Sardar Patel Institute of Technology

Department of Computer Engineering

Sign language recognition and translation to text

May 2021

BY:

Shubham Goel 2019130015

Bhavya Shah 2019130056

Shubham Thakar 2019130065

Under the supervision of

Dr. Anant .V. Nimkar

ACKNOWLEDGEMENT

We are glad to publish “Sign Language recognition and translation to text” as our report and would like to take this opportunity to convey our heartfelt gratitude to our college for giving us this opportunity and to all those who assisted us in the completion of this project.

We are grateful to Dr. Anant V. Nimkar, our project guide, for his insightful and timely assistance throughout the many stages of our project. He guided us in every step right from the best way to approach a problem to the correct methodology of writing a report. He provided us with all the necessary resources and assistance as and when required. His encouragement, patience, and belief in our talents, as well as his flexibility in terms of working and reporting schedules motivated us to give our best.

Shubham Thakar
Shubham Goel
Bhavya Shah

Contents

1. Problem definition:

466 million people worldwide have impaired hearing loss (more than 5 percent of the world's population). Sign language is one of the oldest and most natural forms of language for communication for people having speech disabilities. Most people do not know sign language and interpreters are very difficult to come by.

To counter this problem we built a real time method using neural networks for **fingerspelling** based American Sign Language.

2. Literature survey

Over the past several years there has been a lot of research on the topic of hand gesture recognition. During our survey, we found many methodologies used to achieve the same. The techniques can mainly have 2 steps as:

1. Data acquisition
2. Gesture classification

Data acquisition

This is the step for taking input data from the user in the form of an image or electrical signals. There are mainly 2 approaches to it.

Use of sensory devices

1. In this approach, there are external sensors used to capture one's hand information. That can be a glove based approach or an electromagnetic sensor. Both do a good job of capturing high quality data but are expensive and not user friendly.
2. Our product should be available to all the user demographic so we chose not to use this method.

Vision based approach

1. In vision based methods, a camera is the input device for observing the signs of hands or fingers.
2. Vision Based methods require only a camera, hence these methods are cheaper and are more accessible.

Gesture classification

There many different ways in which gesture classification was done. One of the research papers used the Transfer learning model through GoogleLeNet . Another paper preprocessed data with Gaussian Mixture Model (GMM) then used convolutional neural networks for classification.

We used a transfer learning model called VGG16 for better results.

3.Market survey

There is a demo video present for sign language to speech and speech to sign using augmented reality. There is no production for this product. Another product named Gnosys was promised to be launched in 2019. There were no updates of the same.

In conclusion, there is no app in production currently available aimed at solving the problem of real time sign language to text conversion.

4. Scope of Work and Objectives

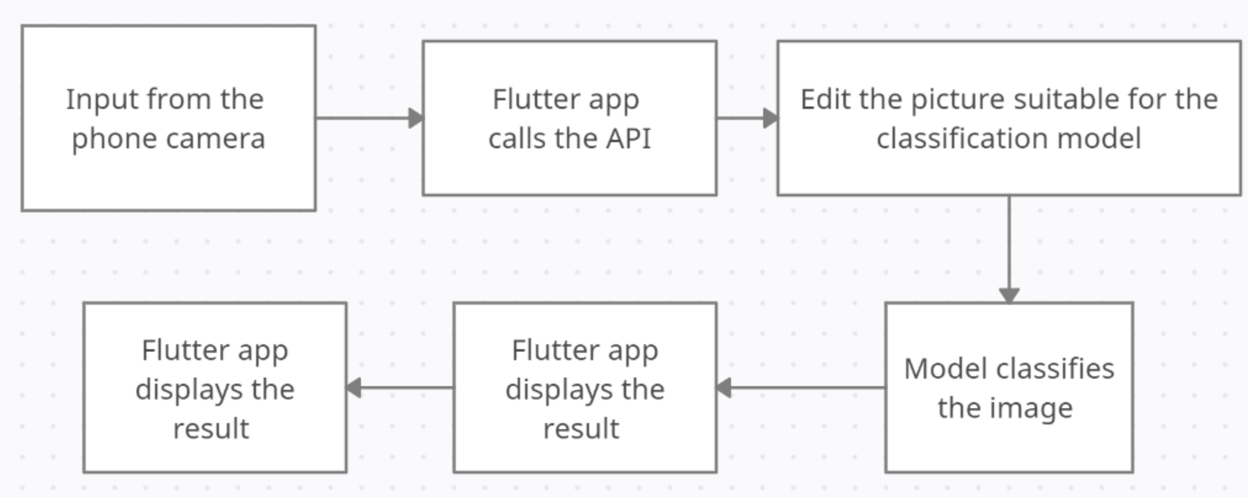
The app was created to give every person the ability to have smoother and more natural conversations with people who use sign language for communication.

