

## **ABSTRACT**

The American Sign Language (ASL) is a standard communication methodology, which is propagated widely by schools of the deaf & dumb community for an ease of communication. There is a need to help people communicate with each other in a more accessible and intuitive way. Hence, with the help of technology like neural network (CNN) and computer vision (opencv), this project bridges that gap. It is a camera- based sign language recognition system which is highly useful for the deaf & dumb people, by converting sign language gestures to text data.

This software is acting as a mediator between two people who want to communicate with each other in American Sign Language. It interprets the signs made by different hand gestures and also forms accurate sentences in live video feed using the trained model. The system is fast enough to catch up with communication in real time and provides a decent accuracy at the same time. It is also be flexible enough to provide sentence formation feature as well features like edit, remove etc. to minimize human error.

# Chapter1: Introduction

---

## INTRODUCTION

### American Sign Language

ASL also known as American Sign Language originated in the early 19th century in the American School for the Deaf (ASD) in West Hartford. Since then, ASL use has propagated widely by schools for the deaf & dumb community for an ease of communication.

There was need of a technology which can make the sign language interpreting process easier. One of the earliest attempts in creating the system was by Mr. Pugeault and Mr. Bowden in 2011. They developed a classifier for 24 letters of American Sign Language.

Since, it is a standard collection of hand gestures which represents several characters and Alphabets. Hence it has gained popularity and systems are being devised in this area to improve the communication process with the help of technology.

### 1.1 PRESENT SYSTEM

This project aims to bridge the gap between the deaf and dumb people. The basic idea of this project is to make a system using which these community people can use to significantly communicate with all other people using their normal hand gestures. The system requires the background to be perfectly white. The project uses trained CNN model to identify, standard English alphabetic sign language used by the deaf people to communicate and converts them into text so that normal people can understand.

Using technologies like opencv and convolutional neural network (deep learning), which is a subset of Artificial Intelligence a system is devised such that when a hand gesture is made in live camera, it predicts the accurate sign language or hand gesture according to standard American sign language format. It also ensure that the system ignores the minimal hand gestures error as every person has different hand size and may slightly make different hand gestures. Hence, angle, shape, size are some of the features that are taken care of. For that the model is trained in a diversified dataset, which is self created and contain appropriate number of images per gesture for a proper functioning of the system. Data of more than one person is taken to maintain diversity and each gesture is having 750 train images and 250 set of test images. The American sign language interpreter predicts 26 different sign languages with a good accuracy and also help in sentence formation, edition and deletion. All this happens on real-time basis. For features like removal and deletion I have added 4 extra gestures in the dataset. The extra gestures are respectively for blank space, one letter removal, entire sentence removal and an additional no gesture dataset. This additional data is for situation when no gesture is made. It is like default sign so that our model will not make random false prediction. This a software which can act as a mediator between two people who want to communicate with each other in American sign language. It interprets the signs made by different hand gestures and also form accurate sentences in live video feed using the software. The system is fast enough to catch up with communication in real time and provides a decent accuracy at the same time. It is also be flexible enough to provide sentence formation feature as well features like edit, remove etc. to minimize human error.

## **1.2 PROPOSED SYSTEM**

Sign Language is a primary means of communication for many deaf and dumb people. They make different hand gestures which represents the signs. But this sign language is only understood by very few people.

There are situations when one person who is unable to speak and his or her only means of communication is the sign language, wants to communicate something to

another person who does not recognize the gestures. To communicate, the other person will either need to learn the hand gesture or make a guess.

Develop a software which can act as a mediator between two people. It should interpret the signs made by different hand gestures and also form accurate sentences in live video feed using the software. The system should be fast enough to catch up with communication in real time and provide a decent accuracy at the same time. It should also be flexible enough to provide sentence formation feature as well features like edit, remove etc. to minimize human error.

Using technologies like deep learning, which is a subset of Artificial Intelligence a system needs to be devised such that when a hand gesture is made in live camera, it should predict the accurate sign language or hand gesture according to standard American sign language format. Also, ensure that the system should ignore the minimal hand gestures error as every person has different hand size and may slightly make different hand gestures. Hence, angle, shape, size are some of the features that need to be taken care of. So, a diverse dataset is appreciated.

The model should be trained in a diversified dataset, which should be self created and contain appropriate number of images per gesture for a proper functioning of the system. The American sign language interpreter will predict 26 different sign languages with a good accuracy and also help in sentence formation, edition and deletion. All this will happen on real-time basis. For features like removal and deletion add 4 extra gestures in your dataset. The extra gestures will be respectively for blank space, one letter removal, entire sentence removal and an additional no gesture dataset. This additional data is for situation when no gesture is made. It is like default sign so that our model will not make random false prediction.

The project should be a camera-based sign language recognition system which would be in use for the deaf for converting sign language gestures to text data. The objective

is to design a solution that is intuitive and simple. Communication for the majority of people is not difficult and an integral part of life. It should be the same way for the deaf and dumb community.

