

**CSE-3200: System Development Project**

**Convolutional Neural Network based Sign Language Detection on Android Application**

**By**

**Name:** Md Hefzul Hossain Papon

**Roll:** 1807089

**Name:** Argho Deb Das

**Roll:** 1807066

**Supervisor:**

Dr. Pintu Chandra Shill

Professor

Department of Computer Science and Engineering

Khulna University of Engineering & Technology,Khulna

# Approval

This Project Report has been submitted for examination with the approval of our supervisor.

[Signature]

Dr. Pintu Chandra Shill

# Acknowledgement

We are thankful to our supervisor, **Dr. Pintu Chandra Shill** to give us proper guidelines to complete our project successfully. We have completed our “CSE 3200 System Development Project” titled “**Convolutional Neural Network based Sign Language Detection on Android Applicaition”** with help of our supervisor sir.

Argho Deb Das &

Md Hefzul Hossain Papon

Abstract

Sign language is the communication medium used by the deaf and dumb people around the world. The deaf people have a hearing disability and the dumb or mute people have a speaking disability except other things are normal for them. So they use sign language to communicate with each other and with the rest of the world. So sign language is very necessary thing to communicate with them. Many researchers worked with sign language in different language, Someone work with only digit. Many researcher work with this sign language but it should be improved. Nowadays Android application platform is becoming very popular. That’s why we choose Android Application to implement sign language. Here we work with sign language detection for alphabets sign. We have mainly focused on Convolutional Neural Network (CNN) besides we used KNN, Random Forest. In our model we have used dataset that is collected form Kaggle. We have used approximately 10500 images for alphabet in dataset. We also wish to work further work on this field.

**Contents**

Approval\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ I

Acknowledgement \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_II

Abstract\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_III

1. Introduction \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_1

2. Background \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_1

3. Methodology \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2

3.1 Dataset creation \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_3

3.2 Dataset Preprocessing\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_3

3.3 Train Test Split \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_3

3.4. Model Optimization\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_3

4. Model Evaluation\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_5

5. Software development & Graphical User Interface\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_6

6. Conclusion \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_10

7. References\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_10



**Convolutional Neural Network based Sign Language Detection on Android Application**

1. Introduction

Sign language is manual communication commonly used by people who are deaf. Sign language is not universal.The hands, eyes, face, lips, and body are used as visual clues. In sign language, the motions or symbols are arranged linguistically. It combines a variety of nonverbal communication techniques, such as finger spelling, hand gestures, body language, facial emotions etc. The majority of sign language users are deaf persons. Others who have trouble hearing also find it to be a useful tool for communication.. It is also used by those hearing individuals who can’t use speech, be it due to a disability or condition, or by those who have Deaf family members, and used even sometimes by monks who have taken a vow of silence.

About 5% of the global population are speech impaired, and they use Sign language to communicate with other deaf and dumb people or a normal people. Normal people unable to understand The Sign language. For making communication between normal people who don’t understand sign language and deaf and dumb people, a sign language interpreter can be used. Every country has its own sign language. In our world, there are various sign languages like American Sign Language (ASL), British Sign Language (BSL) etc.

Many researchers worked on different sign languages and till working now on different projects. But sign language detection with android app has low researched work. We have worked on English alphabet recognition and implemented it on android platform. We used different Deep learning models to get a better results. Mainly we have focused on Convolutional Neural Network (CNN), besides we used k-Nearest Neighbor (KNN) and Random Forest (RF).

Huge number of deaf and dumb peoples in our world are not burden in our society. If they are able to communicate with others they can do all kinds of works like a normal people. Using our system the hearing impaired and speech impaired people can lead a modern life. Our main target is to educate people about sign language so that they are able to communicate with this unfortunate group of people.

