# American Sign Language (ASL) Detection using Deep Learning

## AIM -

To implement and analyze Vision based Systems for Hand Gesture Recognition using Deep Learning and build a Real-Time Interpreter having a user interface.

## Developed By a Team of -

Priyansh Bhatnagar (DTU)
Tarun Arora (DTU)
Shaurya Gupta (DTU)

# Introduction

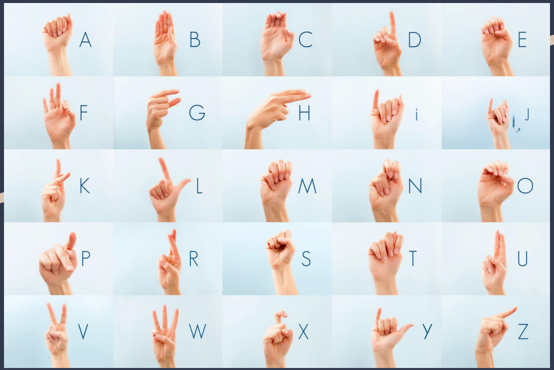
## Technologies Used

**Programming Language:** Python

**Software Platforms:** Jupyter Notebook/ Spyder (Python IDE)

**Machine Learning:** Tensorflow and Keras

**Vision:** OpenCV **User Interface(UI):** Tkinter



Only a small percentage of the population is aware of sign language. It is also not an international language, contrary to common perception. This makes communication between the Deaf population and the hearing majority even more difficult.

The purpose of this work is to contribute to the field of automatic sign language recognition.

The goal of this project is to build a convolutional neural network able to classify which letter of the American Sign Language (ASL) alphabet is being shown given an image of a hand gesture. This project is the first step towards building a possible sign language translator, which can take communications in sign language and translate them into written and oral language.

Most of the papers published generally train on the static images and test on the static images as it easy to collect static images and their well curated datasets. This project trains on the static images but tests on both static and dynamic images in a continuous frame, that is when prediction is done on a running video frame. Moreover, the papers referred have built a model with a maximum of 92-93% accuracy, this project also gives a better and more accurate model.

