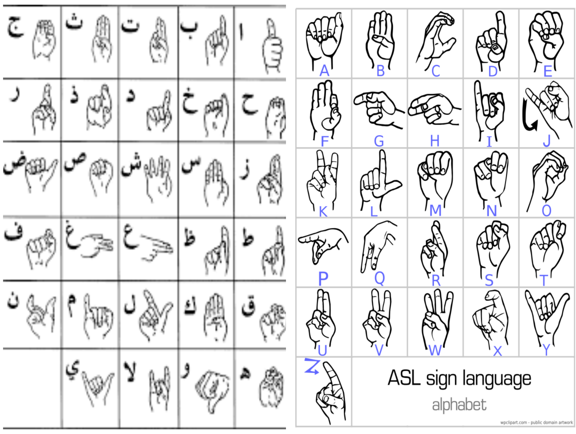

Fig. 5. ArSL (left) and ASL (right) available letters.

The idea behind the proposed system is similar to ML but it works per user without the need for a data set. Its aim is to be able to detect SL using different hand sizes. As the flex sensors readings change with the hand size and posture, therefore this is the point targeted by this program. The system has two sides: The front-end which is accessible by the user on the screen and the back-end which is the side responsible for storing and manipulating the data that is not directly accessed by the user. First, the flow of the user interface is as following:

1) The user should sign up to have an account that includes a username and a password to be able to log in using it later.
2) Choosing the SL used by the user, either ASL or ArSL. In case ArSL is chosen, the user has the option to choose the language of the user interface, as the user interface is available in both English and Arabic.
3) Checking the instructions of the system. These instructions are provided to help the user understand the required steps to be followed.
4) The user can also check the dictionary that includes all the available alphabets and words of the chosen SL. This part is also added in case a deaf person does not have a SL background, so it can be used to teach them first. All the words or alphabets in the dictionary are provided with a picture.
5) The user should then start doing all the required alphabets and words according to the given instructions, which is considered as the "Learning Process" for the system. This process is done only once when the user signs up for the system and it takes about thirty minutes to complete.
6) Whenever the user is done with the learning process, the system will be ready to detect the SL and translate it to text and speech. The detected gesture appears on a window that contains: a picture of the detected sign, a label mentioning it and an audio button. The user can use the program and keep communicating as many times as he wants.

On the back-end, the system is programmed as follows:

1) The serial communication between Arduino and Python is commenced.
2) During the sign up, the user is asked to enter information about himself and all the entries are then saved in a CSV file. This helps the developer to know more about the users' age group and regions, for example.
3) During the learning process, the user provides the program with the necessary sensors' readings. The program takes the readings of each sign from the user 20 times and stores them in another CSV file with the user's name.
4) Those saved readings of each user are then processed to calculate the range of each gesture, using the readings from the five flex sensors and the MPU. The range is then calculated by going through all the 20 rows of a specific sign and finding the maximum and minimum values to set the range for each gesture and store it in another CSV file with the name of the user as well, this represents the trained model.
5) Whenever the user starts communicating, an implemented function receives the current flex sensor values from Arduino and searches in the previous CSV file for the range that they lie in to detect the gesture. In addition, the proposed system finds the best fitting letter according to the current sensor readings. Basically, it outputs the gesture with the best fit even if one of the sensor values is offset from the range.

Fig. 6. The final prototype

IV. Experiment

A. Experimental Setup

A total of 10 volunteers were recruited for the primary experiment to collect the sensors data variations according to the various hand sizes and shapes. The participants did not possess any muscular disorder that could affect the gestures. They had no prior knowledge of SL and non of them had hearing or speech impairment. The volunteers were requested to put on the glove and use the system using the provided instructions in order to be able to test the simplicity of the user interface as well as the functionality of the glove.

B. Experimental Results

Multiple iterations of calibration tests were organized to attain maximum efficiency. The testing process was divided into two parts: one for ASL and the other for ArSL. Since most of the ArSL gestures are static and quite different, the ArSL showed remarkable results and a higher recognition accuracy than ASL. However, a mismatch occurs with gestures with high similarity like: (letter ن and the word يؤلم كثيرا), (letter أ and the word عمل جيد), (the word توقف and the letter س), (letters س and ش), (letters U and V) and (letters A, S, E and T). As static gestures with high similarity give approximately the same sensors' readings so the system cannot differentiate between them leading to a mismatch. All the other static gestures had over 95% recognition rate. The detection of dynamic gestures differed from one user to another. As it depends on the speed and accuracy of the posture, where the same speed should be maintained during the learning and detection processes, otherwise, the system is not capable of detecting the dynamic gestures correctly. This basically requires the user to replicate an equal speed or at least close to the speed they used to teach the system. That's why dynamic gestures like letters (J and Z) and the words (No) and (لا) had lower accuracy, ranging from 80 to 88%. Figure 7 shows the detection process and some of the gestures with high similarity.