# Background

In the 1960s, a young English professor at Gallaudet College, William Stokoe, who had studied linguistics, began to look at American Sign Language (ASL) as a linguist and discovered that it was full of regularities and structure, very much like a spoken language. He applied for and received an NSF grant to study it further, a grant for which the NSF was publicly excoriated by the reigning establishment in deaf education (primarily descendants and intellectual heirs of Alexander Graham Bell, who favored an "oralist" approach that included preventing deaf students from using sign language). With the grant, Stokoe published the Dictionary of American Sign Language. The ASL dictionary attracted the attention and interest of a number of other linguists and psychologists. This work quickly led to a new understanding that ASL is in fact a full-blown language. Besides opening up a whole new dimension in linguistic study, which is still being pursued vigorously, this finding revolutionized deaf education in the U.S.

# Methodology

The model was first trained with a dataset. After that we have generated a tflite file and imported that into android studio. Then using Tensor Flow Lite and OpenCv we have managed to develop an android app which recognizes hand sign and also able to educate people about the signs which will help them to communicate with the deaf community.The entire procedure of Dataset pre-processing, Train test split, Model Optimization, Model evaluation, Implementation, Application development are described bellow:

## **3.1 Dataset creation**

We collected our dataset from *Kaggle.net.* The dataset has 26 classes(A, B, C, D, E, F, G, H, I, J, K, L, M, N, O, P, Q, R, S, T, U, V, W, X, Y, Z). For each letter we have approximately 425 images and filtered out some of them which contain signs in the wrong style. After filtered out finally we have about 11791 images total.

|  |  |  |  |
| --- | --- | --- | --- |
| **Class Name** | **Image Number** | **Class Name** | **Image Number** |
| A | 397 | N | 500 |
| B | 409 | O | 482 |
| C | 399 | P | 469 |
| D | 420 | Q | 453 |
| E | 431 | R | 479 |
| F | 422 | S | 482 |
| G | 440 | T | 481 |
| H | 477 | U | 472 |
| I | 481 | V | 448 |
| J | 487 | W | 465 |
| K | 467 | X | 451 |
| L | 439 | Y | 469 |
| M | 471 | Z | 400 |

Table 3.1: Dataset summary

## 

## **hi**

Fig 3.2: Dataset Preview

## **3.2 Dataset Preprocessing**

The training images needs to be pre-processed before they can be used in CNN or any other model. Despite of taking many steps at the dataset creating time, out dataset still contained many errors and is not suitable to use directly in our model. So pre-processing must needs to train our model.

## **3.3 Train Test Split**

At the time of letter training we split dataset 85% for training and 15% for testing. From out total 11,400 images, 9690 images were used in training and 1710 images used in testing.

## **3.4. Model Optimization**

**Convolutional Neural Network (CNN):** Multi-layer convolutional neural networks with interconnected layers were employed to recognize our characters as seen in figure 3.2. We have stacked the layers and created an architecture. The first layer is a ‘functional’ layer which was pre-trained model named EfficientnetB0 and it uses dataset from Imagenet . Then we have added a ‘global-average-pooling’ layer . We have used a 30% dropout before the fully connected layer to reduce overfitting. With the default activation function ReLU, we have used two fully connected layers. We have compiled the model using the ‘Adam’ optimizer and the ‘mae’ loss. We fit the model with 50 epochs and a batch-size 32. Fig.3.3 is showing the summary of CNN model.