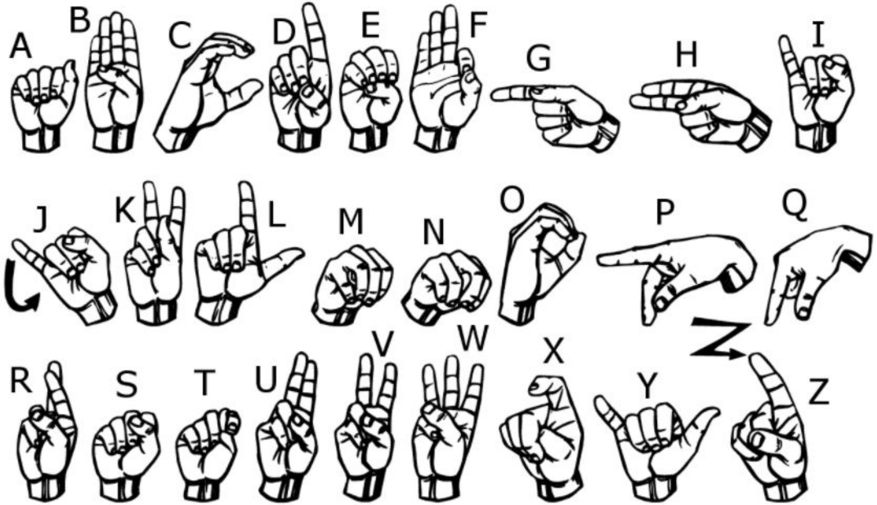
# Problem Formulation

## Objectives -

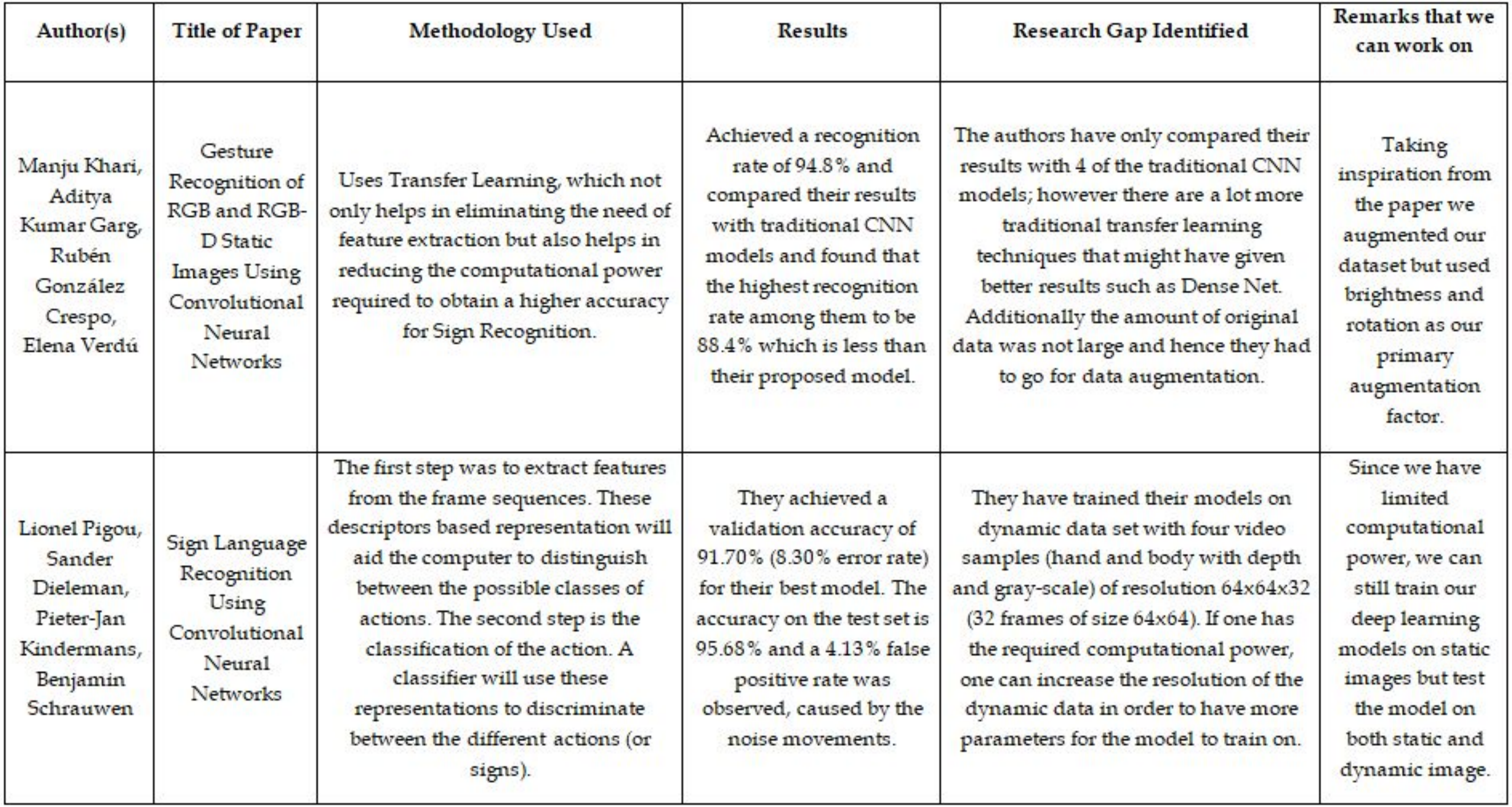
- To study the proposed and existing projects in the field through research papers.
- To implement Hand Gesture Sign Recognition Classifier that can detect the ASL alphabets.
- To form sentences using the detected alphabets that can help the deaf people to communicate and convert these sentences into speech using text-to-speech APIs.

