Proceedings of the Third International Conference on Smart Systems and Inventive Technology (ICSSIT 2020)
IEEE Xplore Part Number: CFP20P17-ART; ISBN: 978-1-7281-5821-1

Convolutional Neural Network based Bidirectional Sign Language Translation System

1st Lance Fernandes
Department of Electronics Engineering
Sardar Patel Institute of Technology
Mumbai, India
lance.fernandes14@gmail.com

2nd Prathamesh Dalvi
Department of Electronics Engineering
Sardar Patel Institute of Technology
Mumbai, India
dalviprathamesh30@gmail.com

3rd Akash Junnarkar
Department of Electronics Engneering
Sardar Patel Institute of Technology
Mumbai, India
akashjunnarkar13@gmail.com

4th Professor Manisha Bansode
Department of Electronics Engineering
Sardar Patel Institute of Technology
Mumbai, India
manisha_bansode@spit.ac.in

***Abstract*—A considerable amount of gap in communication exists amongst the speech and hearing-impaired individuals with the other people; which is of paramount importance to be bridged. The aim is to study various methods for effective intercommunication between Sign language and the English language. Initially, a Hardware glove is implemented which has of flex sensors whose accuracy is proved to be not very high. To further improve the accuracy, a model using a convolutional neural network was trained on an existing data-set. Since the data-sets were not versatile and the scope was narrow, a new diversified data-set was created and the model was further improved. The new model has very high accuracy and it can predict almost every alphabet. Various other gestures having facial features and gestures including both the hands were added to our data-set. This model has a huge potential as it can interpret any gesture of various sign languages if provided in the data-set. The user can also add extra gestures in the data-set, making it highly customized. Further, the data is sent to an application which will convert the received text to speech. To reduce the communication gap, the system is made wholly bidirectional i.e. speech can also be converted to the sign language. Initially, the speech is taken at the input and is converted to the text which acts as the input for the next step in which the converted text is directly taken as the input to be converted to the corresponding gesture according to the convenient sign language. Thus, the input speech is translated into a video consisting of a sequence of gestures of the American sign language which can be extended to other languages as well. Bidirectional Sign Language Translating system consists of a software system. It is named as a bidirectional system as it not only converts the sign language to speech via text conversion but also incorporates a system which translates the speech to the prescribed sign language with text conversion as the mediator. The methodology has been explained in the further sections.**

Keywords— Sign Language Translation, Convolutional Neural Network, American Sign Language, Alphabet, Sign-to-Speech, Speech-to-sign, Hearing-impaired, Speech.

I. INTRODUCTION

A language which incorporates the implementation of hand movements, body orientation and facial expressions for communication; without relying on acoustic waves is known as a sign language. As per a report published by the World Health Organization mentioned in [1], "a total of 466 million individuals globally have a hearing impairment and this number is predicted to rise to 900 million individuals in another 30 years, by the year 2050. Furthermore, current estimates suggest an 83 percent gap in the aid available i.e. only 17 Percent of the those who require can use it." Thus, now, it is more than just necessary to create an economical and comprehensive system to bolster these individuals. Inability to communicate not only affects their smooth functionality but also eventually isolates them from others. This isolation takes a toll on their confidence, and specifically for students, might also be a barrier in fostering their development. Thus, this system aims to not only bridge the communication gap but to also enhance the learning curve of the individuals and make them more confident. There are various data-sets and models available but none of them is proven to be very accurate because of inconsistency in data-set or the model. There have been hardware innovations in the past which have focused on phrase conversion, but this is not very useful as the scope is based on the number of phrases included in the data-set. whilst, in our proposed system, the translation of each alphabet in the American Sign Language to text has been implemented.

II. LITERATURE SURVEY

In [13], a sensor-dependent gesture interpretation system has been proposed which implements flex sensors for sensing hand gestures. In the proposed system, the flex sensor is implicitly used to compute the angular tilt to which the finger is bent by noting down the change in resistor readings. The accelerometer is used to compute the angular degree to which the finger is tilted, taking into consideration a 3-dimensional model with all the three axes and the tactile sensor is incorporated in the system in order quantify the physical interaction amongst the fingers. The sensor system data is then sent to the micro-controller where it is compared with predefined values, further, it is then sent to be converted to speech from text, by making use of the Hidden Markov Model (HMM) which is used to computationally generate speech in the English language. In the method proposed in [7], the images are captured and converted to a series of

Authorized licensed use limited to: Middlesex University. Downloaded on October 25,2020 at 14:54:39 UTC from IEEE Xplore. Restrictions apply.