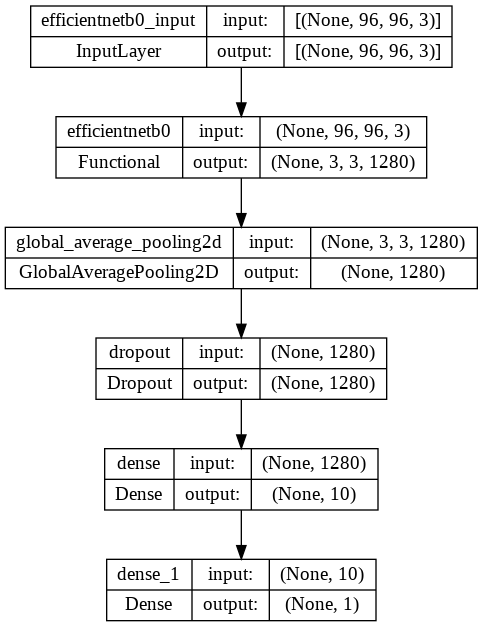
****

Fig 3.3: Summary of CNN model

**K- Nearest Neighbor (KNN):**

KNN is a non-parametric and lazy learning algorithm. Non-parametric means there is no assumption for underlying data distribution. According to KNN, every data point that is close to another belongs to the same class. It chooses a number k as the nearest neighbour to the data point to be identified. In our proposed model convolution layers are used as input to the KNN classifiers for the prediction. We have used python scikit-learn toolkit for the implementation of KNN.

**Random Forest (RF):**

A random forest is a meta estimator that fits a number of decision tree classifiers on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting. Classic CNNs can face difficulties because of over-fitting issues. In our proposed model we have used convolution layers as a input to random forest. Thus we can make our model more generalized and reduce the possibility of over-fitting. 50 trees have been used in our random forest model.

# Model Evaluation:

We have used an ASL dataset consisting of 11791 images having many varieties and that’s why call it novel dataset. Our CNN model, developed with digit dataset provided a 98.9% goodness of fit . We use 85% of our total dataset for training and 15% of our total dataset for testing. We have trained our model using three different classifier. We have displayed the result of R2\_score and Mean average error dataset in table 4.1 .We have trained our letter model with 10022 images and evaluate our letter model with 1769 images . After a few epochs of running, the decreasing behavior of training loss indicated that models have been well-trained on the dataset.

We have calculated the accuracy by dividing the total number of estimates by the number of right estimates. The evaluation metrics are shown in the table 1 below:

Table 4.1: Accuracy of Letter dataset over different model

|  |  |  |
| --- | --- | --- |
| **Model** | **R2 Score** | **Mean Absolute Error** |
| CNN | 98.89% | 0.18 |
| KNN | 70.0% | 1.7 |
| RF | 76.9% | 1.54 |

From the table we can see that CNN is a superior approach . We decided that the CNN model may be used as a benchmark model for sign language detection after analyzing the overall findings. The graphical representation of accuracy of different model is shown in figure 4.1 and 4.2.

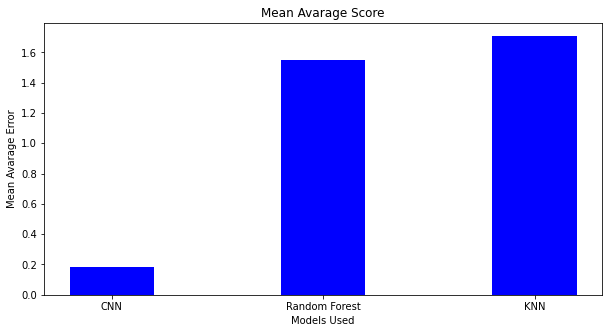


Fig 4.1: Graphical representation of Mean Absolute Error of model.

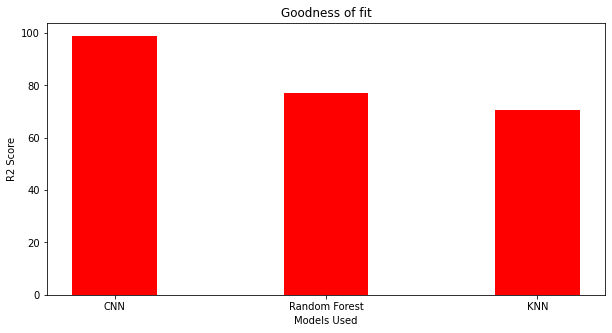


Fig 4.2: Graphical representation of R2 score of model.

