

**Ramrao Adik Institute of Technology**

**Department of Computer Engineering**

***TE Mini Project Presentation***

***On***

***“Sign Language Recognition using Convolutional  
Neural Network”***

***By***

**19CE5503**

**19CE5502**

**18CE7013**

**Kunal Kamble**

**Pranit Jadhav**

**Atharva Shanware**

**Guided by**

**Mrs Pallavi Chitte**

# Introduction

---

- The number of deaf-mutes in the country are roughly calculated between 1.8 million and 7 million. (The wide range in population estimates exists because the Indian census doesn't track the number of deaf people — instead, it documents an aggregate number of people with disabilities.)
- Sign language substantially facilitates communication to people with such disability.
- However, there are only 2,50,000-5,00,000 speakers which significantly limits the number of people that they can easily communicate with.
- In order to diminish this obstacle and to enable dynamic communication, we present an sign language recognition system that uses Convolutional Neural Networks (CNN) in real time to translate a video of a user's ASL signs fingerspelling into text.
- Our problem consists of three tasks to be done in real time:
  - Obtaining video of the user signing (input).
  - Classifying each frame in the video to a letter.
  - Reconstructing and displaying the most likely word from classification scores (output).

# Problem Statement

---

The goal of this problem statement is to develop a system that captures the image through the live webcam only if a particular gesture is present and give the text or letter associated with the gesture. The system shall accept input of a static gesture through the webcam, preprocess the image, feed it to the CNN model and display the text as an output. The system only gives output for the ASL fingerspelling. Our problem consists of three tasks to be done in real time:

1. Obtaining video of the user signing (input).
2. Classifying each frame in the video to a letter.
3. Reconstructing and displaying the most likely word from classification scores (output).

From a computer vision perspective, this problem represents a significant challenge due to a number of considerations, including:

- Environmental concerns (e.g. lighting sensitivity, background, and camera position)
- Occlusion (e.g. some or all fingers, or an entire hand can be out of the field of view)
- Sign boundary detection (when a sign ends and the next begins)
- Coarticulation (when a sign is affected by the preceding or succeeding sign)

The system shall recognize the specified letter through the webcam. It shall also form words and sentences using custom gestures. Then after confirming the sentence it will convert the entire sentence into speech, and also translate the sentence in the specified language and that into speech too. The aim of this problem statement is to close the communication gap faced by the people with hearing and speech disability. The proposed system is limited to static gestures which can only recognize ASL fingerspelling but can be later updated for ISL, JSL and also dynamic hand gesture and two hand gestures.



## Objectives

---

The Sign Language Recognition Software is developed to solve the problem of communication between the people that are unable to hear and speak. The problem occurs because these people communicate with help of hand gestures that are not known to the normal people. So with the help of Sign Language Recognition Software, hand gestures performed by the D&M people are converted into text and also spoken words. The Software also provides a feature to convert the text into various other languages like chinese, japanese and many more. In order to address this problem and to perform dynamic communication, we present a sign language recognition system that uses Convolutional Neural Networks (CNN) to translate a video of a user's ASL signs fingerspelling into text.



## Literature Survey of existing system

---

- Sign language recognition is a topic which has been addressed multiple times and is not new.
- Over the last few years, different classifiers have been applied to solve this problem including linear classifiers, neural networks and Bayesian networks.
- Singha and Das obtained accuracy of 96% on 10 classes for images of gestures of one hand using Karhunen-Loeve Transforms.
- Starner and Pentland used a Hidden Markov Model (HMM) and a 3-D glove that tracks hand movement.
- Since the glove is able to obtain 3-D information from the hand regardless of spatial orientation, they were able to achieve an impressive accuracy of 99.2% on the test set.
- The most relevant work to date is L. Pigou et al's application of CNN's to classify 20 Italian gestures from the ChaLearn 2014 Looking at People gesture spotting competition.
- They use a Microsoft Kinect on full body images of people performing the gestures and achieve a cross-validation accuracy of 91.7%.