RGB (Red Green Blue) pixels. Two classifiers: Raw Feature classifier and Histogram feature classifier are used. These two classifiers are then trained using Back Propagation neural network algorithm. The Precision, recall and F1 score is calculated for the model. The accuracy for the classifiers is 70 percent and 85 percent respectively, as aforementioned. The data-set used for training was limited to a few gestures. In the method proposed in [8], initially, the features from static images and videos are extracted. The face is removed from the image using Viola and Jones algorithm. HSV (Hue, Saturation and Value) segmentation is used to isolate and extract the binary image of the hand. The fingertips and center of gravity of the hand are tracked to interpret whether they are dynamic or static. To classify the model, Support Vector machine classifiers are used. A speech recognition model is used to translate the input speech to text. In [11], a system in which flex sensor and accelerometer is used for recognizing the gestures has been proposed. One flex sensor is deployed for the thumb and the little finger; and two flex sensors for remaining fingers of the hand. The specific values for the flex sensor and accelerometer are realized corresponding to the gesture made. The obtained values are sent to the micro-controller where it is compared with the predefined values which help in detection of the alphabet which is further transmitted to the android application for text to speech conversion. In [3], a method in which the author has used his data-set consisting of limited gestures having the most commonly used words has been proposed. Convolutional neural network (CNN) is used along with Recurrent neural network (RNN) to classify individual gestures and sequences of images. Inception Model was used to build the CNN. The accuracy with the SoftMax layer was 90 percent and the accuracy of the pool layer was around 58 percent. In [9], the input data i.e. the image is captured using a camera module which is then sent for image processing to eliminate noise and adjust the brightness and contrast as per user requirement. Following which, the image in the RGB format is converted to the YCbCr format ('Y' stands for the Luma component; Cb and Cr represent the Chromo components). HSV (Hue, Saturation and Value) segmentation is used to isolate and extract the image of the hand and specify the skin color margins to Hue and Saturation Values. Gray-scale images are then converted to Binary Images by specifying a threshold value. Blob detection is then implemented to differentiate between the concerned object and other regions; this categorizing is done based on brightness and color. Contour detection is done by implementing the convexity hull algorithm to identify and extract the hand image by creating boundaries around the palm and fingers. Finally, the distance of each fingertip from the center of the palm is computed, corresponding to which the number detection is performed. The research work mentioned in [4], focuses on word level conversion of sign language to text and further to speech specifically in the Indian Sign language. The process involves three main stages starting with the data preprocessing stage, moving on to the classification stage and ending with the speech synthesis stage.

The data preprocessing stage is further divided into steps: Inputting the Image, Background subtraction, blob analysis, filtering the noise, converting to grayscale and brightness and contrast adjustment. The classification stage involves the use of Haar Cascade classification algorithm to interpret the gesture, based on the training performed using 500 positive samples, 500 negative samples and 50 test image samples of each gesture. The concluding stage converts the text into speech. The accuracy comes out to be 92.68 percent accuracy. In the work mentioned in [6], a glove is created, consisting of flex sensors. The system consists of flex sensors, LCD, Accelerometer and a keypad. The motive of the project is not only to bridge the communication gap but to also develop a self-learning system where people can learn the American Sign Language. It consists of two modes, teaching mode and learning mode. In the teaching mode, a database is created by making different gestures and saving it in the EEPROM of the microcontroller. In the learning mode, the user wears the gloves and makes the gesture he/she wants to depict and tries matching it with the existing database. The LCDs how much more or less a finger should be bent to match the nearest gesture. In [10], the sign language interpreter proposed, uses a glove with sensors which interpret 10 letters of the ASL. The sensor used is an LED- LDR pair for gathering data from each finger and to differentiate the letters. The value from the sensor is given to ADC 10 of the MSP430G2553 microcontroller for analog to digital conversion. The digital sample is encoded in ASCII code and given to Zigbee module for transmission and at the receiver end, the received ASCII code is forwarded to the computer where the character is displayed and audio of character is played. The resistance of LDR (Light Dependent Resistor) decreases with an increased light intensity of LED and vice versa. The LED is kept at one end and LDR on the other end is kept along with the finger. When the finger is straight, maximum light from LED falls on LDR and thus the resistance of LDR is low and thus voltage is also low. When the finger is bent, the light intensity falling on the LDR decreases and thus resistance increases and the voltage also increases. In [5], the proposed system has the following steps: image cropping, data augmentation, splitting into sets and training on an inception model. In image cropping, to avoid unnecessary data to be detected, the image is cropped using Python script which is then further classified. Data augmentation is done by using cropping, scaling and flipping, it is done to ensure that the neural network is not limited to any specified type of images. A custom algorithm is used for splitting the images into testing, validation and training sets where the testing and validation percentages are used as arguments. The accuracy rate is over 90 percent and validation accuracy are over 90 percent.