# Software development & Graphical User Interface:

## **5.1 Technologies used:**

In order to implement the project from start to finish, the programming language we used is ***Python 3.10***. For mathematical operations we used, ***numpy,matplotlib*** and ***pandas*** libraries. For computer vision related tasks like preprocessing we used ***OpenCV***. For machine learning model implementation, training, predicting etc. we used ***TensorFlow 2.0*** and ***Keras*** (integrated with TensorFlow 2.0) platforms. And lastly, for developing the android application we used Java,Open Cv library and Tensorflow lite. Tensorflow lite  provides a set of tools that enables on-device machine learning by allowing developers to run their trained models on mobile, embedded, and IoT devices and computers .

Software used for this purpose are : Android Studio , Jupyter Notebook and Google Colab.

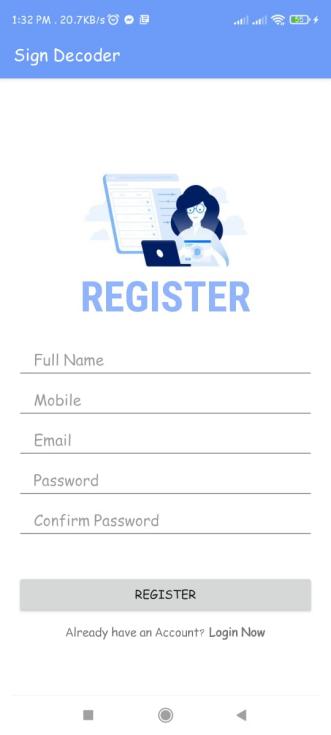
## **5.2 Graphical User Interface & Functionality:**

