A Multi-Modular Approach for Sign Language and Speech Recognition for Deaf-Mute People

D.T.D.M. Dahanayaka, B.G.D.A. Madhusanka and I.U. Atthanayake

Abstract: Deaf and Mute people cannot communicate efficiently to express their feelings to ordinary people. The common method these people use for communication is the sign language. But these sign languages are not very familiar to ordinary people. Therefore, effective communication between deaf and mute people and ordinary people is seriously affected. This paper presents the development of an Android mobile application to translate sign language into speech-language for ordinary people, and speech into text for deaf and mute people using Convolution Neural Network (CNN). The study focuses on vision-based Sign Language Recognition (SLR) and Automatic Speech Recognition (ASR) mobile application. The main challenging tasks were audio classification and image classification. Therefore, CNN was used to train audio clips and images. Mel-frequency Cepstral Coefficient (MFCC) approach was used for ASR. The mobile application was developed by Python programming and Android Studio. After developing the application, testing was done for letters A and C, and these letters were identified with 95% accuracy.

Keywords: Sign language recognition, Automatic speech recognition, Deaf-mute, Mobile APP, Convolution neural network

1. Introduction

Hearing loss is one of the major barriers that one can face in day-to-day activities, especially that involves social interactions. The inability of capturing the sounds from the environment by the ear is known as hearing weakness or deafness. This condition can either be temporary or permanent. A person who cannot speak, either from incapability of speaking or reluctance to speak, is called a mute. Deaf and mute people are unable to communicate efficiently with others and express their feelings to ordinary people. As a solution for this, they use sign language as the first language. Examples of some sign languages are Sri Lankan Sign Language, American Sign Language, British Sign Language, Japanese Sign Language, and the Indian Sign Language. Figure 1 shows American Sign Language (ASL) signs representing numbers 0-9 and letters of the English alphabet. Sign language is the critical source for conveying messages and emotions for deaf and mute people. Also, sign language helps communicate between the speaking community and the person who neither speaks nor hears. Yet the sign language does not help to communicate efficiently between ordinary people. Because the common societies are unaware of the sign language, deaf and mute people are isolated from society due to this communication barrier. Therefore, the aim of the present study is to develop a mechanism to prevent a communication barrier between deaf-mute people and the ordinary people. A mobile

application development was identified as the method for facilitating the said barrier.

Figure 1 - ASL Signs Representing Numbers 0-9 and Letters of the English Alphabet [4]

Eng. (Ms.). D.T.D.M. Dahanayaka,
AMIE(SL), BTec(Eng)(Hons)(OUSL), Department of
Mechanical Engineering, The Open University of Sri Lanka,
Nugegoda.

Email: tharudahanayaka@gmail.com

<http://orcid.org/0000-0003-4622-1872>

Eng. B.G.D.A. Madhusanka, BTech(Eng)(Hons)
(OUSL), MSc(Moratuwa), Lecturer, Department of
Mechanical Engineering, The Open University of Sri
Lanka, Nugegoda. Email: hgmada@ou.ac.lk

<http://orcid.org/0000-0002-2093-2919>

Dr. (Mrs.) I.U. Atthanayake,
B.Sc.Eng.(Hons)(Peradeniya), Mphil(UOM),
PhD(Warwick), Head of the Department, Senior Lecturer
(G I), The Open University of Sri Lanka.

<http://orcid.org/0000-0002-6667-2008>

The most common method that is used by Deaf, Mute People (DMP) to communicate with ordinary people is the use of an interpreter. To remove this gap, various devices have been invented as solutions [1]. Sign Language Recognition (SLR) and Automatic Speech Recognition (ASR) systems are some of them.

One of the recent approaches for the SLR is the sensor-based glove innovation and computer vision-based approaches [2, 3, 4]. The sensor-based approach requires sensors and instruments to detect the hand's motion, position, and velocity. Inertial Measurement Unit (IMU) sensors such as gyroscope and accelerometer have been used to obtain the orientation, angles and acceleration information [5, 6, 7]. For some reason, such as the necessity for sensor-based approaches to be worn all the time, and the need for the circuit to be protected, the device is uncomfortable and challenging to handle. This approach has a reasonable recognition rate, but the computation complexity and high cost are difficult to maintain in public places. To avoid a severe problem, the vision-based approach is more beneficial [8, 9, 10]. Vision-based approaches require the acquisition of images or video of the hand gestures through a video camera. Many research had been done using image processing hand gestures for sign language. But they are low in accuracy and take a long processing time.