Authorized licensed use limited to: Middlesex University. Downloaded on October 25,2020 at 14:54:39 UTC from IEEE Xplore. Restrictions apply.

III. METHODOLOGY

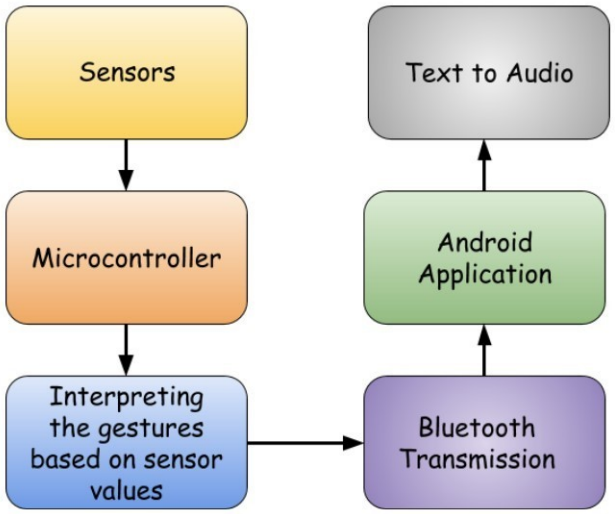

Figure 1: Block Diagram of The Hardware System

The flex sensor and accelerometer values on the glove change according to the gesture made. According to the flex sensor values, and accelerometer, the microcontroller, interprets the gesture-based on predetermined values of ranges for each gesture. This gesture which is interpreted as the text is sent to an android application created by us where the text is converted to speech. This entire process is demonstrated in figure 1.

A. Hardware Design

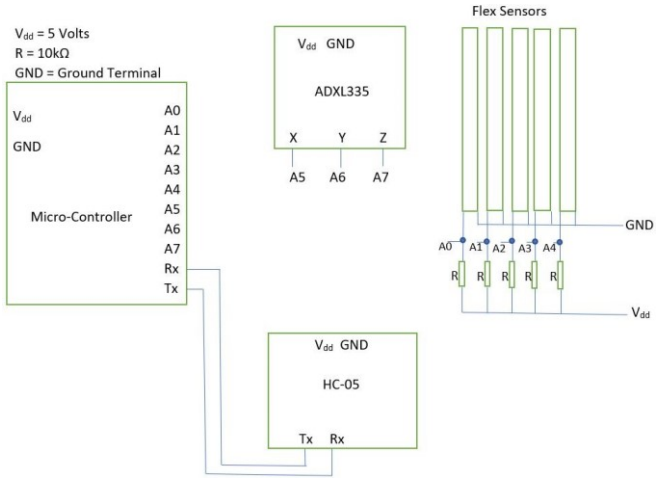

Figure 2: Circuit diagram of the Hardware.

1) Flex Sensor and Accelerometer: Flex sensor is a resistive element made up of carbon. It is cheap and economical and can operate on low voltages. It behaves like a variable resistor. The resistance values range varies from 50 kilo-ohms to 125 kilo-ohms. The size of the sensor is 2.2 inches. The flex sensors are connected to the analog pin of the microcontroller through a potential divider network having 10 kilo-ohms resistances each as shown in figure 2. The ADC (Analog to digital converter) present in the micro-controller is a 10- bit ADC. The output of the analog pin varies between 0 and 1024 according to the resistance. ADXL335 an accelerometer is used for detecting the axis which is important for improving the accuracy. It is small and compact. It has a low voltage rating. It has 3-axis sensing technology that is dependent on the motion along X, Y and Z direction. The 3 output values of the accelerometer vary between 0 to 1024. The output of the accelerometer is fed to the analog inputs of the micro-controller.

