

Sign Recognition Project

Kanchan Patil
Lizzen Camelo

May, 2023

Contents

- 1 Introduction
- 2 Overall Description
- 3 Demo
- 4 LSTM
- 5 Acquisition
- 6 Training
- 7 Testing
- 8 Learnings and Challenges
- 9 Future Scope

Introduction

AI sign language detection systems enable conversations with speech and hearing impaired people. It plays the role of a translator, erasing the need for human translators.

Real world application

Integrating this AI tool in virtual conference calls to facilitate those conversations with people not versed in sign language.

Overall Description

1. Project Functions

- 1.1. Accepting input from webcam
- 1.2. Detecting the signs
- 1.3. Displaying correct output(translated sign)

2. Techstack

2.1. Interfaces/Tools

- 2.1.1. Jupyter
- 2.1.2. Webcam

2.2. Software dependencies

- 2.2.1. OpenCV
- 2.2.2. Mediapipe
- 2.2.3. Tensorflow
- 2.2.4. scikit-learn

Project Demonstration

LSTM

Our project uses the LSTM network. In concept, an LSTM recurrent unit tries to “remember” all the past knowledge that the network is seen so far and to “forget” irrelevant data. This is done by introducing different activation function layers called “gates” for different purposes.

Cons:

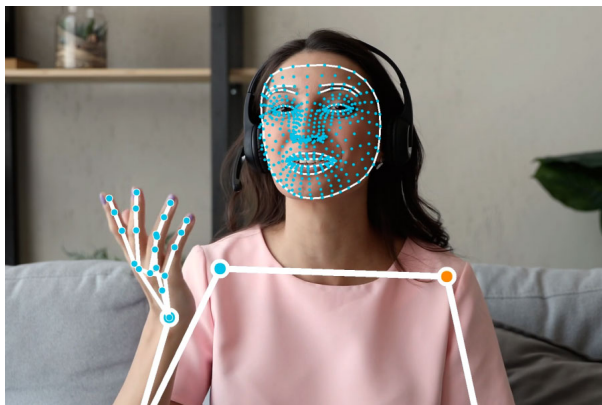
1. LSTMs take longer and more memory to train
2. LSTMs are easy to overfit

Data-set Acquisition

We use an online data-set consisting of numpy array files of extracted key points of various landmarks including pose, hand and face landmarks.

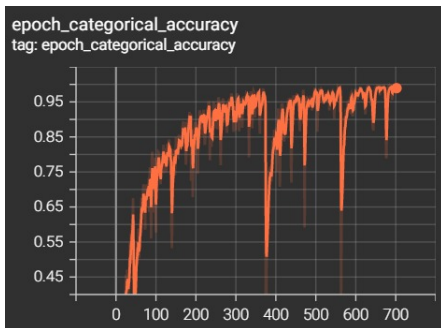
As of now the following are the signs that have been included:

1. The Alphabet(A-Z)
2. First 9 numbers(1-9)



Data-set Training

1. Trained for 800 epochs
2. Accuracy = 99.28%



Model: "sequential"

Layer (type)	Output Shape	Param #
lstm (LSTM)	(None, 30, 64)	442112
lstm_1 (LSTM)	(None, 30, 128)	98816
lstm_2 (LSTM)	(None, 64)	49408
dense (Dense)	(None, 64)	4160
dense_1 (Dense)	(None, 32)	2080
dense_2 (Dense)	(None, 36)	1188

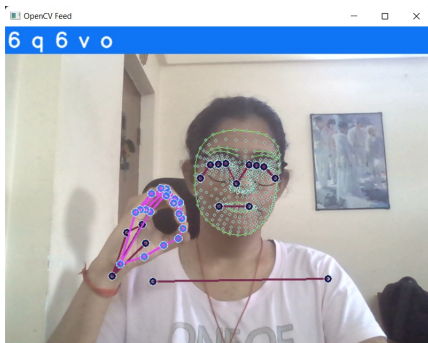
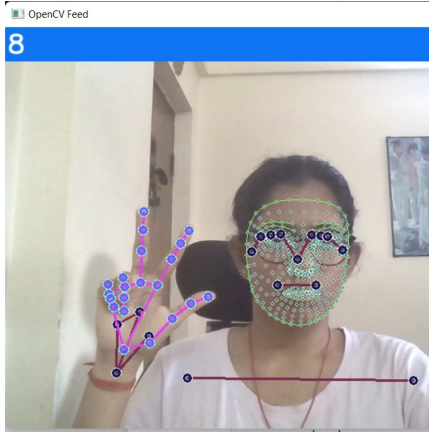
=====
Total params: 597,764
Trainable params: 597,764
Non-trainable params: 0

```
accuracy_score(ytrue, yhat)
```

0.9928774928774928

Data-set Testing

The testing of the model is done once we have the saved trained model. We have used the MediaPipe Holistic to extract key points of the landmarks to pass on to the model. Here we have tested for some of the signs like 6, o, q, v, 8.



Learnings and Challenges

1. Learnings

- The various libraries

- LSTM

2. Challenges

- Online data set integration

- Training

Future Scope

1. Improve accuracy and expand dataset
2. Front-end/UI for the application
3. Conversion into a browser extension