# Dataset Used

The primary source of data for this project was the compiled dataset of American Sign Language (ASL) called the ASL Alphabet from Kaggle. The dataset is comprised of 87,000 images which are 200x200 pixels. There are 29 total classes, each with 3000 images, 26 for the letters A-Z and 3 for space, delete and nothing.



# LITERATURE REVIEW



| Author(s) | Title of Paper | Methodology Used | Results | Research Gap Identified | Remarks that we can work on |
|---|---|---|---|---|---|
| Manju Khari, Aditya Kumar Garg, Rubén González Crespo, Elena Verdú | Gesture Recognition of RGB and RGB-D Static Images Using Convolutional Neural Networks | Uses Transfer Learning, which not only helps in eliminating the need of feature extraction but also helps in reducing the computational power required to obtain a higher accuracy for Sign Recognition. | Achieved a recognition rate of 94.8% and compared their results with traditional CNN models and found that the highest recognition rate among them to be 88.4% which is less than their proposed model. | The authors have only compared their results with 4 of the traditional CNN models; however there are a lot more traditional transfer learning techniques that might have given better results such as Dense Net. Additionally the amount of original data was not large and hence they had to go for data augmentation. | Taking inspiration from the paper we augmented our dataset but used brightness and rotation as our primary augmentation factor. |
| Lionel Pigou, Sander Dieleman, Pieter-Jan Kindermans, Benjamin Schrauwen | Sign Language Recognition Using Convolutional Neural Networks | The first step was to extract features from the frame sequences. These descriptors based representation will aid the computer to distinguish between the possible classes of actions. The second step is the classification of the action. A classifier will use these representations to discriminate between the different actions (or signs). | They achieved a validation accuracy of 91.70% (8.30% error rate) for their best model. The accuracy on the test set is 95.68% and a 4.13% false positive rate was observed, caused by the noise movements. | They have trained their models on dynamic data set with four video samples (hand and body with depth and gray-scale) of resolution 64x64x32 (32 frames of size 64x64). If one has the required computational power, one can increase the resolution of the dynamic data in order to have more parameters for the model to train on. | Since we have limited computational power, we can still train our deep learning models on static images but test the model on both static and dynamic image. |

| Author(s) | Title of Paper | Methodology Used | Results | Research Gap Identified | Remarks that we can work on |
|---|---|---|---|---|---|
| M. M. Rahman, M. S. Islam, M. H. Rahman, R. Sassi, M. W. Rivolta and M. Aktaruzzaman | A New Benchmark on American Sign Language Recognition using Convolutional Neural Network | Proposed a novel Convolutional Neural Network (CNN) whose performance is evaluated in terms of recognition accuracy of alphabets and numerals. They have trained their model on 4 different available datasets. | Their proposed CNN improves the recognition accuracy of ASL reported by 9% than some existing prominent methods. | The authors have shown their training and validation accuracy to be around 99% which is a strong sign that it may be due to overfitting while the training phase of CNN. Additionally, the authors have not tested their model on dynamic image rather just given the validation accuracy on static data. | Test the performance of our model on real-time dynamic data as the ultimate goal of the project is its real-life applicability |
| Ching-Hua Chuan , Eric Regina , Caroline Guardino | American Sign Language Recognition Using Leap Motion Sensor | This paper is different from the previous ones in the way that they have used 3D Leap Motion Sensor and its API. The machine learning techniques used are k-nearest neighbor and support vector machine. | The experiment result shows that the highest average classification rate of 72.78% and 79.83% was achieved by k-nearest neighbor and support vector machine respectively. | The authors reported that major cause of the inaccuracy in the experiment result is the mislabled data from Leap Motion APIs. During the process of data collection, it may be observed many instances in which the hand and fingers in the visual feedback did not mimic the signer's hand. Since the raw data from the sensor are not publically accessible, authors mentioned that they plan to combine the sensor with a web cam, which provides a separate data source for hand gestures. | This research paper gave us a fresh perspective on how sign language data can be created and the collected data can be used further to train the model. The model trained directly on the sensor's data might have lower accuracy but is more applicable. |

# Methodology

Throughout this section, we discuss about the concepts in action, methodologies used, project setup and give a brief but extensive review of the project.