2) Bluetooth Module: To facilitate wireless communication, an HC-05 Bluetooth module has been used. It is small and compact and it is very economical. It has an encryption and security protocol which helps in the safe and secure transmission of the data. The transmission and receiver pin of the device is connected to the receiver and transmitter pins of the microcontroller respectively as shown in figure 2.

3) Micro controller and Application Development: Any micro-controller can be used for interfacing purposes. Arduino nano has been used as it is cheap and compact which is essential since it is attached to the glove. The flex sensor values for each sensor, corresponding to all the alphabets are recorded. The micro-controller is programmed according to these values. If the values of sensors fall between the range, the micro-controller will predict the alphabet and send it to the android application via Bluetooth. An android application is developed using the MIT App Inventor. The main aim of the application is to receive the data via Bluetooth and convert the text to speech which can be comprehended. Shown in figure 3 is the User Interface while Figure 4 is the hardware implemented.

Figure 3: GUI of the android application

Authorized licensed use limited to: Middlesex University. Downloaded on October 25,2020 at 14:54:39 UTC from IEEE Xplore. Restrictions apply.

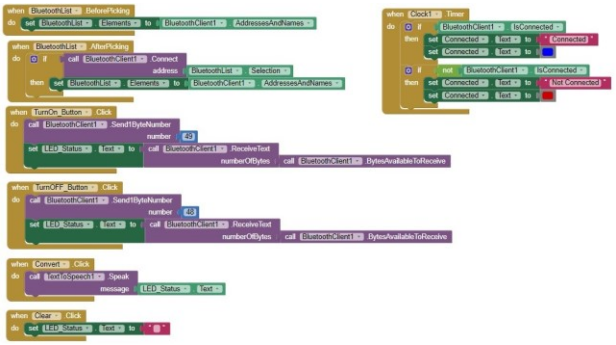

Figure 3.1: Application Code

B. *Software Design*

1) Neural Network Design: A neural network model using MNIST (Modified National Institute of Standards and Technology database) data-set as mentioned in [12], has been developed. The Size of MNIST Images is 28*28 pixels and hence the accuracy is very low. To improve the accuracy, different data-set as mentioned in [2] has been used, which consists of several thousands of images of the size 128*128 pixels. First, the images were preprocessed using the Python Image Processing (PIL) library. To build the model, advanced libraries such as Keras and Tensorflow were imported. The accuracy of the predicted images was still low. Hence to overcome this problem, our very own data-set was created which consisted of 1200 images for each gesture. To capture the images of various gestures; Python libraries such as OpenCV were imported and the data was organized into directories with the implementation of the python program. The images were then preprocessed and converted to grayscale and then to inverse binary images using the threshold for better edge detection. A convolutional neural network model having three convolutional layers was developed [10]. The data is then put into the convolutional neural network model and before training the model the grayscale images are augmented for better accuracy. Further, to improve two-way communication, a code has been built which converts Text to Sign language. To achieve this, the speech is taken as the input which is converted to text and then the alphabets in the word are interpreted individually using a predefined set of Sign Language images in the database. Once interpreted, a video is generated, consisting of the individual images; in sequence, representing each gesture; this is done via OpenCv Python Library.

IV. EXPERIMENTAL RESULTS AND ANALYSIS

A. *Results from the Sensors and glove*

The desired system is developed. A range of values for all the five flex sensors and the accelerometer is taken for all the gestures. The micro-controller is coded with the help of these values and the interpreted data is sent to the application via Bluetooth module. The ADC of the Micro-controller is 10-bits. The prototype glove can predict many gestures correctly as shown in Fig 4. There is a conflict between sensors values for some gestures such as ‘M’, ‘S’, ‘T’ and ‘E’ because of very close proximity in the gesture. Since this system was largely inaccurate, a new method to classify using computer vision was adopted. The data is further sent to the application via Bluetooth module for text to speech conversion.