The DMP cannot understand the common language since the ASR is significant. Also, ASR is the transcription of human speech into spoken words. It is a challenging task since the human voice signals are inconsistent due to change in speaker qualities, different speaking styles, uncertain background noises, etc. [11].

A mobile application for SLR and ASR has not been developed so far. To the authors' best knowledge this is the first study that uses CNN in Android applications. Furthermore, the real

time conversion of sign language to audio with mobile has not been developed in previous studies.

2. Methodology

Convolution Neural Network (CNN) was used to train audios and images. As the first step, the conversion of the Sign Language into audio using CNN was determined [11, 12, 13]. Using CNN can achieve accurate results. CNN converts images into a lower dimension without losing their characteristics. In CNN, there are different layers, Convolutional Layers with ReLU, Pooling Layers, Flatten Layers, Fully connected (FC) layer, Softmax, and Output layer. A simple CNN architecture is shown in Figure 2.

A. Training for SLR

For the training process, models are trained using Colab because it offers free access to a computer with an appropriate GPU, a cloud service based on Jupyter Notebooks, and it supports Python language. There are various algorithms widely adopted in this field for object detection. They are You Only Look Once (YOLO), Faster Region CNN (F-RCNN) and Single Shot Detector (SSD) [14, 15, 16, 17].

In this project, SSD300 was used to train images. SSD is designed for object detection in real-time. SSD uses VGG16 to extract feature maps. After the VGG16, SSD adds six more auxiliary convolution layers. A Single Shot Multi Box Detector architecture is shown in Figure 3.

For the image training, the block diagram is shown in Figure 4. As shown in the figure, first, the frames were extracted from input videos.

Then annotation was done. Then images were trained. After the training process using CNN, the Trained classifier was obtained.

Figure 2 - A Simple CNN Architecture

Figure 3 - Single Shot MultiBox Detector Architecture

Figure 4 - Block Diagram for the Image Training

In this study, the first four letters of the ASL alphabet, "A", "B", "C" and "D", were used for the training process. Then these images were divided into four classes, "A", "B", "C" and "D". There were 3000 RGB images in each class. Then all images were converted into a fixed size of 300×300 pixels. Then images were split as 80% training data and 20% testing data. Then, as shown in Figure 5, the annotation was done for all images using labellmg.exe. Images were trained by inserting the required for the SSD

Figure 5 - Annotation Done for the Image using labellmg.exe

algorithm. After that, the training model was obtained.

B. Training for ASR

ASR is the recognition of spoken words by a computer or a machine. It is difficult to identify the correct words. Same words may have different pronunciations and may have a different time duration. Also, may have different speech qualities and indeterminate background sounds, etc. Therefore, the training process requires a large amount of data. Hence, feature extraction is very important for the ASR. It transforms the input data into a set of features. Then, it removes annoying information and retains only the valuable information called the set of properties or parameters of utterances like speech flow, loudness, intonation, and overtone intensity. There are several feature extraction methods. They are Mel frequency Cepstral Coefficient (MFCC), Linear prediction coding (LPC), Linear prediction cepstral coefficient (LPCC), Discrete Wavelet Transform (DWT), Wavelet Packet Decomposition (WPD), Perceptual Linear Prediction (PLP), etc. [18, 19, 20]. In this study, MFCC was used for the

feature extraction. MFCC feature extraction process steps are, first take the Fourier transform of the signal. Then map the log amplitudes of the spectrum to the Mel scale. Then discrete Cosine transform the Mel log-amplitudes. Then MFCCs are the amplitudes of the resulting spectrum. MFCC feature extraction process steps are shown in Figure 6.

Figure 6 - MFCC Feature Extraction Process Steps

A frequency measured in Hz can be converted to the Mel scale using the following formula, where f denotes the physical frequency in Hz,

and $Mel(f)$ denotes the perceived frequency. An example of the MFCC is shown in Figure 7.

$$Mel(f) = 2595 \log[1 + (f/700)]$$

Figure 7 - MFCC

The block diagram shows the audio classification in Figure 8. First, all the audios were converted into images. Then the feature extraction was done. After that, the training process was done using CNN. Then the trained classifier was obtained.