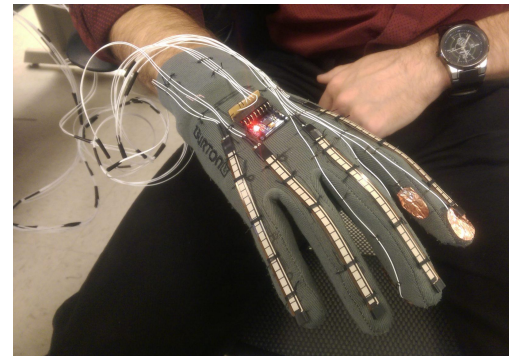
## Limitations of existing system

---

- Most of the existing systems require a High-Tech glove with motion sensors to capture the gesture in 3 dimensions or even a Microsoft Kinect, both of which are pretty expensive.
- These also impose scalability issues due to the equipment dependency.
- Benefits of portability has to be compromised, with the equipment having to be carried everywhere where the system is to be used.



Microsoft Kinect



Motion sensor gloves

# Proposed Methodology

---

This is a vision based system, it uses only a webcam for its functioning. This eliminates the use of equipment such as flex sensors, kinect, etc for interaction. So all the user has to do is place the gestures in front of the camera.

## Data Preprocessing:

Before feeding the data to the CNN model it is necessary to preprocess the data. This enables the CNN model to recognize the labels more accurately. There is also the need to eliminate the background, as it can cause hindrance to the recognition system. So the data preprocessing takes place as follows:

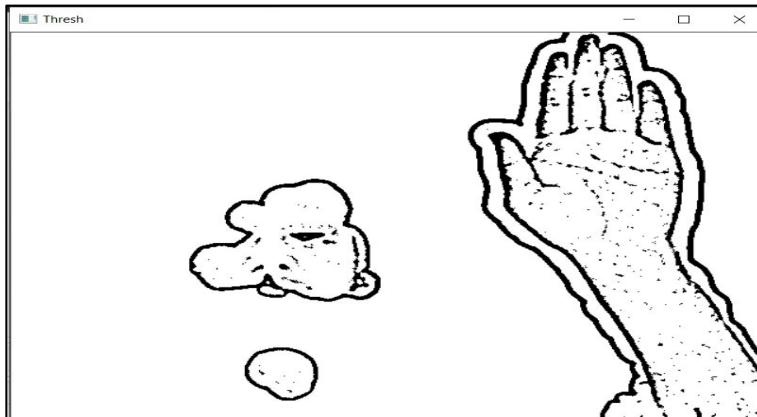
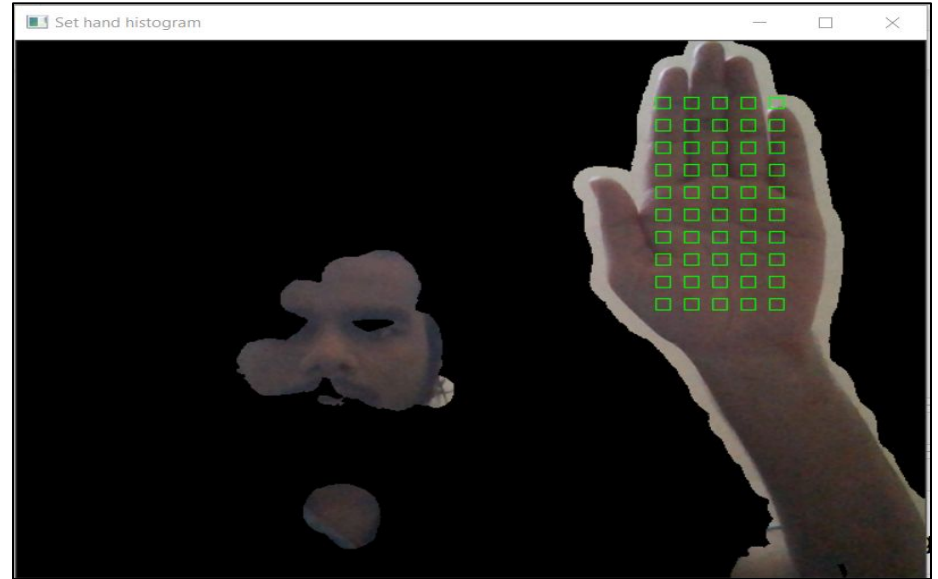
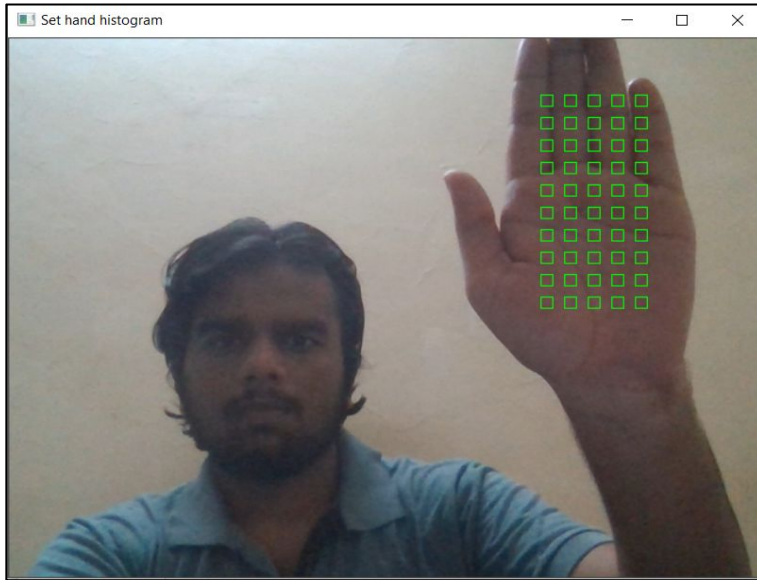
- Set borders: The entire image is not considered for recognition; only a part where the hand is supposed to be placed is cropped.
- Skin Masking: Skin masking is applied so that only the hand is visible and all the other background noises are eliminated. The cropped image is converted to HSV and the histogram of that image is calculated. The histogram is used to find the features of the image using an opencv function `calcBlackProject()`. This function is used to find the features of the image, in this case it is used to find the flesh color areas in the image.