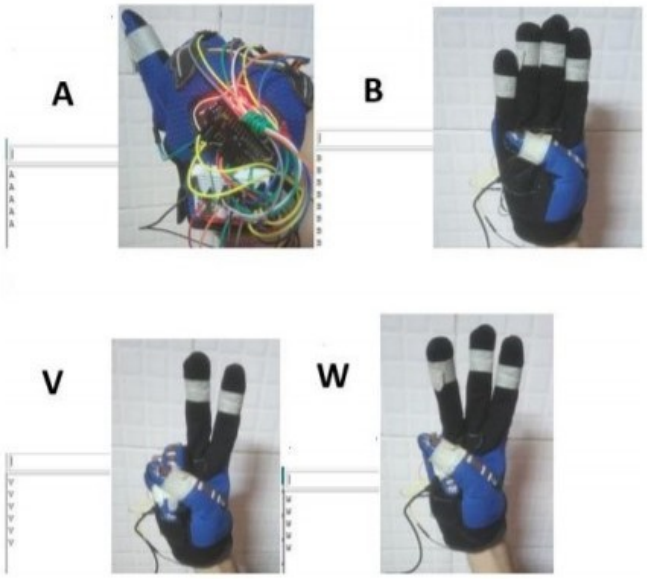

Figure 4: The System predicting the values shown on its left

B. *Results from The Neural Network*

1) Results from the MNIST (Modified National Institute of Standards and Technology database) Data set:

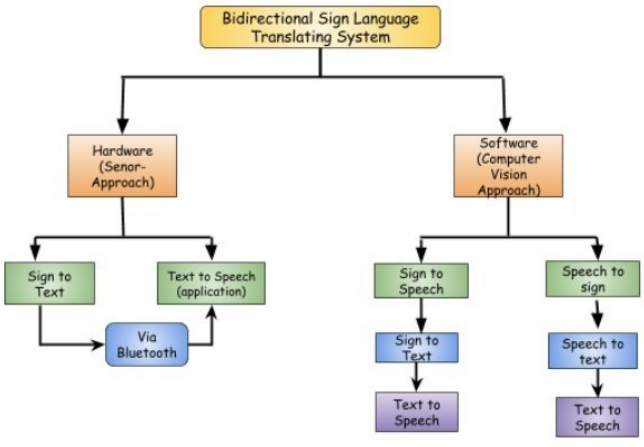

Figure 5: Flowchart of the System with both approaches.

Authorized licensed use limited to: Middlesex University. Downloaded on October 25,2020 at 14:54:39 UTC from IEEE Xplore. Restrictions apply.

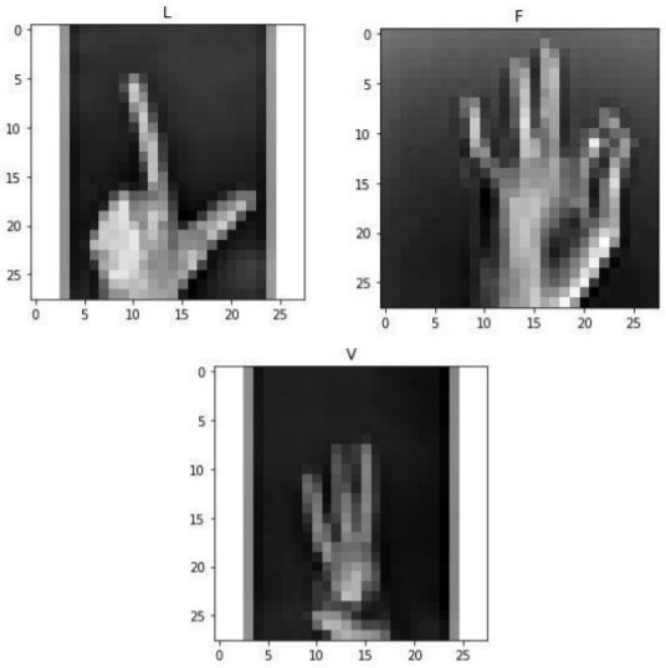

Figure 6: Correct prediction of alphabet L and F respectively

MNIST Dataset has been used initially as mentioned in [12] The pixel values of grayscale images in CSV format, ranging from 0-255 are taken as the input. The total number of images used in MNIST dataset for training and testing the model is 34,627 images. For the split, images used in training the model is 27,455 and testing the model is 7,172 to the MNIST data-set. The accuracy of the model comes out to be 99.2187 percentage. The major disadvantage is the very small pixel size of the image. The model is not able to classify custom images of larger sizes accurately Figure 6 shows the correct prediction of Alphabet L and F respectively. It also shows us the incorrect prediction of Alphabet 'W' as 'V', as seen in the bottom center image in figure 6, due to the low image quality.

