

SUNIYE: An Indian Sign Language Detection App

¹ Pavan Kumar Shetty, ²Anushka Vyas, ³Ayush Shetty, ⁴Kanchan Dabre

^{1,2,3} Department of Computer Engineering, Universal College Of Engineering, Mumbai, India,

⁴ Asst.Professor, Dept. Of Computer Engineering, Universal College of Engineering, Mumbai, India.

Abstract : The paper is all about the system and interface developed, that allows deaf mutes and other people to communicate with each other help of Sign Language. Since the possibility of learning Sign Language for each and every person this project will help them to communicate with ease . It uses the of concepts like Deep Learning, Convolutional Neural Network, Tensor Flow. A webcam first captures the hand gestures, then this system will convert it to text format.. For this purpose, Neural Networks are linked with Tensor flow library. The final text module will be displayed on the screen.

IndexTerms - Sign Languages, Deep Learning, Tensor Flow, Convolutional Neural Network Hand Gestures.

I. INTRODUCTION

A sign language is a way of communication by using the hands and other parts of the body. It should not be confused with body language. Sign languages are an important way for deaf people to communicate. Deaf people often use them instead of spoken languages.

Indian sign language (ISL) is sign language used in India. The most important part of Indian Sign Language (ISL) is it does not include grammar. Sign Language used in India is different than American Sign Language. A government website www.indiansignlanguage.org has been launched for empowering the deaf, which presents a huge database of Indian Sign Language (ISL) signs. Since learning Sign Language is not possible for every person we build this project in order to bridge the gap between such impaired people.

The agenda of the project is to develop an interface that will help the Deaf-mutes as well as other people. As of now, it might seem irrelevant to design such a system but in a longer run it might help deaf-mutes to equally enjoy their social and personal life. Designing such an interface will make them find their freedom and might boost their confidence in this Digital Age. The proposed system makes use of following technologies: TensorFlow which is the most important library used for designing and developing the model of these system, Convolutional Neural Network, is an Deep Learning Algorithm that have been used for serving the purpose of Image Recognition, that helps in images in form of matrix that can be understood by the model and making it ready, and lastly OpenCV that will act as an Eye of the system that will capture and process Real-time Hand Gestures and predict results with help of Classifier.

II. METHADODOLOGY

The basic explanation for this project is to performed a hand gesture in front of the webcam. This sign gesture is converted to text and then display it on the display screen.

Now to understand the technical workflow of the system -The very first step is to provide training dataset and train the system with a variety of hand gestures named with their respective labels. This is the most time consuming step. Better the underlying system software specifications, lesser the time required for training will be. Once training is complete, the next step is detection mode. It now uses the input image from a webcam and runs it through the classifier to find its closest gesture based on the training examples and labels provided in the previous step. If a certain detection threshold is crossed, it will append the label on frame as suggesting that system recognized the following hand gesture accordingly. The entire process can be repeated multiple number of times. But the hand gestures will be only recognized if they meet the trained data set standards.