Our app captures the images of your hand gestures and translates them to english alphabets. Along with this it provides a functionality to convert a string of text to sign language so that an average person who is curious to learn more about the ASL can do so.

The main objective is:

1. To successfully create a user friendly app that recognizes and converts sign language to text.
2. Also, provide the additional functionality of text to sign language conversion for people curious about sign language.

5. Block Diagram


Input from the phone camera
Flutter app calls the API
Edit the picture suitable for the classification model
Model classifies the image
Flutter app displays the result
Flutter app displays the result

Input of the hand gesture is taken through the application, it then calls an API and sends the image to it. The image is transformed into a format fit for the model. The image is then classified into 26 alphabets. This classification is then sent to the flutter application through the API.

6. Implementation

Techstack

- Python 3 - as the main programming language
- OpenCV - as the image processing library
- Django REST framework - for building the API
- Tensorflow deep learning framework - learning model
- Flutter - android application

Machine Learning :

Used VGG-16 model for Convolutional Neural Network.

Added a flattened layer along with two dense layers and trained it on 70.000 images which were resized to (64,64) pixels.

Used OpenCV to build a pipeline for real time detection of sign language.

Deployed this model on Django so that we can build a rest API that can be accessed by anyone.

Django Rest Framework :

This was used to allow us to deploy our model on a server and build a rest API so that we can use the Machine learning model in any application.

The API required a base64 encoded image in a post request.

Once a post request is made, the image would be cropped and resized. The image is then sent to the Deep learning model where a detection was made and sent back to the application which requested it.

Flutter Application:

We built an application for taking the input of hand gestures through a smartphone camera.

A. Sign to text

1. Taking the Input:

   This application takes the input of the hand gesture every second. These images were given in YUV format which had to be converted to RGB images.

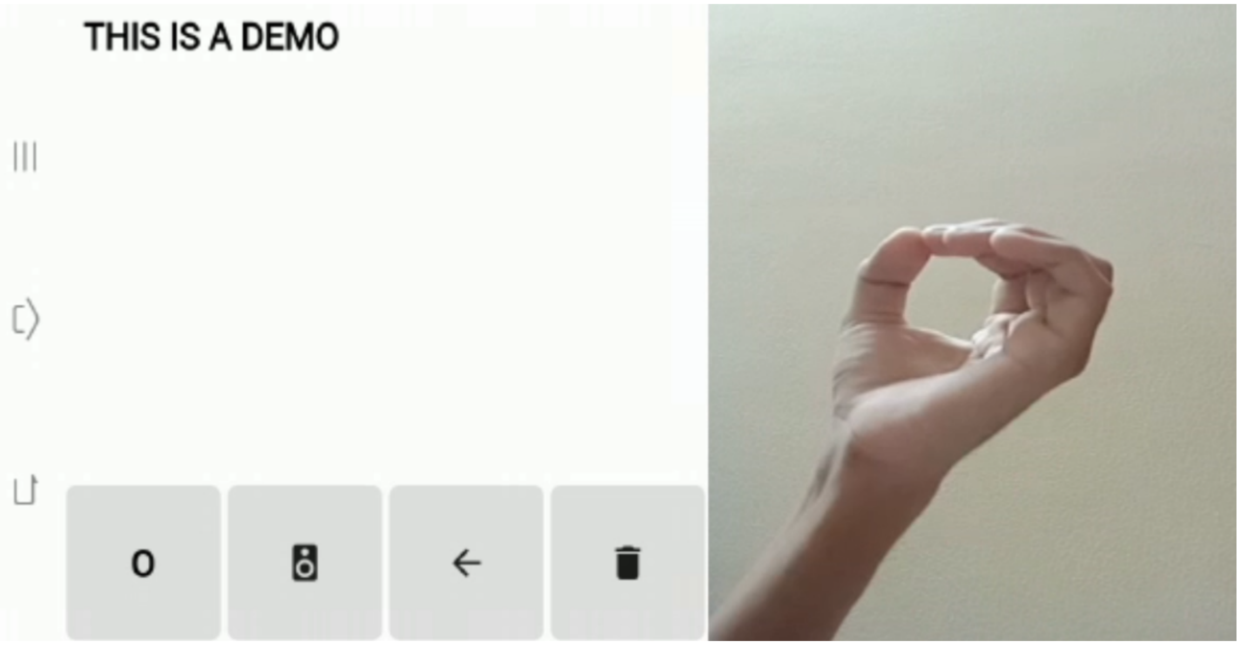
These converted images were converted into base64 format. It makes a POST request to the rest API with that image as represented in the text field.

Inside the Django rest API, the base64 image would be converted back to RGB, and then it would be cropped and resized.

The model would then make the prediction and send it back to the application which requested it.