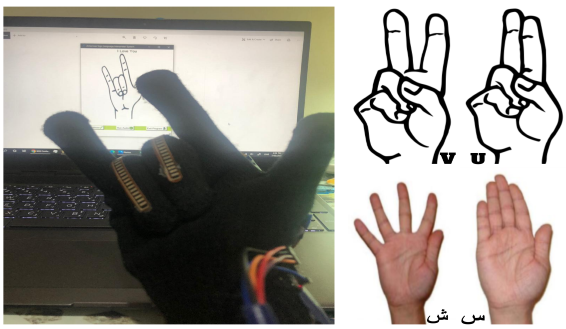

Fig. 7. The detection process and the high similarity between some gestures

V. Conclusion

An alternative system for ML which is capable of translating both ASL and ArSL alphabets and few words to text and speech, with the help of various sensors and a control unit mounted on a hand glove, displayed on a simple user interface using Python is presented. The main objective is to detect the change in gestures and convert them into human understandable form to bridge the communication gap between the deaf and the able people. The proposed model succeeded in reaching the target of replacing the necessary data set of ML with a more flexible system that works per user using a similar methodology, reaching 95% recognition accuracy with static gestures and up to 88% with dynamic ones. The recognition rate depends on the user's accuracy to replicate the gesture during learning and detection processes. Taking into consideration the economic condition as the system is made from low-cost modules without compromising the system performance. There are few reasons for the wrong detection of some of the words and letters as the small size of the glove, the lack of accuracy while doing the gesture, human error or even errors from the sensors while providing the readings. There are multiple suggested improvements as using contact sensors between the fingers, on the tips of each finger, to be able to avoid the mismatch between letters that have same orientation and sensors readings. Moreover, a left-hand glove could be added to be able to detect even more words using both hands. A bigger size glove would be preferably used to suit all people not only kids and female and also using 4.4-inches flex sensors to be able to cover all the glove's fingers completely to have more accurate readings. Lastly, the systems could also be modified to translate other sign languages and a feature of translating from one language to another can also be added, so that the deaf community can easily communicate with people around the world.

References

[1] W. H. Organization, "Deafness and hearing loss." https://www.who.int/news-room/fact-sheets/detail/deafness-and-hearing-loss, 2020.

[2] A. A. Z. M. M. S. a. M. M. b. L. Mohamed Aktham Ahmed, Bilal Bahaa Zaidan, "A review on systems-based sensory gloves for sign language recognition state of the art between 2007 and 2017," *Sensors (Basel)*, vol. 18, 2018. PMID: 29987266.

[3] T. M. Hina Shaheen, "Talking gloves: Low-cost gesture recognition system for sign language translation," *2018 IEEE Region Ten Symposium (Tensymp)*, pp. 737–742, 2018.

[4] I. J. M. A. W. Arslan Arif, Syed Tahir Hussain Rizvi and M. R. Shakeel, "Techno-talk: An american sign language (asl) translator," *2016 International Conference on Control, Decision and Information Technologies (CoDIT), Malta*, pp. 665–670, 2016.

[5] M. S. Z. K. Shaheer Bin Rizwan and M. Imran, "American sign language translation via smart wearable glove technology," *2019 International Symposium on Recent Advances in Electrical Engineering (RAEE)*, vol. 4, pp. 1–6, 2019.

[6] A. C. Vaibhav Mehra and R. R. Choubey, "Gesture to speech conversion using flex sensors, mpu6050 and python," *International Journal of Engineering and Advanced Technology (IJEAT)*, vol. 8, no. 6, p. 2249 – 8958, 2019.

[7] A. M. Vijayalakshmi P., "Sign language to speech conversion," *2016 International Conference on Recent Trends in Information Technology (ICRTIT), Chennai*, pp. 1–6, 2016.

[8] L. C. F. Q. K. S. Abhishek and D. Ho, "Glove-based hand gesture recognition sign language translator using capacitive touch sensor," *2016 IEEE International Conference on Electron Devices and Solid-State Circuits (EDSSC)*, 2016.

[9] F. R. Paulo Trigueiros and L. P. Reis, "Vision-based portuguese sign language recognition system," *Advances in Intelligent Systems and Computing*, vol. 275, 2014.

[10] P. L. O. M. A. Rini and K. Y. Chow, "Real-time malaysian sign language translation using colour segmentation and neural network," *Instrumentation and Measurement Technology Conference, Warsaw, Poland*, 2007.

[11] Y. S.-w. S. M.-h. T. Wen-kai, L. Chung-chi and S.Ching-lung, "Adaptive motion gesture segmentation," *International Conference on Embedded Software and Systems Symposia (ICESS2008),England,UK*, 2008.

[12] MertArduino, "Arduino - mpu6050 gy521 - 6 axis accelerometer + gyro (3d simulation with processing)." https://www.instructables.com/id/Arduino-MPU6050-GY521-6-Axis-Accelerometer-Gyro-3D/.

[13] "About python." https://www.python.org/about/.