## Chapter 2 : System Design

---

### 2.1 SYSTEM FLOWCHART

The data is first collected thorough applying threshold and converted as black and white image. Then we train the data using convolutional neural networks. Finally we predict the gestures using the trained data. Steps involved in design and planning of system are –

### 2.2 SYSTEM DESIGN AND PLANNING

- **Research on algorithm**

CNN was the most suitable algorithm for this project as it is used for object classification.

- **Set environment**

Anaconda jupyter notebook was set up and a separate environment other than base environment was created.

- **Install and import required libraries**

Installation of required libraries like keras, CNN and OpenCV. Then they were imported in the project.

- **Collect Data**

User defined dataset was collected for each gesture and stored in the dataset folder.

- **Splitting into train and test data**

Dataset was split into test and train folder . 750 images was stored in the train folder and 250 in the test folder for each gesture.

➤ **Pre-Processing of data**

Pre-processing is required for better accuracy. Resizing of data, gray-scale conversion and threshold was applied to assign pixel values.

➤ **Train the model**

Then we created a sequential model and trained our data using CNN. Convolution and max-pooling layer were applied twice for better feature extraction.

➤ **Prediction**

Hand gestures will be captured in live video feed and predict the sign languages in real time.

## **Dataset**

American Sign Language (ASL) is standard language that serves as the communication means of Deaf communities all across the globe. It is a set of 26 hand gestures known as the American manual alphabet, which can be used to spell out words from English.

This is widely popular among deaf and dumb community to communicate using a standard communication language.

We store self created dataset through live video feed and split the data into two parts train and test. The training data for each gesture consist of 750 images and test data contains 250 images.

## **Convolutional Neural Network**

We use CNNs over other NNs as we do not need to flatten the input images to 1D as they are capable of working with image data in 2D. This method helps in retaining the “spatial” properties of images. CNN works best with image like data. If we use ANN then processing of such huge no. of pixels will be computationally expensive.

A Convolutional Neural Network is a Deep Learning algorithm which takes in an input image, assign weights and biases to pixels in the image and extract features from them. This enables it to differentiate one image from the other.

It sequentially builds model layer by layer.

It consists of convolutional layer, max-pooling layer, flattening layer and dense layer.

Activation functions are applied in the layers according to the need of the features.

### **Layers of CNN:**

#### **➤ Convolutional Layer**

A filter is placed and updated by and weights are initialized randomly. SO we place the filter in top left of the image and it moves throughout the image. We take the average value of pixels and central value is updated. The filter may contain the positive and negative value as the weights are randomly initialized.

#### **➤ Normalization Layer**

In this layer we pass an activation function, normally relu so that the negative values becomes zero. As, negative pixel intensities are no use to our system.

#### **➤ Pooling Layer**

The pooling layer reduces the no. of pixels as it pools out the maximum value over which the window is placed. It does not repeat pixels unlike convolutional layer.



### ➤ **Fully connected Layer**

Then we flatten the result and feed it to a fully connected layer.

## **Python**

Python is a widely used programming language for the domain of Machine learning and AI. Because of it's easy to use, rich libraries, readability and fast features it is the most used programming language.

## **Artificial Intelligence and Deep Learning**

Artificial Intelligence is a vast field. Deep learning is the subset of AI. Unlike traditional machine learning, we do not have to provide the feature to the algorithm. It automatically extracts features from it. It is best for high computational data. CNN is a type of neural network on which deep learning is based.

## **Computer Vision**

Computer Vision is inspired by human eyes. It makes the process easier and faster as compared to traditional methods. Generally it is preferred for image like data. OpenCV is a computer vision library being used here, which helps in image extracting and processing.

## **Chapter 3 : Hardware and Software Details**

---

### **SPECIFICATIONS**

#### **Hardware Specification**

CPU - Dual Core 2.2 GHz

Processor – Intel Core i3 7<sup>th</sup> Generation

Ram - 4 GB minimum

OS - 64-bit Windows 10

#### **Software Specification**

Language - Python

Tool – Anaconda

IDE - jupyter notebook

Technologies – Artificial Intelligence

Libraries :

- cv2
- numpy
- os
- Sequential from keras.models
- Convolution2D from keras.layers
- MaxPooling2D from keras.layers
- Flatten from keras.layers
- Dense from keras.layers

## Software and Hardware Requirements

- The CNN and keras code needs to be written in a programming language, Since, we are using AI for our task. So, programming language such as R and Python is best suitable. In this project, I have implemented in Python as it is having rich library and many more benefits over R.
- Anaconda software was used and jupyter notebook for IDE. We installed the required libraries in Anaconda separate environment other than base environment.
- A CPU such as i7-7500U is capable of training an average of 115 examples/second. So for smaller tasks CPU is sufficient. However for complex tasks a GPU may be needed otherwise it will slow down the training process.
- A GPU can perform convolutional based operations on a batch of images of 128 or 256 images at once in just a few milliseconds. However, the power consumption is huge which leads to a total of 400W.
- In this system one epoch took approximately 7-8 minutes to run. Hence, an average of 25 epochs would need 3-4 hours in CPU.