## Removing noise and Smoothening:

- Adaptive Thresholding: In simple thresholding, the threshold value is global, i.e., it is the same for all the pixels in the image. Adaptive thresholding is the method where the threshold value is calculated for smaller regions and therefore, there will be different threshold values for different regions. In OpenCV, you can perform Adaptive threshold operation on an image using the method `adaptiveThreshold()`.
- Gaussian Blur: Gaussian blur is applied to the image which helps in extracting various features of image from ROI. ADAPTIVE\_THRESH\_GAUSSIAN\_C: The threshold value is a gaussian-weighted sum of the neighbourhood values minus the constant C.



# Proposed Methodology



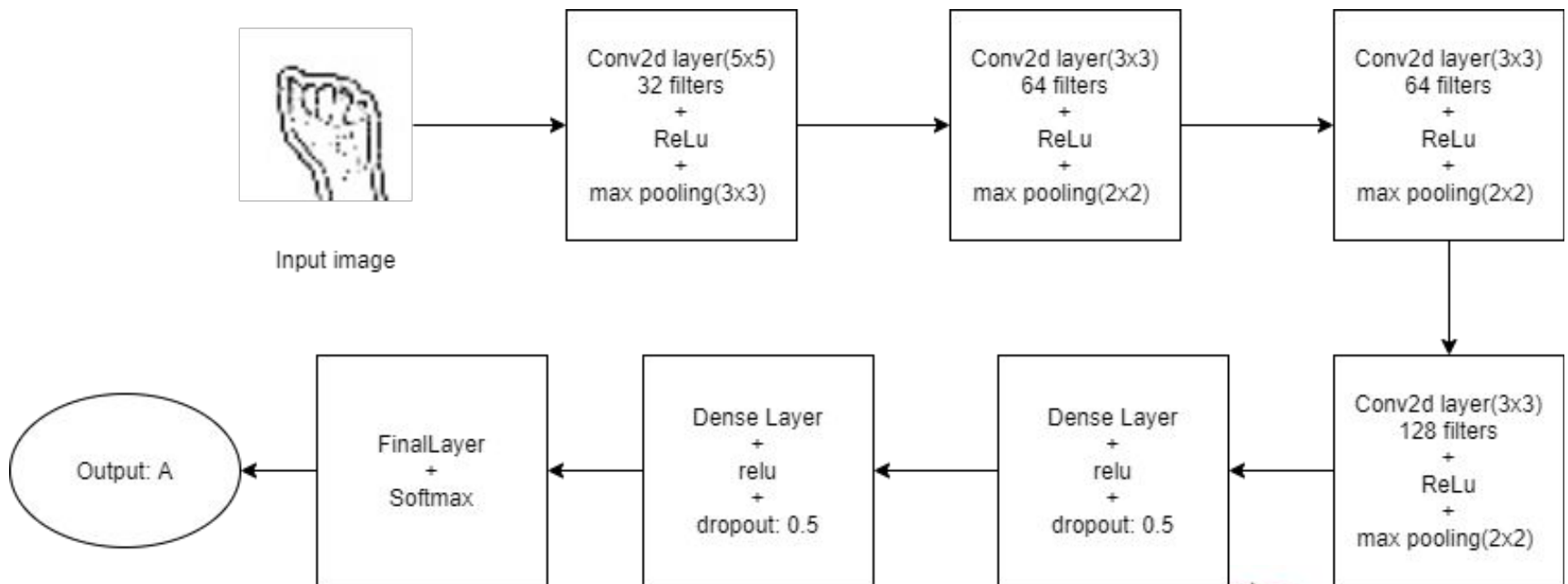


# Proposed Methodology

## Convolutional Neural Network Model:

A Convolutional Neural Network (ConvNet/CNN) is a Deep Learning algorithm which can take in an input image, assign importance (learnable weights and biases) to various aspects/objects in the image and be able to differentiate one from the other.

The CNN model is fed with the preprocessed dataset of ASL fingerspelling. The architecture used for the CNN model consists of 32, 64, 64, 128 bit architecture, the filter used in the convolution layer is of size 5x5 in the first layer and 3x3 in the rest. Max pooling is also applied with the filter size of 3x3 in the first layer and 2x2 in second.