2. Displaying the output

According to the RESPONSE received from the rest API we display the result in text form. Our application also has an option to speak out the received output build using the Flutter_Tts library.



THIS IS A DEMO

B. Text to sign

This module converts text input into a list of images for making the user understand how to sign a particular set of alphabets.

3. Taking the Input:

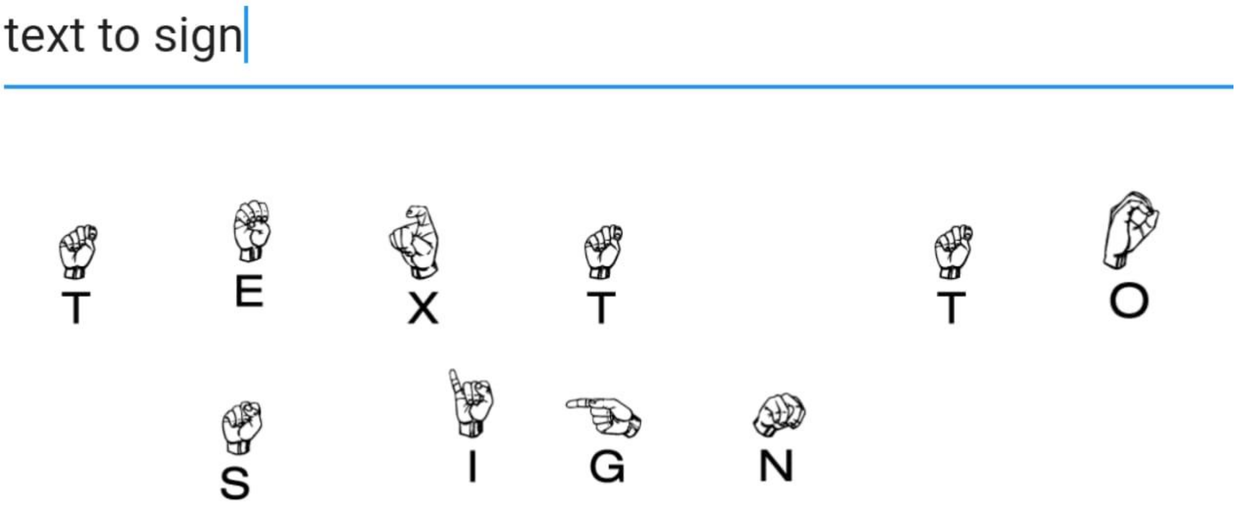
The required sentence will be added in the text box.

These sentences should have only lowercase letters and no punctuation marks.

4. Displaying the output

The output displayed would be corresponding to the sentence added in the text field.

The sentence added in the text field would be scanned from left to right and the corresponding image for each character would be shown.



text to sign

7. Timeline

Phase 1

In Phase 1, the basic outline for our project was prepared. Using the references from the market and literature survey the modules to be used in the project and the other important features were discussed and finalized. We started the process of data set generation for our model during this phase.

Phase 2

In Phase 2, the machine learning model was developed. We created an API using Django rest framework to input image data in base64 format and get an output as to which alphabet does the gesture represents.

Phase 3

In Phase 3, the integration of the API with the flutter application was done. Flutter application was developed. Added features like text to sign and sign to voice. Integration of the parts like the API, machine learning model and the flutter application was done.

8.Conclusion

A functional real time vision based American Sign Language recognition application was developed. We achieved this by keeping all the modules independent of each other for better reusability. Our model gave an accuracy of 100% and we created a restful API that can be consumed by any front end application to provide ASL to text or text to ASL functionality. During this project we learned about the machine learning models, building a restful API using Django and creating a flutter application for a better user experience.

9. Future Scope

Building a chrome extension to make sign-to-text conversion possible during online video conferences. This can be achieved by using our existing rest API for processing the input image.

References

[1] Sign-to-speech translation using machine-learning-assisted stretchable sensor arrays. Zhihao Zhou1,2, Kyle Chen2, Xiaoshi Li1, Songlin Zhang (2020).

[2]American sign language recognition and training method with recurrent neural network. Department of Industrial and Systems Engineering, The Hong Kong Polytechnic University, Hung Hom, Hong Kong, China (2021)

[3]Real-time American Sign Language Recognition with Convolutional Neural Networks/ Brandon Garcia Stanford University Stanford,2016 [4]Human hand gesture recognition using a Convolution Neural Network(2016)INSPEC Accession Number: 14692084 New Taipei, Taiwan

[5] Hand gesture recognition using a convolution neural network, 18-22 Aug. 2014, 10.1109/CoASE.2014.6899454 Hsien-I Lin; Ming-Hsiang Hsu; Wei-Kai Chen

[6] Article on NYU Tandon School of Engineering graduate students for demo app

[7] App that converts sign language into text promised to be released in 2019

[8] Dataset used