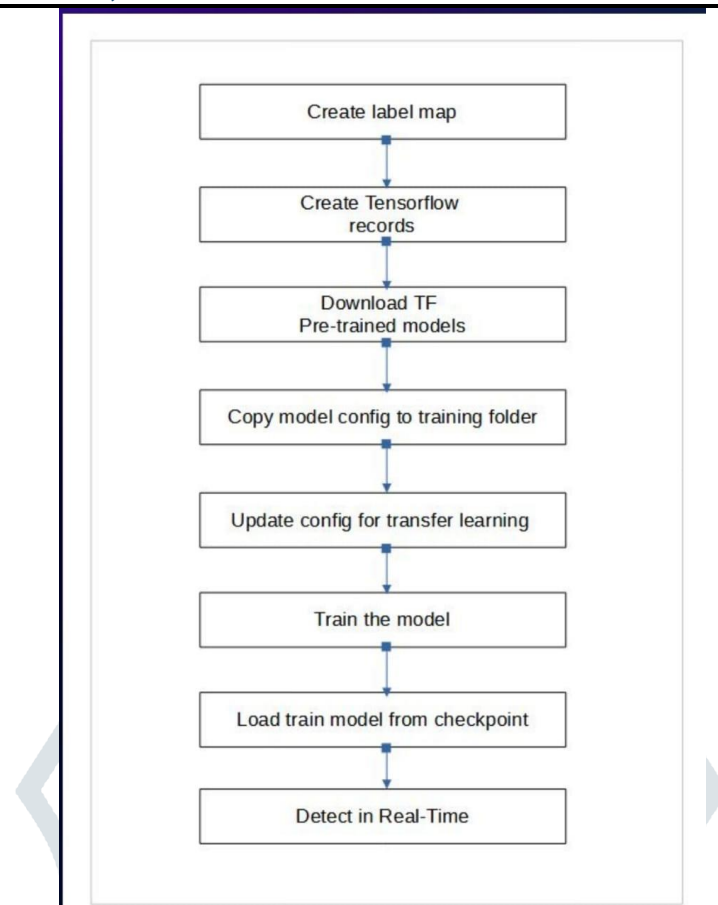


Fig.1: Flowchart of System

2.1 Training Dataset

The idea of using training data in machine learning programs is a simple concept, but it is also very foundational to the way that these technologies work. The training data is an initial set of data used to help a program understand how to apply technologies like neural networks to learn and produce sophisticated results. It may be complemented by subsequent sets of data called validation and testing sets. Since our system is an interface for Real-Time Detection of Hand Gestures our Dataset will purely consist of large number of Images in form of .jpeg, .jpg, these are the only two extensions that can be accepted by our system. The designed model makes use of a Labeled dataset method for training our system, thus assigning labels to folder names will simply use sub-files of images to be trained under assigned labels. while training the models we train our models for 20k step in cmd prompt within the folder of our project so that we get a higher accuracy rate of detection, higher the number of steps higher the accuracy. Once the model is completely trained for a set of particular labelled images it gets Classifier ready and can be used for testing the system's prediction rate. if the model will be trained more number of times for the same set of labels, higher is the success rate.

```

I1105 13:10:02.918599 9044 model_lib_v2.py:652] Step 18600 per-step time 0.127s loss=0.139
INFO:tensorflow:Step 18600 per-step time 0.101s loss=0.113
I1105 13:10:14.583548 9044 model_lib_v2.py:652] Step 18600 per-step time 0.101s loss=0.113
INFO:tensorflow:Step 18700 per-step time 0.125s loss=0.115
I1105 13:10:26.093042 9044 model_lib_v2.py:652] Step 18700 per-step time 0.125s loss=0.115
INFO:tensorflow:Step 18800 per-step time 0.122s loss=0.093
I1105 13:10:37.549224 9044 model_lib_v2.py:652] Step 18800 per-step time 0.122s loss=0.093
INFO:tensorflow:Step 18900 per-step time 0.137s loss=0.103
I1105 13:10:49.080930 9044 model_lib_v2.py:652] Step 18900 per-step time 0.137s loss=0.103
INFO:tensorflow:Step 19000 per-step time 0.103s loss=0.112
I1105 13:11:00.743386 9044 model_lib_v2.py:652] Step 19000 per-step time 0.103s loss=0.112
INFO:tensorflow:Step 19100 per-step time 0.102s loss=0.139
I1105 13:11:12.960543 9044 model_lib_v2.py:652] Step 19100 per-step time 0.102s loss=0.139
INFO:tensorflow:Step 19200 per-step time 0.131s loss=0.221
I1105 13:11:24.555228 9044 model_lib_v2.py:652] Step 19200 per-step time 0.131s loss=0.221
INFO:tensorflow:Step 19300 per-step time 0.132s loss=0.104
I1105 13:11:36.231521 9044 model_lib_v2.py:652] Step 19300 per-step time 0.132s loss=0.104
INFO:tensorflow:Step 19400 per-step time 0.102s loss=0.128
I1105 13:11:47.656452 9044 model_lib_v2.py:652] Step 19400 per-step time 0.102s loss=0.128
INFO:tensorflow:Step 19500 per-step time 0.097s loss=0.117
I1105 13:11:59.260923 9044 model_lib_v2.py:652] Step 19500 per-step time 0.097s loss=0.117
INFO:tensorflow:Step 19600 per-step time 0.124s loss=0.117
I1105 13:12:10.829460 9044 model_lib_v2.py:652] Step 19600 per-step time 0.124s loss=0.117
INFO:tensorflow:Step 19700 per-step time 0.102s loss=0.119
I1105 13:12:22.488073 9044 model_lib_v2.py:652] Step 19700 per-step time 0.102s loss=0.119
INFO:tensorflow:Step 19800 per-step time 0.162s loss=0.118
I1105 13:12:34.071034 9044 model_lib_v2.py:652] Step 19800 per-step time 0.162s loss=0.118
INFO:tensorflow:Step 19900 per-step time 0.120s loss=0.113
I1105 13:12:45.550648 9044 model_lib_v2.py:652] Step 19900 per-step time 0.120s loss=0.113
INFO:tensorflow:Step 20000 per-step time 0.119s loss=0.099
I1105 13:12:57.122153 9044 model_lib_v2.py:652] Step 20000 per-step time 0.119s loss=0.099
  