# Implementation

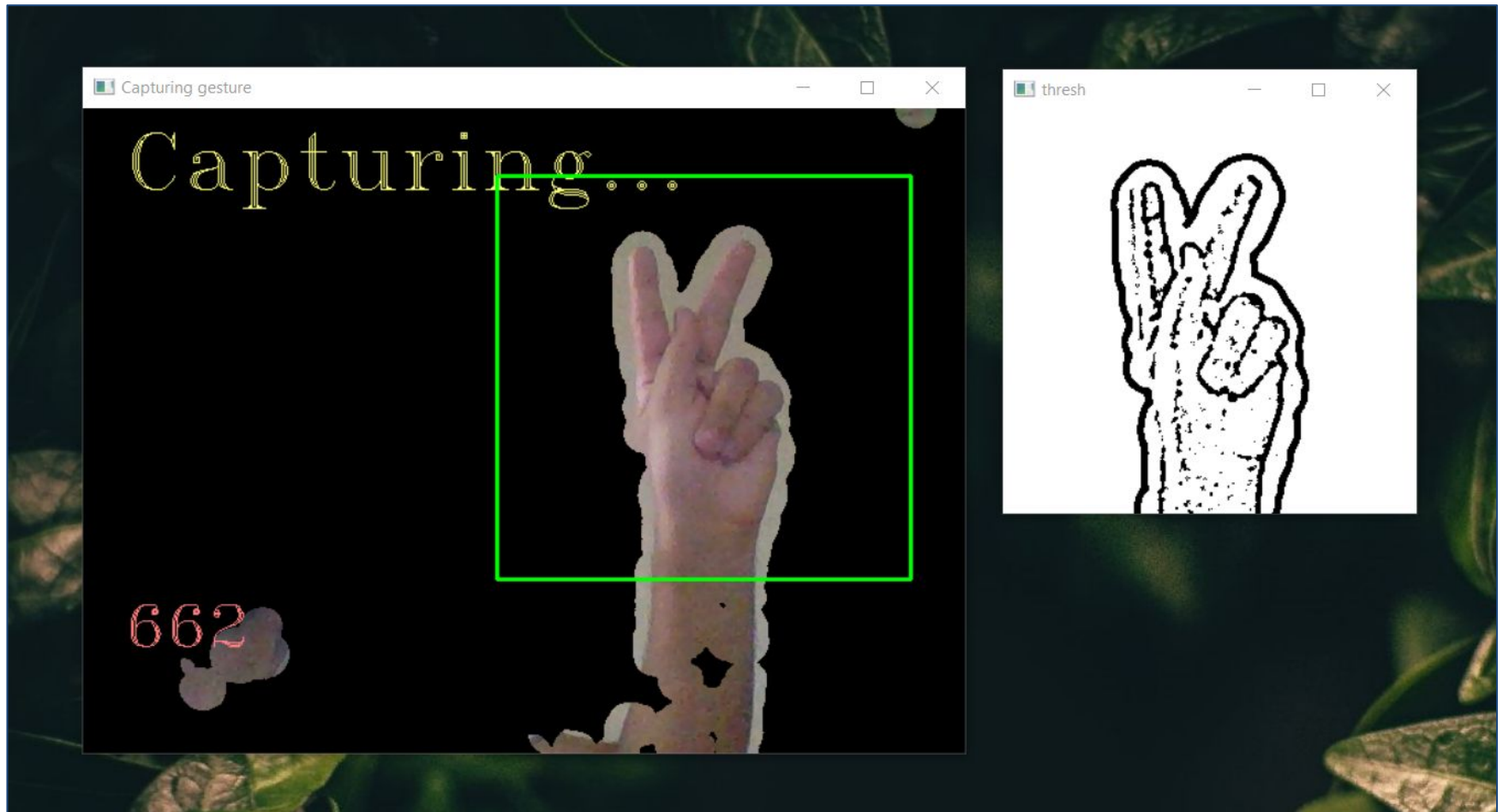
---

The system is implemented entirely with python. Tkinter is used to implement the GUI. The input image is preprocessed using one of the many image processing and computer vision libraries supported by python: OpenCV. The CNN model is implemented using tensorflow and keras library.

The steps followed for implementation:

1. **Dataset Generation:**  
Due to the lack of availability of the raw images of ASL fingerspelling matching the preferred requirement database was created using opencv and python. 6000 images for the 29 classes which includes alphabet a to z, custom gestures for space and full stop were captured using python open cv and webcam. The 6000 images for each class were divided into 3 sets of hands each containing 2000 images to avoid overfitting. Total 174000 images are contained in the dataset. The images are preprocessed while capturing hence the dataset contains preprocessed images.
2. **Create train and test data:**  
The dataset created is divided into train and test data. Pickle files for test, train images and train, test label is created. The train pickle files contain 80% Images and the test pickle files contain 20% images.

# Implementation



# Implementation

---



3. Gestures present in dataset:
4. Train the CNN model:  
The data is trained in convolution neural networks using keras. Convolutional neural networks (ConvNets or CNNs) are more often utilized for classification and computer vision tasks. Convolutional neural networks now provide a more scalable approach to image classification and object recognition tasks, leveraging principles from linear algebra, specifically matrix multiplication, to identify patterns within an image. That said, they can be computationally demanding, requiring graphical processing units (GPUs) to train models.
5. GUI Interface:  
Interface is implemented using python tkinter.

# Implementation

---

## 6. Implementing Sentence formation:

The output section is divided into 4 parts, predicted letter, word, sentence and translated sentence.

If the accuracy returned by the model for the particular frame is greater than 96 only then the predicted class is returned. When a predicted class is returned the counter for that class in the dictionary is incremented by 1. When the counter of the specific class becomes ten that class letter is displayed in the predicted letter section and that letter is appended to the word and the dictionary is cleared. When the custom gesture for space gesture is used the word is appended to the sentence.