**Splash Screen:** It is shown when app is opened

****

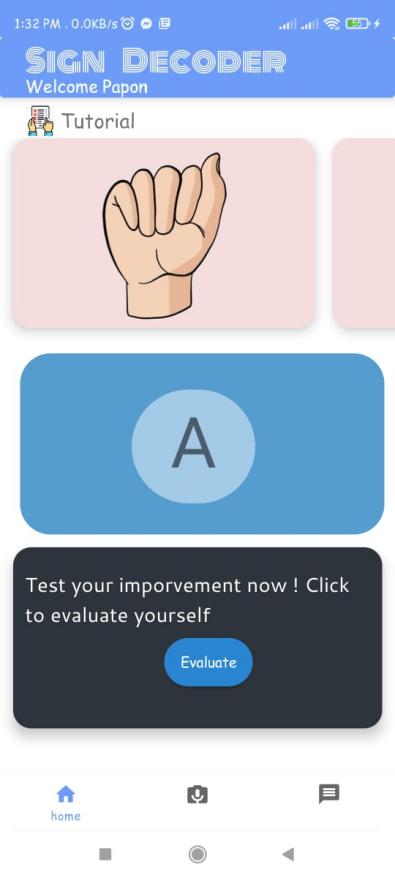
**Fig 5.1 :** Splash Screen

**Registration Page:** Login and Registration done with Firebase .

****

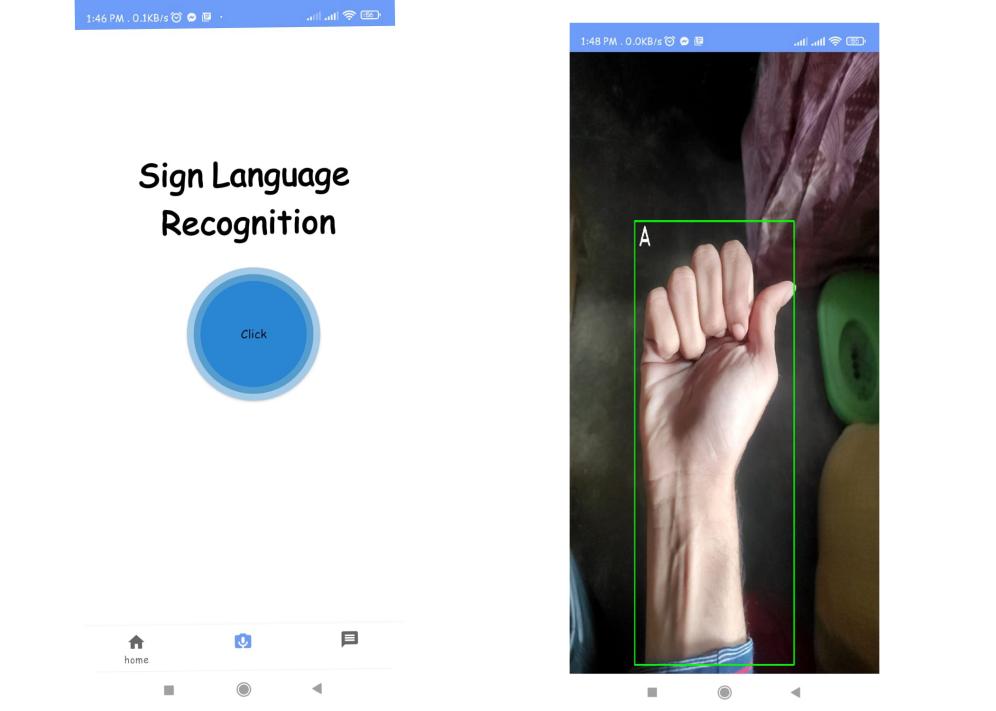
**Fig 5.2 :** Register Screen

**Homepage :** Shows hand sign tutorials

****

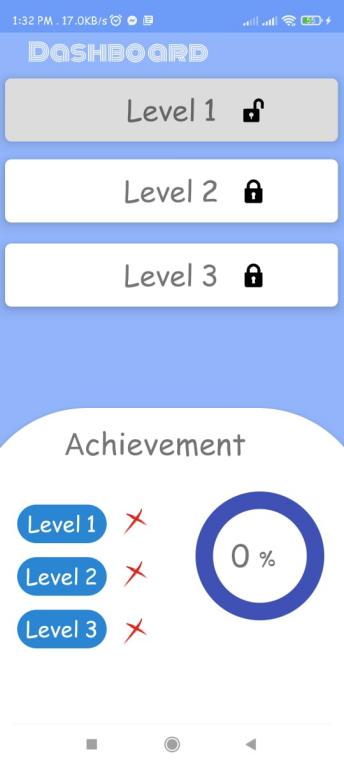
**Fig 5.3 :** Home Screen

**Recognizer Page:** Recognizes hand signs .

****

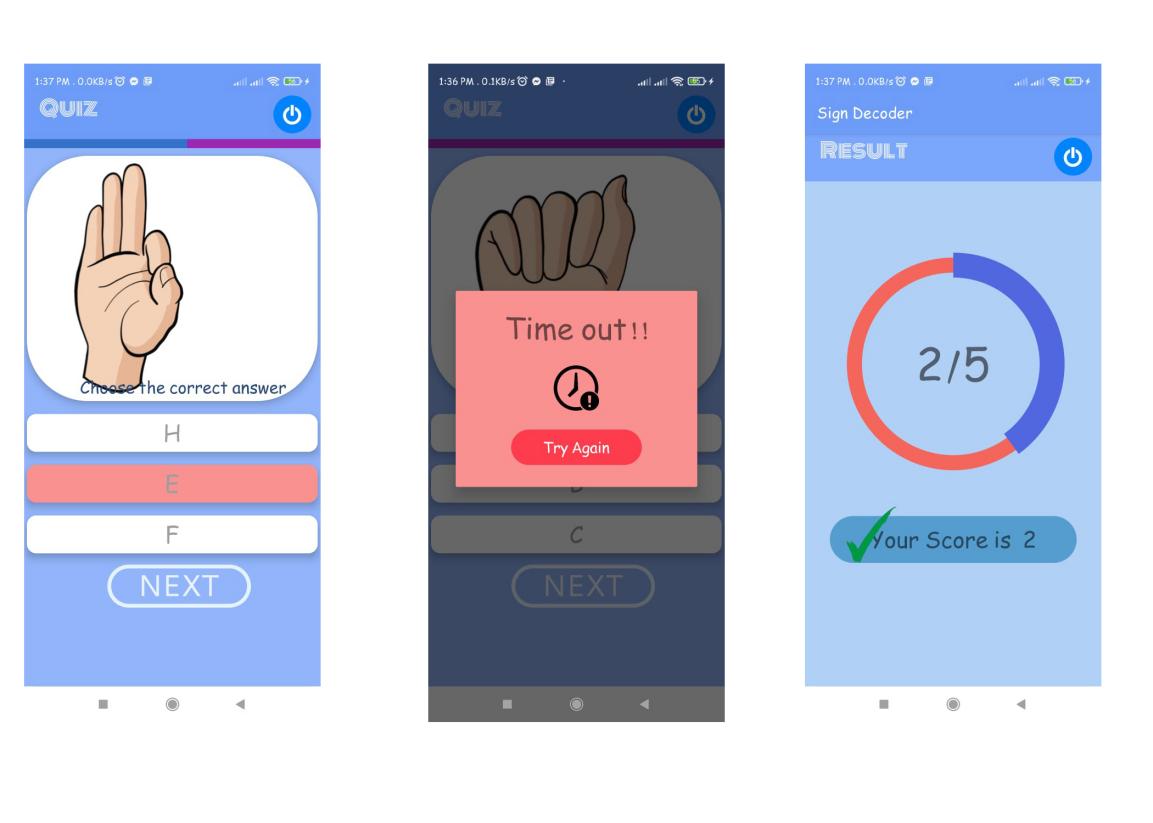
**Fig 5.4 :** Recognition Screen

**Dashboard:** Shows achievements

****

**Fig 5.5 :** Dashboard

**Quiz and Evaluation :** User can take quiz and evaluate himself

****

**Fig 5.3 : Quiz** Screen

1. **Conclusion**

In this research, we have evaluated the performance of the proposed model on an Android Application in real time. In our work we have focused on hand sign recognition as well as hand sign tutorial. The development of this application and the public sharing of it might minimize the restrictions of future study on it. This report presents a Sign Language Recognition System based on deep learning. The proposed models produce text output, which helps to eliminate communication barriers between and deaf and dump people. In future, we wish to use our dataset to get better accuracy. We also wish to implement Bengali Sign Language recognition on our android app.