```

Fig.2: Testing Accuracy

2.2 Tensor Flow

The best part of using TensorFlow library is that it is an open Source Library with lots of pre designed models , useful in Machine Learning and especially Deep Learning. For understanding the conceptual use of Tensor Flow is required to understand the meaning of two terms, where the Tensor here is considered as N-Dimensional Array and Flow refers to graph of operations. Every mathematical computation in TensorFlow is considered as graph of operations where Nodes in the Graph are operations and Edges are nothing but tensors. Any mathematical computation is written in form of data flow diagram in Python Frontend or C++ or Java, as in our case Python is used. Then, TensorFlow Execution Engine comes into picture and makes it deployable on any of the hardware of Embedded System let it be CPU or Android or IOS. TensorFlow is a Machine learning framework that comprises of uses the dataset to train Deep learning models and helps in prediction and also improvise future results. The biggest advantage of using TensorFlow is it's feature of providing Abstraction, that is the developer does not need to work on every small aspects of designing the model as it is managed by the library itself, thus giving the developer the freedom to focus on logic building, which was clearly explained in . TensorFlow in our system helps us in training the model using the provided dataset. TensorFlow object recognition algorithms helps us classify and identify different hand gestures when combined with use of OpenCV. By analysing thousands of photos, Tensorflow can help classifying and identifying real-time hand gestures. It makes possible to develop a model which can help identify 3D images and classify it on basis of 2D images from its feed dataset. TensorFlow is capable of processing more information and spot more patterns.

2.3 Deep Learning

Deep learning is a branch of machine learning which is completely based on artificial neural networks, as neural network is going to mimic the human brain so deep learning is also a kind of mimic of human brain. In deep learning, we don't need to explicitly program everything. Deep Learning makes use of Neural Network most of the times to implement its functioning. A Neural Network is a collection of layers that transforms the input in some way to produce output. Image can be termed as matrix of pixel values so it may seem that classification can be an easier task simply based on matrix classification but that is not the case with complex matrix images or images with similar forms of matrix or a very huge dataset of images with minimal changes in the matrix. This may lead to clash in prediction scores and thereby affecting the accuracy and speed of classifier model. This is where Neural Network comes into picture and thus it is required to use deep learning over machine learning. Machine Learning works with lesser number of layers when compared with Deep Learning as observed from and thus not preferred for technologies like Image Recognition which requires need of Convolutional Neural Networks.

2.4 OpenCV

OpenCV is an open source library for Computer Vision. Now since all the training and classification is ready to be executed when it needed an eye for the designed system to capture real-time images of Hand Gestures which can then be sent for detection and identification. OpenCV adds intelligence to Deep Learning models for visualization image processing. Here images are considered over 2 channels as: RGB Channel and Grey Scale Channel so once the image is captured by OpenCV it first converts into Grey channel so it can then undergo morphological processing as shown. OpenCV makes use of Numpy Library for numerical computation of Images in form of matrix of pixels. A blue box of particular dimension has been designed with help of OpenCV in a way that it will consider hand gestures present inside this blue box. It then converts the image over different channels and then convert the image into convoluted form of matrix so the Classifier model can compare it with previously learned labelled images. It will then predict a suggestion of gesture on basis of the score generated. As OpenCV is converting real-time hand gesture it will be continuously suggesting predictions because of slightest of motion of real-time hand gesture.

III. RESULTS AND DISCUSSION

The short Demo of our test system is given below. We Performed five gestures and it also indicating the accuracy of detection by the model.

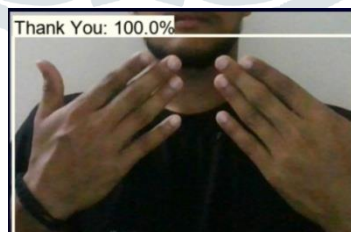


Fig. 3: Gesture for Thank You and accuracy



Fig. 4: Gesture for Hello and accuracy



Fig. 5: Gesture for Yes and accuracy



Fig.6 : Gesture for No and accuracy



Fig -7: Gesture for I Love You and accuracy

IV. CONCLUSIONS

The proposed system was successfully able to capture Hand Gestures using the integrated Web Camera and process and convert into text format. the system was able to provide the accurate and best of its results. However sometimes in poor light conditions and in absence of proper background the system struggled to produce correct and expected results.

The system can be considered as a boon to people with hearing disabilities or speaking disabilities or other people as well. These system would not only bring technology into their Personal lives but also give rise of opportunities in their professional life.

REFERENCES

- [1] Dipali Rojasara and Nehal G Chitaliya, 2013. Indian Sign Language Recognition –A Survey. International Journal of Engineering Research & Technology (IJERT)
- [2] Guru99.com. 2021. Tensorflow Image Classification: Available at:<https://www.guru99.com/convnet-tensorflow-image-classification.html>.
- [3] Deep Learning With Tensorflow: Part 1 — Theory And Setup. Available at: <https://towardsdatascience.com/deep-learning-with-tensorflow-part-1-b19ce7803428>
- [4] Pankajakshan, P. and Thilagavathi B, 2015. Sign language recognition system. 2015 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS).
- [5] Ertham, F. and Aydin, G., 2017. Data Classification with Deep Learning using Tensorflow. IEEE 2nd International Conference on Computer Science and Engineering, pp.757-7