The dataset is trained on a CNN (Convolutional Neural Network) model that classifies the images into classes A-Z, Space, Nothing, Delete.

Convolutional Neural Network:

In deep learning, a convolutional neural network (CNN or ConvNet) is a class of deep neural networks, most commonly applied to analyzing visual imagery.

Layers/Blocks: After passing through a convolutional layer, the image becomes abstracted to a feature map, with shape (number of images) x (feature map height) x (feature map width) x (feature map channels).

Pooling layers: Pooling layers reduce the dimensions of the data by combining the outputs of neuron clusters at one layer into a single neuron in the next layer. There are several non-linear functions to implement pooling among which max pooling is the most common. We have used Max-Pooling as our Pooling layer.

Dense Neural Network: After several convolutional and max pooling layers, the high-level reasoning in the neural network is done via fully connected layers. Neurons in a fully connected layer have connections to all activations in the previous layer, as seen in regular (non-convolutional) artificial neural networks.

Moreover, we have used ‘categorical cross entropy’ as our loss function as this is a multi class classification problem.

## Validation Technique

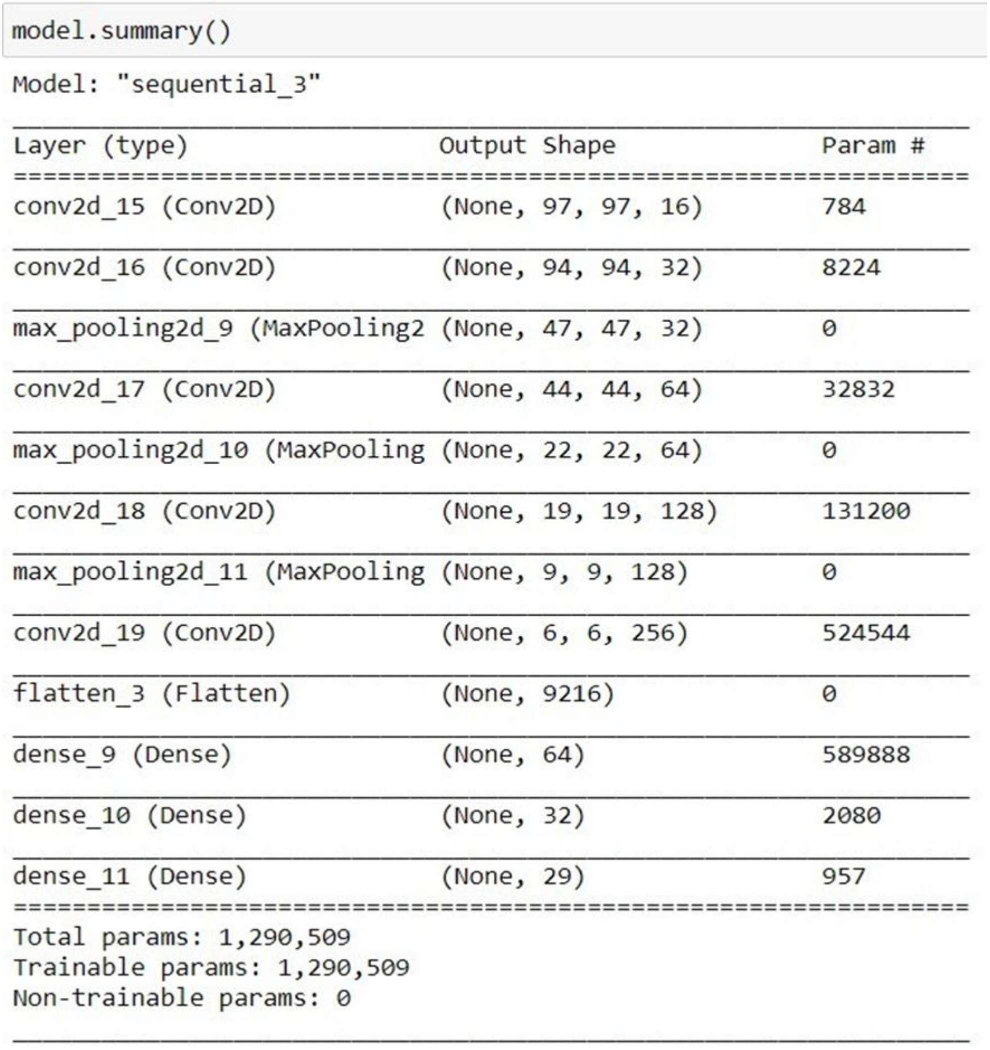
We will be using *Hold-Out* technique for validation.