A lossless audio file format is the best format for sound quality. WAV files are lossless and uncompressed music files. An MP3 file is a compressed music file. WAV files are also the right choice for editing a podcast and MP3 files for distribution. Therefore, WAV music files were used for this study. First, the audio clips

Figure 8 - Block Diagram for the Audio Classification

were collected for four keywords. They were "up", "left", "stop", and "on". Then audio samples were divided into four classes, and they were "up", "left", "stop", and "on". Then audio samples were broken into one second small audio clips.

If the WAV file load is a NUMPY array in a simple NN, it will be stored as a 1-D matrix. Therefore, many neurons need the first layer as input neurons, and calculations will add extreme complexity. But, if MFCC representation for the .wav files is used with CNN, it will give highly accurate results [21, 22, 23]. There were 3500 RGB images in each class. Then all images are converted into a fixed size of

300×300 pixels using the MFCC representation. Then CNN was used to train the images. Figure 9 shows the way MFCC files feed to CNN.

C. Android Mobile Application

First of all, SLR and ASR trained models were converted into. tflite models using the TensorFlow Lite Converter. TensorFlow Lite, is the next development of TensorFlow. For mobile and embedded devices, TensorFlow Lite is TensorFlow's lightweight solution. It enables on-device machine learning inference with a small binary size and low latency. TensorFlow Lite supports hardware acceleration with the Android Neural Networks API [24, 25]. TensorFlow Lite architecture is shown in Figure 10.

Figure 9 - CNN Model with MFCC File

Figure 10 - TensorFlow Lite Architecture

To create an Android mobile Application Android Studio was used. For the new Android Studio project, Java language was selected. After that, as shown in Figure 11, layouts were created for the Android application user interface.

D. Image Prediction for SLR

The block diagram for image prediction for SLR is shown in Figure 12. A trained classifier

(model) was used to recognize the sign captured from Mobile Application. First, video is captured from the phone camera. Then the real-time video is converted into frames. Then trained classifier (model) is used to predict with frames. After that, the output can be obtained.

Figure 11 - Basic Layout of the APP

Figure 12 - Block Diagram for the Image Prediction for SLR

E. Audio Prediction for ASR

Figure 13 - Block Diagram for the Audio Prediction for ASR

Block diagram for the Audio prediction for ASR is shown in Figure 13. When unknown audio is captured from the phone, the feature extraction is done. Then the prediction is made with a trained classifier. Finally, the result can be obtained.

classifier over time. If the model makes less losses, it means that the model has learned all the details.

B. Audio training results

3. Results

A. Image training results

Figure 14 - Image Training Results

Image training results for SLR are shown in Figure 14. The progress of the model training process can be obtained using the Tensorboard graph. One of the important graphs is the Loss graph, which shows the overall loss of the

Figure 15 - Audio Training Accuracy

Figure 16 - Audio Training Loss

The model performance and audio training results are shown in Figure 15 and 16.

C. Mobile application SLR testing in real-time

The real-time video is captured, and the prediction is made with images that have been trained. Finally, as shown in Figure 17 and Figure 18, the result is given as a text and an audio.

Figure 17 - Mobile App Testing Result for ASL Letter "A"

Figure 18 - Mobile App Testing Result for ASL Letter "C"

D. Mobile application ASR testing in real-time

Real-time audio is captured, and the prediction is made.

3. Conclusions

This paper presents a solution for effective communication between ordinary and deaf or mute people. This research facilitates to remove the communication barrier between ordinary people and persons who cannot communicate verbally. A mobile application has been developed to translate sign language into audio so that ordinary people can understand easily, and convert speech into text to facilitate understanding of the deaf people. The main challenge was to operate CNN in android studio to recognise the hand signs. The task was accomplished by using Mel-frequency Cepstral Coefficient (MFCC).

References

1. Sunitha, K. A., Anitha, S. P., Aarthi, M., Jayapriya, K. and Lingam Sunny, 2016. "Deaf Mute Communication Interpreter-A Review", *International Journal of Applied Engineering Research*, 11, pp.290-296.
2. Cheok, M. J., Omar, Z. & Jaward, M. H., 2019. "A Review of Hand Gesture and Sign Language Recognition Techniques", *International Journal of Machine Learning and Cybernetics*, 10(1), pp.131-153.
3. Rajan, R. G., Judith Leo, M., 2019. "A Comprehensive Analysis on Sign Language Recognition System", *International Journal of Recent Technology and Engineering (IJRTE)*.
4. Ghanem, S., Conly, C. & Athitsos, V., 2017. "A Survey on Sign Language Recognition using Smartphones", *In Proceedings of the 10th International Conference on Pervasive Technologies Related to Assistive Environments*, (pp. 171-176). ACM.
5. Lokhande, P., Prajapati, R. & Pansare, S., 2015. "Data Gloves for Sign Language Recognition System", *International Journal of Computer Applications*, 975, p.8887.
6. Ambar, R., Fai, C. K., Wahab, M. H. A., Jamil, M. M. A. & Ma'radzi, A. A., 2018. "Development of a Wearable Device for Sign Language Recognition", *In Journal of Physics: Conference Series*. Vol. 1019, No. 1, p. 012017.