Figure 7: Results obtained from Kaggle data-set

2) Results from the Kaggle Data set: In the data-set obtained from Kaggle as seen in [2], a total of 87,000 images in RGB format are taken as the input and were used for training and testing the model. For the split, images used in training the model is 82,650 and testing the model is 4350 with respect to the Kaggle Data-set. The accuracy comes out to be 99.08 percentage with an F1 Score of 98.85 percentage precision of 99.93 percentage and a recall value of 98.77 percent. The reason for low accuracy as compared to the MNIST data-set as the hand contour detection is tough as it may get merged with the background in particular cases. The major drawback is that it only predicts correctly on a plain background. If there are objects in the background, the accuracy significantly reduces as shown in figure 7. Many gestures are predicted correctly but few gestures are predicted incorrectly. The prediction is mentioned and also the correct alphabet associated with the gesture is mentioned in the name of the image.

3) Results from our Data set:

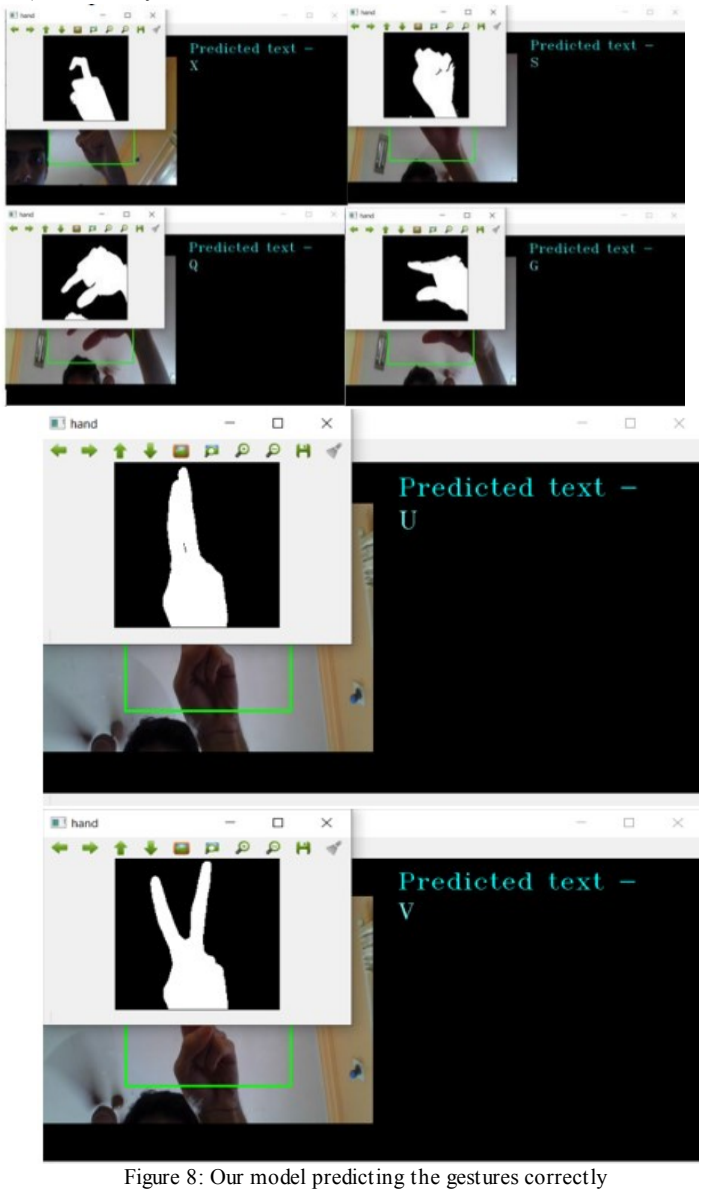

Figure 8: Our model predicting the gestures correctly

Authorized licensed use limited to: Middlesex University. Downloaded on October 25,2020 at 14:54:39 UTC from IEEE Xplore. Restrictions apply.