# Working Methodology

The major working as the topic infers is based on deep neural networks for determining the hand sign presented through the real time video.

A custom deep architecture is built and experimented upon to make prediction in real time scenario and allow user to form texts for the same.

Though just using a custom deep network won’t be that big of a novelty, thus further novelty is pushed by further including a segmented input of the user hands.



```
model.summary()
```

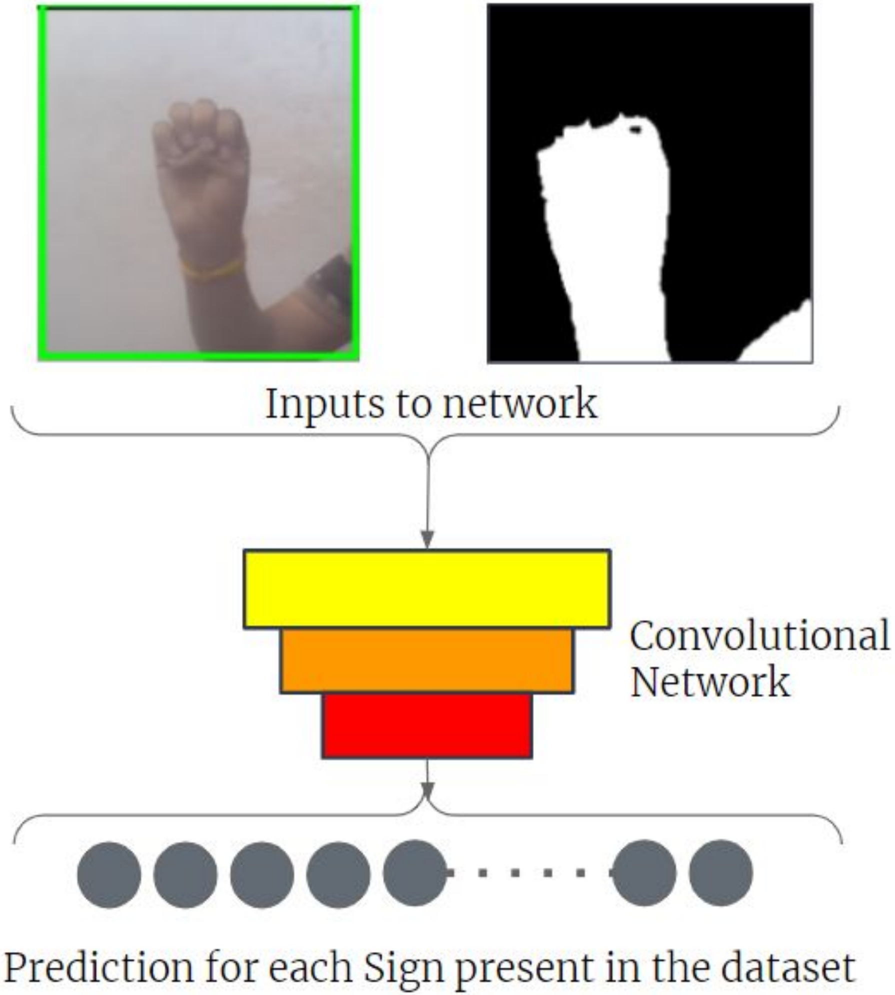
```
Model: "sequential_3"
_________________________________________________________________
Layer (type)                 Output Shape              Param #   
=================================================================
conv2d_15 (Conv2D)           (None, 97, 97, 16)        784       
_________________________________________________________________
conv2d_16 (Conv2D)           (None, 94, 94, 32)        8224      
_________________________________________________________________
max_pooling2d_9 (MaxPooling2 (None, 47, 47, 32)        0         
_________________________________________________________________
conv2d_17 (Conv2D)           (None, 44, 44, 64)        32832     
_________________________________________________________________
max_pooling2d_10 (MaxPooling (None, 22, 22, 64)        0         
_________________________________________________________________
conv2d_18 (Conv2D)           (None, 19, 19, 128)       131200    
_________________________________________________________________
max_pooling2d_11 (MaxPooling (None, 9, 9, 128)         0         
_________________________________________________________________
conv2d_19 (Conv2D)           (None, 6, 6, 256)         524544    
_________________________________________________________________
flatten_3 (Flatten)          (None, 9216)              0         
_________________________________________________________________
dense_9 (Dense)              (None, 64)                589888    
_________________________________________________________________
dense_10 (Dense)             (None, 32)                2080      
_________________________________________________________________
dense_11 (Dense)             (None, 29)                957       
=================================================================
Total params: 1,290,509
Trainable params: 1,290,509
Non-trainable params: 0
_________________________________________________________________
```