## 7. Implementing text to speech:

The recognized text and the sentence formed can be given output in the speech form using pyttsx library of python.

## 8. Language translator:

A language translator is implemented using python googletrans library. The translated sentence is converted to speech using the python's playsound library.

## Results

---

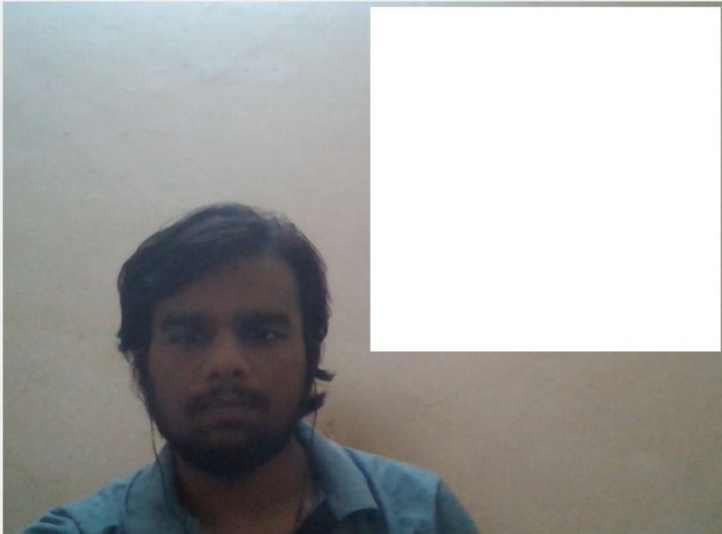
The Sign Language Recognition using CNN is implemented as a desktop application. The title window has two options to start the Sign Language Recognition application or to set the skin masking. Skin masking allows the user to set the histogram for skin masking according to the features of its skin color.



# Results

The start button navigates to the sign language recognition section. Which is divided into five parts; live cam feed, recognized text section, word formation and sentence formation section and translated text section.

SLR



### Text conversion:

Characters:

Words:

Sentences: normal click statement

### Translated text:

english click to select language



# Results

The input is taken from the live cam feed and the result is shown in the recognized text section. Two custom gestures are used to form words and a sentence. After the sentence formation is complete the sentence is translated into the specified language. During each letter recognition, sentence formation and translation the output is also given in speech format to that text.

SLR



**Text conversion:**

Characters: Words:

m--l hearts

Sentences:

dedicate your hearts !

**Translated text:**

あなたの心を捧げる !





## Conclusion

---

Sometimes there is a language barrier between people of different culture, nationality or ethnicity causing language barrier, but for the people with hearing and speech disability there is always a communication gap. Sign Language Recognition bridges the gap by introducing a computer application in the communication path so that the sign language can be automatically captured, recognized and translated to text for the benefit of deaf and mute people. This system uses CNN model which achieved 98.79% accuracy. It also eliminates the problem of background noise. The system pre-processes the image to the required nature for it to be fed into the model. The system is an approach to ease the difficulty in communicating with those having speech disabilities. The amount of training and validation loss observed with the proposed CNN architecture was less.

# References

---

- [1][IRJET-V7I1155.pdf](#)
- [2][What are Convolutional Neural Networks? | IBM](#)
- [3][A Comprehensive Guide to Convolutional Neural Networks — the ELI5 way | by Sumit Saha | Towards Data Science](#)
- [4][Teena Varma, Ricketa Baptista, Daksha Chithirai Pandi, Ryland Coutinho “Sign Language Detection using Image Processing and Deep Learning”, \(2020\).](#)
- [5][Brandon Garcia, Sigberto Alarcon Viesca “Real-time American Sign Language Recognition with Convolutional Neural Networks”, \(2017\).](#)



# Thank You



**D Y PATIL**  
DEEMED TO BE  
**UNIVERSITY**  
— **RAMRAO ADIK** —  
INSTITUTE OF TECHNOLOGY  
NAVI MUMBAI