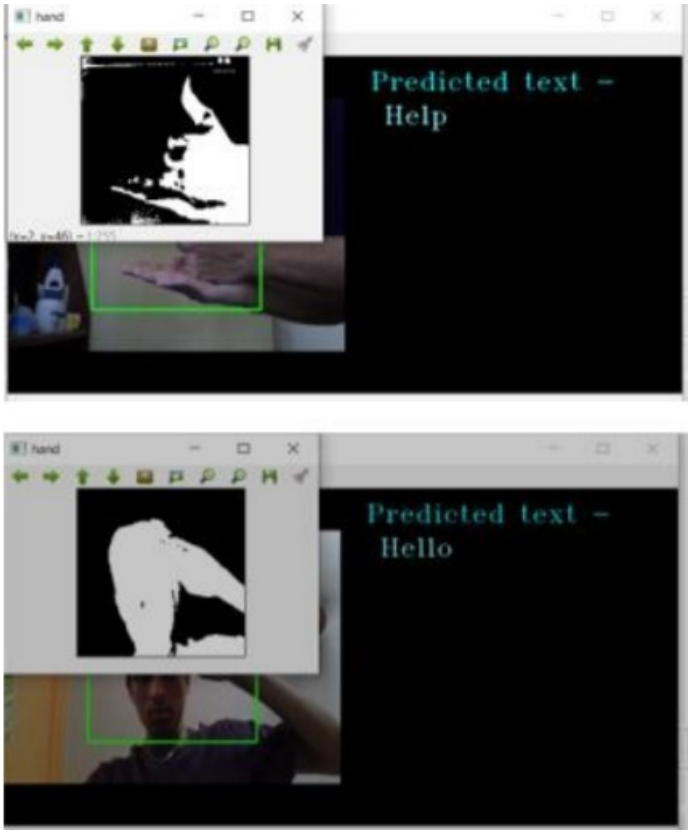

Figure 9: Prediction of Word gestures which includes face and hands

After MNIST and Kaggle, a data-set of our own was created to enhance accuracy. A data-set was created consisting of 1,200 images per gesture in the American Sign Language. A total of 31,200 images were used, out of which 21,840 were used for training the model while 9360 images were used for testing the model. Accuracy of the model using our data-set came out to be 99.98 percentage with an F1 score of 99.96 percentage, precision of 99.96 percentage and recall value of 99.96 percentage. The accuracy is more as compared to other data sets as the RGB image is converted to grayscale and then to an inverse binary image hence the gesture extraction becomes easier which in turn enhances accuracy. As shown in figure 8, every prediction is correct, the only alphabets that this model fails to predicts is J and Z. Since acustomized data set was built, different other gestures can also be included in this. Furthermore, the size of data-set can also be increased for better accuracy at the cost of computational power. Figure 9 shows that a different gesture is used for the word 'hello' and 'help'. In this way, additional gestures for different words which require face and both the hands can be added. The alphabets are further sent to the application for text to speech conversion.

C. *Speech to Text Translation*

To create a fully bidirectional functional system, sign to text translation has also been incorporated which is further subdivided into two parts. Initially, the speech is taken as the input and is converted to text with the implementation of the imported 'Speech Recognition' and 'PyAudio' libraries in python. An additional benefit is the inbuilt external noise cancelling features provided in it which reduces the disturbance in the input, thus giving better accuracy. The converted text is the output which automatically acts as the input for the next step.

D. *Text to Sign Translation*

The output i.e. the converted text in the preceding step is directly taken as the input and then the alphabets in the text are interpreted and the corresponding gesture is the output. A video is generated of the images consisting of hand gestures in a sequence of the alphabets in the text. This video can easily be interpreted in sign language communication. The vocabulary can be increased by adding the gesture images of the words in the database. This software can easily convert any input text to American sign language. It is important to note here, that once the individual alphabets are interpreted, it may be possible that a special case exists where those alphabets form a phrase which already has a special gesture for the entire phrase. In this case, the video will consist of the gesture for that phrase, hence making it easily comprehendible for the individuals. For instance, if the individual letters from the phrase "THANK YOU", then, the video will consist of the special gesture in the American Sign Language used for this phrase rather than showing a sequence of the gestures for each letter, starting from the letter 'T', and ending with 'U'

V. FUTURE SCOPE

The future scope for this project holds a broad scope across varied domains. Initially, it can be to build an application using 'TensorFlow lite' software and then integrate the Bluetooth application that has built for text to speech conversion. The data-set can be improved for more classification of more gestures. Several gestures include the movements of hands and hence to identify these gestures, a video classification algorithm can be incorporated to make the system, a complete ecosystem.