# Novelty of the Network architecture

Only using the RGB image through the network has been a traditional approach in computer vision problem formulation.

To further aid the learning process of the deep network we include the use of threshold segmented human hands to further ease the learning of the network and converge over the considered dataset quickly.

We use mean thresholding over the pixels to determine the threshold. Though a differentiating background is still required.



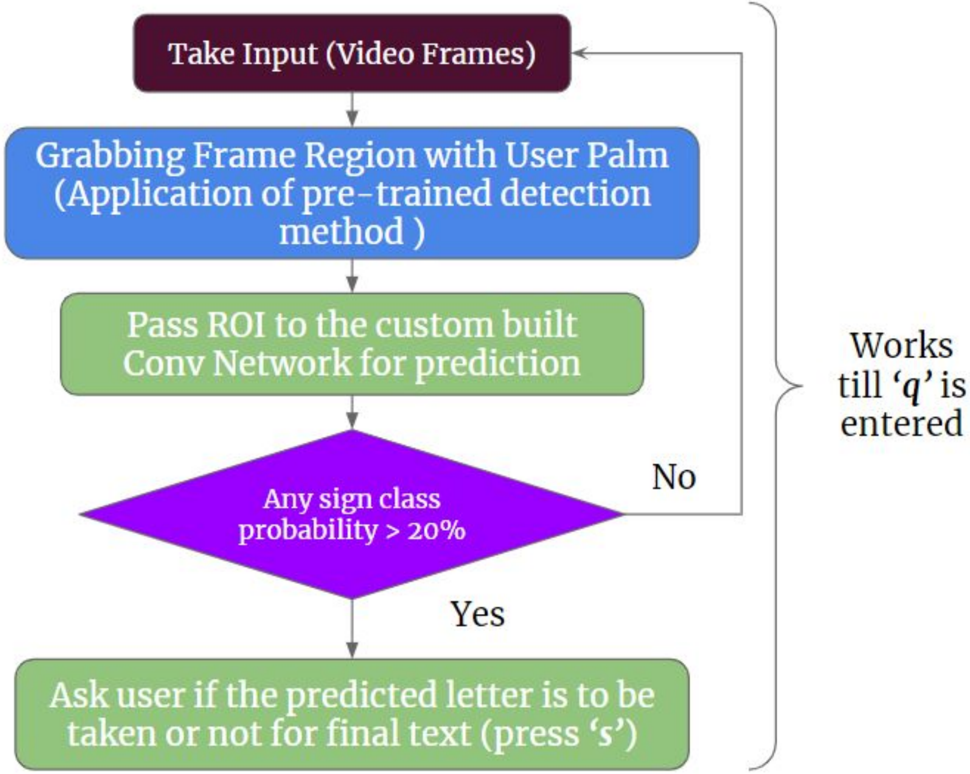
# Application Improvements

To improve the application shown, a pre-trained detection model is further used to follow the hand.

At any instant when the hand makes sign language predictions which have a probability of more than a threshold of 20%, the user is allowed to enter his/her words in the white pane through the press of ‘s’ key from keyboard.