# References

**[1]** According to WHO : Over 5% of the world’s population – or 430 million people – require rehabilitation to address their ‘disabling’ hearing loss (432 million adults and 34 million children), <https://www.who.int/news-room/fact-sheets/detail/deafness-and-hearing-loss> [Accessed last: 27 December, 2021]

**[2]** Abdulwahab A. Abdulhussein Firas A. Raheem; Hand Gesture Recognition of Static Letters American Sign Language (ASL) Using Deep Learning; Engineering and Technology Journal, 2020, Volume 38, Issue 6, Pages 926-937

**[3]** Pahlevanzadeh M., Vafadoost M. and Shahnazi M. 2007. Sign language recognition. Signal Processing and Its Applications. ISSPA 2007. 9th International Symposium on. pp. 1-4.

1. [G. Anantha Rao](https://ieeexplore.ieee.org/author/37086349926); [K. Syamala](https://ieeexplore.ieee.org/author/37086350124); [P. V. V. Kishore](https://ieeexplore.ieee.org/author/37085376939); [A. S. C. S. Sastry](https://ieeexplore.ieee.org/author/37679767000) ; Deep convolutional neural networks for sign language recognition. Publisher: IEEE[Mar 2018]
2. Munib Q., Habeeb M., Takruri B. and Al-Malik H. A. 2007. American Sign Language (ASL) recognition based on Hough transform and neural networks. Expert Systems with Applications. 32: 24-37.