VI. CONCLUSION

In this project, various methodologies which translates sign language into text and further to speech, have been carefully analyzed. Initially, a hardware glove with flex sensors and accelerometer was designed. The accuracy of the glow was not very high. There were conflicts between some sensor values which caused the glove to predict some alphabets incorrectly. To improve accuracy, a model was built using existing data-set. The accuracy of the model was not as high as the data- set was not very versatile. Hence, software consisting of code to capture image was built and a custom data-set was created. The created data-set was preprocessed by converting it to a gray-scale image, and then further converting it to an inverse binary image using a suitable threshold value; this value was calculated by trial and error. This was then sent

Authorized licensed use limited to: Middlesex University. Downloaded on October 25,2020 at 14:54:39 UTC from IEEE Xplore. Restrictions apply.

to a model for training. The accuracy was 99.98 percent as the data-set was made versatile. Several new gestures can be added by training the model with new required data-set. Hence this model increases the scope of the sign language prediction. The model can predict various other sign languages such as Indian sign language and British Sign language. Further to incorporate a fully functioning bidirectional communication system, a speech to sign conversion software is also built to convert the input speech to text and then further convert it to a video of image sequences consisting of the gestures. Furthermore, as the database is customized for user convenience, the vocabulary can be enhanced by the addendum of more images in the database.

ACKNOWLEDGMENT

We avail this opportunity to express our indebtedness to our mentor, Professor. Manisha Bansode, Department of Electronics Engineering at the Sardar Patel Institute of Technology, Mumbai for her valuable guidance and help at various stages. We are grateful to Dr. D.C Karia, Head of Department, Electronics Engineering at the Sardar Patel Institute of Technology, Mumbai for allowing us to access laboratory facilities in the department.

REFERENCES

[1] World health organization report. [Online]. Available: https://www.who.int/news-room/ fact-sheets/detail/deafness-and-hearing-loss.

[2] Akash. Asl alphabet. [Online]. Available: https://www.kaggle.com/grassknoted/asl-alphabet, 2018.

[3] Kshitij Bantupalli and Ying Xie. American sign language recognition using deep learning and computer vision. In *2018 IEEE International Conference on Big Data (Big Data)*, pages 4896–4899. IEEE, 2018.

[4] Kanchan Dabre and Surekha Dholay. Machine learning model for sign language interpretation using webcam images. In *2014 International Conference on Circuits, Systems, Communication and Information Technology Applications (CSCITA)*, pages 317–321. IEEE, 2014.

[5] Aditya Das, Shantanu Gawde, Khyati Suratwala, and Dhananjay Kalbande. Sign language recognition using deep learning on custom processed static gesture images. In *2018 International Conference on Smart City and Emerging Technology (ICSCET)*, pages 1–6. IEEE, 2018.

[6] Kunal Kadam, Rucha Ganu, Ankita Bhosekar, and SD Joshi. American sign language interpreter. In *2012 IEEE Fourth International Conference on Technology for Education*, pages 157–159. IEEE, 2012.

[7] Tülay Karayılan and Özkan Kılıç. Sign language recognition. In *2017 International Conference on Computer Science and Engineering (UBMK)*, pages 1122–1126. IEEE, 2017.

[8] A. Kumar, K. Thankachan, and M. M. Dominic. Sign language recognition. In *2016 3rd International Conference on Recent Advances in Information Technology (RAIT)*, pages 422–428, 2016.

[9] A. S. Nikam and A. G. Ambekar. Sign language recognition using image based hand gesture recognition techniques. In *2016 Online International Conference on Green Engineering and Technologies (IC-GET)*, pages 1–5, 2016.

[10] S. Albawi, T. A. Mohammed and S. Al-Zawi, "Understanding of a Convolutional neural network," 2017 International Conference on Engineering and Technology (ICET), Antalya, 2017, pp. 1-6, doi:10.1109/ICEngTechnol.2017.8308186.

[11] Shaheer Bin Rizwan, Muhammad Saad Zahid Khan, and Muhammad Imran. American sign language translation via smart wearable glove technology. In *2019 International Symposium on Recent Advances in Electrical Engineering (RAEE)*, volume 4, pages 1–6. IEEE, 2019.

[12] Tecperson. Sign language mnist. . [Online]. Available: https://www.kaggle.com/datamunge/ sign-language-mnist, 2017.

[13] P Vijayalakshmi and M Aarthi. Sign language to speech conversion. In *2016 International Conference on Recent Trends in Information Technology (ICRTIT)*, pages 1–6. IEEE, 2016.

Authorized licensed use limited to: Middlesex University. Downloaded on October 25,2020 at 14:54:39 UTC from IEEE Xplore. Restrictions apply.