## Chapter 4 : Implementation Work details

---

### 4.1 Real Life Applications

- This system is widely used among many American deaf and dumb communities and their organizations.
- Also it is practiced and taught in deaf and dumb European schools.
- A google translator app Gynosys uses deep learning neural network and computer vision for translating sign language.
- Many deaf people and dumb people use mobile applications which helps them translate the sign language and have a communication with ease with other people.
- The Hand Talk is an app which is lead by Hugo translates the ASL texts and also the brazillian sign language through the help of technology like Artificial Intelligence.
- Mimx3D sign language app is another popular android application available for free in google playstore.

### 4.2 Data Implementation and Program Execution

The code is executed in three phases.

#### 1. Collecting Data

Firstly, a dataset will be created of different hand gestures. They will be split into test and train part.

#### 2. Training

Then we will train our data using CNN. A Convolutional Neural Network is a feed forward NN just like ANN. It works best with image like data. We train the collected data through CNN layers, compiling it and fitting it into the model.

### **3. Prediction**

Hand gestures will be captured in live video feed and predict the sign languages in real time. The system will also form sentences.

## **Data Collection**

- create folders using os library
- draw a rectangle for extracting region of interest (roi) from each frame
- convert roi to grayscale image
- Feed the new roi to threshold function
- The threshold function gives entire black and white image using thresh\_binary method
- Write the image to the train and test folders

## **Train the data using CNN**

- **CREATE CNN MODEL**
  - Create Neural network sequential architecture
  - Add Layers (convolution, normalization, maxpooling) twice
  - Flatten the model and feed it to NN
  - Add Output layer

➤ COMPILE THE MODEL

optimizer

loss function

metrics

➤ PREPROCESS AND DATA AUGMENTAION

#ImageDataGenerator

rescale 0-255 to 0-1

shear range

zoom

horizontal flip

#train\_datagen.flow\_from\_directory

Generator generating augmenting images

resize image

color\_mode (gray scale)

class\_mode (categorical)

➤ TRAIN AND SAVETHE MODEL

classifier.fit\_generator

loss : during training

val\_loss : loss in validation

acc : during training

val\_acc : accuracy in validation

## Prediction

➤ Load Model

➤ Open Webcamra

➤ Create dictionary repressing each gesture

➤ Extract ROI

- Gray conversion
- Apply binary threshold
- `result=loaded_model.predict()`
- Predict

## Chapter 8 : Individual Contribution

---

**Aparajita Halder**

**(170396)**

I am the sole contributor of this project. This is a self made project created within a span of 2-3 months under the guidance of Ms. Sonal Shukla. I collected the dataset on my own. The dataset was contributed by my friends, family and myself. It was collected through live webcam of my laptop.

Then I pre-processed the collected data. I converted the image into gray scale conversion and stored in binary (black and white format), which will make the prediction process easier.

Then the collected dataset was trained through CNN. Convolutional and MaxPooling layers were applied to the images. The model was compiled and preprocessed. Then, it was fit into the model and then trained using 25 epochs. After training the model was saved into the data.

In the prediction process the trained model was used for predicting the outcomes and all the alphabets alongwith four extra hand gestures were stored in a data dictionary format in python so that I can easily extract and match with the gestures and thus, system makes a good prediction and also helps in sentence formation, edition, deletion and execution.

I presented it in front of the assigned panel the outcome of my project which follows proper system design and planning as per first phase of evaluation. The report was compiled and made by me under the guidance of my mentor. All the tasks from designing, planning and executing was contributed by me. I will ensure future enhancement and maintainability of this project.



# CONCLUSION

## LIMITATION

### 1. Fluctuation in prediction

Due to huge no. of classes and similar type of hand gestures there may be fluctuations in the model. Though, we can minimize the false prediction by extracting the gesture which has maximum no. of occurrence in a given time period.

### 2. Only predicts pre-defined gestures

The system will only be able to guess only the pre-defined gestures. If the user shows some other gesture it will predict some similar hand gesture.

### 3. Speed

It may take some time to predict the hand gesture and form a sentence than the user would have been able to communicate in a real life scenario.

## SCOPE

In future this project can be deployed in cloud and its benefits can be leveraged by not only specially abled people but also by normal people to have an ease of communication. This system can be used among many American deaf and dumb communities and their organizations. Also it can be practiced and taught in deaf and dumb European schools. It can also act as a teacher/translator. A google translator app Gynosys uses deep learning neural network and computer vision for translating sign language. Many deaf people and dumb people use mobile applications which helps them translate the sign language and have a communication with ease with other people. The Hand Talk is an app which is lead by Hugo translates the ASL texts and also the brazilian sign language through the help of technology like Artificial Intelligence. Mimx3D sign language app is another popular android application available for free in google playstore. Existence of such apps makes our belief in the future of this system more dominant.

## **FUTURE ENHANCEMENT**

### ➤ **Dataset Update**

There might be a need to add new hand gestures in future. The system is providing a feature so that users will be able to collect their own user defined hand gestures as per their requirement.

### ➤ **Optimization**

Measures will be taken to maximize efficiency and reliability of the system by updating code and updating library versions.

### ➤ **Computational Speed**

The speed of the system can be optimized by using GPU instead of CPU. If we implement the system in GPU then, we will be able to increase the accuracy of the model by increasing no. of epochs and dataset captured.