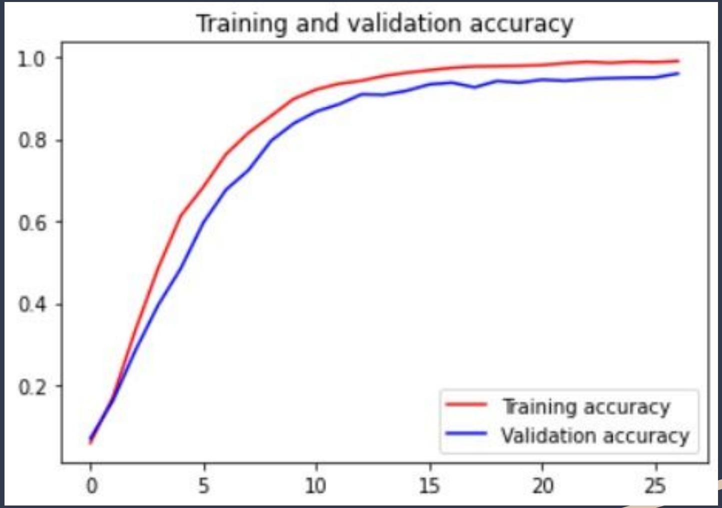
In presence of no-hand, space is considered as the character for entering into the text.

## Flow of Algorithm:


Take Input (Video Frames)
Grabbing Frame Region with User Palm (Application of pre-trained detection method )
Pass ROI to the custom built Conv Network for prediction
Any sign class probability > 20%
No
Yes
Ask user if the predicted letter is to be taken or not for final text (press ‘s’)
Works till ‘q’ is entered

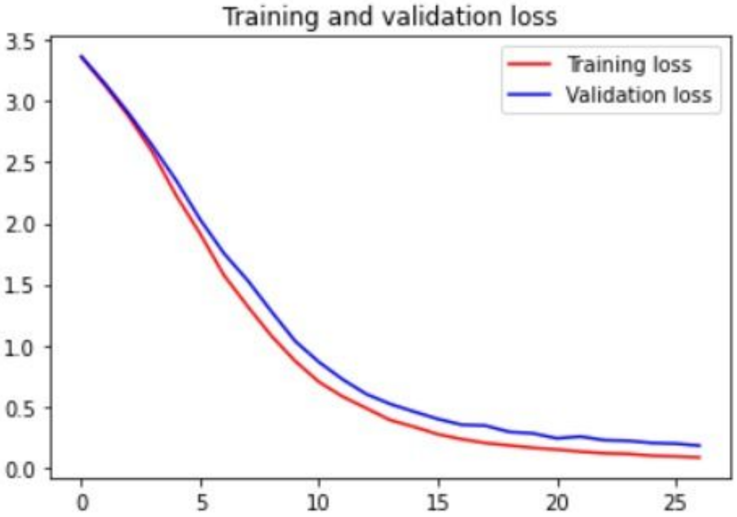
A video of the working project code has been shown. As we can see that changing the orientation and position of hands, results in prediction of a different label. A bounding box is created around the hand, and the pattern made by the pixels is processed and analyzed with the help of the Deep Learning Model that we have made. VIDEO LINK