7. Ahmed, M. A., Zaidan, B. B., Zaidan, A. A., Salih, M. M. & Lakulu, M. M. B., 2018. "A Review on Systems-Based Sensory Gloves for Sign Language Recognition State of the Art between 2007 and 2017", *Sensors*, 18(7), p.2208.
8. Sagar, P., Sattar, A., 2015. "HAND GESTURE RECOGNITION SYSTEM FOR DUMB PEOPLE".
9. Kaur, C., Gill, N., 2015. "An Automated System for Indian Sign Language Recognition", *International Journal of Advanced Research in Computer Science and Software Engineering*.
10. Shivashankara, S., Srinath, S., 2018. "American Sign Language Recognition System: An Optimal Approach". *International Journal of Image, Graphics & Signal Processing*, 10(8).
11. Ivanko, D., Ryumin, D. & Karpov, A., 2019. "Automatic Lip-Reading of Hearing Impaired People". *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 42(2/W12).
12. Rahim, M. A., Islam, M. R. & Shin, J., 2019. "Non-Touch Sign Word Recognition Based on Dynamic Hand Gesture Using Hybrid Segmentation and CNN Feature Fusion", *Applied Sciences*, 9(18), p.3790.
13. Ramkumar, L. K., Premchand. S & Vijayakumar, G. K., 2019. "Sign Language Recognition using Depth Data and CNN", *SSRG International Journal of Computer Science and Engineering (SSRG – IJCSE)*, V6(01).
14. Shinde, S., Kothari, A. & Gupta, V., 2018. "YOLO Based Human Action Recognition and Localization", *Procedia computer science*, 133, pp.831-838.
15. Kang, Y., 2020. "Research on SSD Base Network", *IOP Conference Series: Materials Science and Engineering*, 768, p.072031.
16. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. & Berg, A., 2016. "SSD: Single Shot MultiBox Detector", *Computer Vision – ECCV 2016*, pp.21-37.
17. Kumar, A., Srivastava, S., 2020. "Object Detection System Based on Convolution Neural Networks. Using Single Shot Multi-Box Detector", *Procedia Computer Science*, 171, pp.2610-2617
18. Naziya S., S., Deshmukh, R., 2016. "Speech Recognition System – A Review", *IOSR Journal of Computer Engineering*, 18(04), pp.01-09.
19. Abdel-Hamid, O., Mohamed, A.R., Jiang, H., Deng, L., Penn, G. & Yu, D., 2014. "Convolutional Neural Networks for Speech Recognition", *IEEE/ACM Transactions on audio, speech, and language processing*, 22(10), pp.1533-1545.
20. Sandasarani, N. A. C., 2015. "Sinhala Speech Recognition", *International Journal of Engineering Research & Technology (IJERT)*, ISSN: 2278-0181.
21. Han, Y., Lee, K., 2016. "Acoustic Scene Classification using Convolutional Neural Network and Multiple-Width Frequency-Delta Data Augmentation", *arXiv Preprint arXiv:1607.02383*.
22. Wyse, L., 2017. "Audio Spectrogram Representations for Processing with Convolutional Neural Networks", *arXiv preprint arXiv:1706.09559*.
23. Khamparia, A., Gupta, D., Nguyen, N. G., Khanna, A., Pandey, B. & Tiwari, P., 2019. "Sound Classification Using Convolutional Neural Network and Tensor Deep Stacking Network", *IEEE ccess*, 7, pp.7717-7727.
24. Choe, D., Choi, E. & Kim, D., 2020. "The Real-Time Mobile Application for Classifying of Endangered Parrot Species Using the CNN Models Based on Transfer Learning", *Mobile Information Systems*, 2020, pp.1-13. Ghg.
25. Maung, S. Z. M., Aye, N., 2019. "Convolutional Neural Network Based Recognition of Myanmar Text Warning Sign for Mobile Platform", *International Journal of Engineering Research & Technology (IJERT)*, V8 (01).