# Results and Discussion

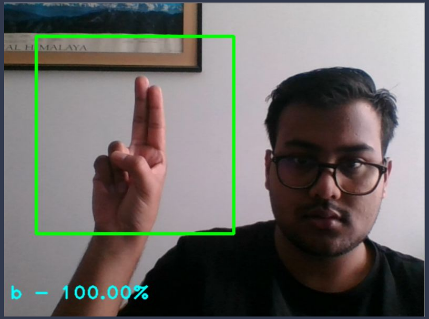


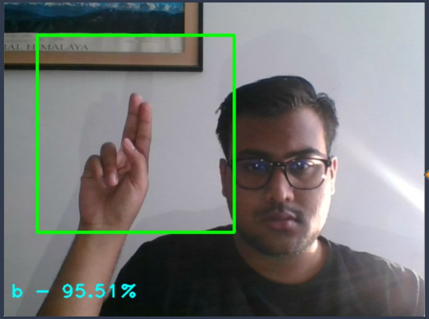
As we can notice in the above graph that our training accuracy improves from 10.5% to 98% in just 25 epochs. Whereas, our validation accuracy improves from 11% to 96%. This shows that the model has not overfitted as our validation accuracy is almost similar to the training accuracy.

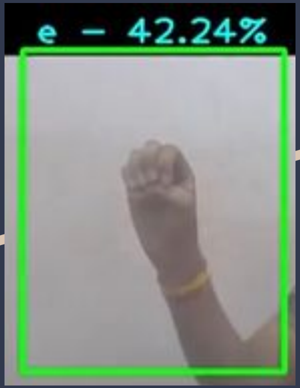
In the graph below, the training loss drops from 3.4694 to 0.0605 after 25 epochs. Whereas, the validation loss drops from 3.312 to 0.127. This shows are model is working well as the loss is decreasing.

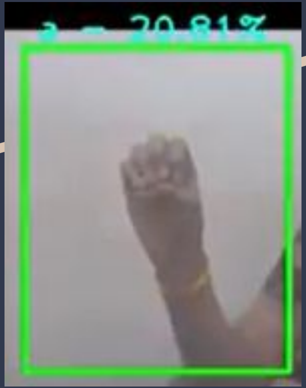


Current Computational Power::
8 GB RAM/CPU
i5-8300H CPU @ 2.30GHz Processor









# Critical Commentary

1. Due to limited computational power and memory, the images have been converted to 100*100, which is a considerably low resolution image these days. This makes the extracted features less informative and hence decreases the accuracy for identifying fine details.
2. Due to improper lighting conditions (which is quite common in a real world environment), the illumination across the pixels vary significantly making the extracted information inaccurate.
3. Due to similarity in the signs in dataset (like k-l, s-e-a-m-n, etc), there is significant overlapping in the pixels, making it difficult for the model to predict a particular label with large probability. In this case, Softmax function is used to determine the letter with the largest predicted probability.

# Summary

Through this project we have started with an aim to contribute to the society by assisting the deaf and the dumb to communicate effectively. We have applied the knowledge of ROBOT/COMPUTER VISION in order to predict what one has to say/communicate.

We have used Image Processing techniques to retrieve useful information out of it and finally extract the features using Convolution masks. Further the retrieved information is passed through a Custom made Deep Learning Model and weights  are trained on the American Sign Language (ASL)  Dataset consisting of 87,000 images with signs representing A-Z alphabets and 3 for space, delete and nothing.

# THANK YOU !!