Chapter 1: INTRODUCTION

1.1 Overview

Communication can be defined as the act of transferring information from one place, person, or group to another. It consists of three components: the speaker, the message that is being communicated, and the listener. It can be considered successful only when whatever message the speaker is trying to convey is received and understood by the listener. It can be divided into different categories as follows: formal and informal communication, oral (face-to-face and distance) and written communication, non-verbal, grapevine, feedback, and visual communication, and the active listening. The formal communication (official communication) is steered through the channels that are pre-determined. The unofficial or grapevine communication is the spontaneous communication between individuals in one's profession that does not have any formal protocol or structure. The oral communication (face-to-face and distance) is the communication in which words are exchanged between people who are present in front or at a distance (with the help of technology including voice and video calls, webinars, etc.). The written communication is the communication in which letters, emails, notices, or any other written form is used for communicating. But a problem still exists, not many people possess the knowledge of sign language. Deaf and dumb may be able to communicate amongst themselves using sign languages but it is still difficult for them to communicate with people having normal hearing and vice-versa due to the lack of knowledge of sign languages. This issue can be resolved by the use of a technology driven solution. By using such a solution, one can easily translate the gestures of sign language into the commonly spoken language, English.

1.2 Objectives

The Objectives of our project is:

Our mission is to break communication barriers between deaf and hearing people. In the world there are approximately 466 million people with hearing loss and 2 millions of American sign language speakers. We are developing a technology capable of recognizing and translating sign language in real time.

Chapter 2: LITERATURE SURVEY

The purpose of the Literature Survey is to give the brief overview and also to establish complete information about the reference papers. The goal of Literature Survey is to completely specify the technical details related to the main project in a concise and unambiguous manner. In different approaches have been used by different researchers for recognition of various hand gestures which were implemented in different fields. The whole approaches could be divided into three broad categories

1) Hand segmentation approaches

2) Feature extraction approaches and

3) Gesture recognition approach

Communicate with someone who is hard of hearing: a) Speak clearly, not loudly b) Talk at a reasonable speed. c) Communicate face to face. d) Create a quiet space e) Seek better options. f) Make it easy to lip-read. g) Choose a mask that allows for lip-reading. These solutions are for persons who have a little hearing impairment; nevertheless, if a person is completely deaf, he or she will be unable to understand anything. At this time, Sign Language is their best and only option. Deaf-dumb people rely on sign language as their primary and only means of communication. Because sign language is a formal language that uses a system of hand gestures to communicate (by the deaf), it is the sole means of communication for those who are unable to speak or hear. Physically challenged persons can convey their thoughts and emotions via sign language.

In this paper, a unique sign language identification technique for detecting alphabets and motions in sign language is suggested. Deaf individuals employ a style of communication based on visual gestures and signs. The visual-manual modality is used to transmit meaning in sign languages. It is mostly utilized by Deaf or hard of hearing individuals. Sign language is used by youngsters who are neither deaf or hard of hearing. Hearing nonverbal children who are nonverbal owing to problems such as Down syndrome, autism, cerebral palsy, trauma, brain diseases, or speech impairments make up another big group of sign language users.

The American Sign Language alphabet is used for fingerspelling. There is a symbol for each letter of the alphabet. On your palm, we may use these letter signs to spell out words – most commonly names and locations.

Chapter 3: TOOLS USED

Tools used for this project are discussed in this chapter.

1) Interface: Jupiter Notebook for inserting python libraries in a notebook format, it is typically a python code where we can easily estimate our data sets model in one single notebook.

2) Operating System Environment: Windows 10

3) Software: Python (3.10.0), IDE (Jupiter), NumPy (version 1.16.5), cv2 (open CV) (version 3.4.2) TensorFlow (version 2.9.0), GitHub , Virtual Studio (2022) ,CUDA(10.1) and Cu DNN(7.6) (For NIVIA GPU for faster training model), Pycocotools .

4) Hardware Environment: RAM- 16GB, GRAPHIC CARD – 6GB, ROM-1060TB V.

5) TensorFlow: It is an open-source artificial intelligence package that builds models using data flow graphs. It enables developers to build large-scale neural networks with several layers. TensorFlow is mostly used for classification, perception, comprehension, discovery, prediction, and creation.

3) Object Detection API: It is an open source TensorFlow API to locate objects in an image and identify it.

4) OpenCV: OpenCV is an open-source, highly optimized Python library targeted at tackling computer vision issues. It is primarily focused on real-time applications that provide computational efficiency for managing massive volumes of data. It processes photos and movies to recognize items, people, and even human handwriting.

5) Label Image: Label Image is a graphical image annotation tool that labels the bounding boxes of objects in picture.

Chapter 4: METHODOLOGY

4.1. Proposed Model

The proposed system is designed to develop a real-time sign language detector using a TensorFlow object detection API and train it through transfer learning for the created dataset. For data acquisition, images are captured by a webcam using Python and OpenCV following the procedure described under Section 3.

Following the data acquisition, a labeled map is created which is a representation of all the objects within the model, i.e., it contains the label of each sign (alphabet) along with their id. The label map contains 26 labels, each one representing an alphabet. Each label has been assigned a unique id ranging from 1 to 26. This will be used as a reference to look up the class name. TF records of the training data and the testing data are then created using generate TF record which is used to train the TensorFlow object detection API. TF record is the binary storage format of TensorFlow. Binary files usage for storage of the data significantly impacts the performance of the import pipeline consequently, the training time of the model. It takes less space on a disk, copies fast, and can efficiently be read from the disk.

A. **Transfer Learning*:*** Transfer learning is a machine learning technique where a model trained on one task is re-purposed on a second related task. Transfer learning is an optimization that allows rapid progress or improved performance when modeling the second task. Transfer learning is related to problems such as multi-task learning and concept drift and is not exclusively an area of study for deep learning. Transfer learning is simply the process of using a pre-trained model that has been trained on a dataset for training and predicting on a new given dataset.

Figure.1Transfer learning block diagram

B. **SSD Mobile net V2**: The Mobile Net SSD model is a single-shot multi box detection (SSD) network that scans the pixels of an image that are inside the bounding box residual models, the model's architecture is built on the notion of inverted residual.

structure, in which the residual block's input and output are narrow bottleneck layers. In addition, nonlinearities in intermediate layers are reduced, and lightweight depth wise convolution is applied. The TensorFlow object detection API includes this model

C. **Pretrained Model Name:**

SSD_MOBILENET_V2_FPNLITE_320x320_COCO17_TPU-8 SPEED (m s) - 22 COCO MAP (mean average precision) - 22.5 OUTPUT – Boxes Accuracy is inversely proportional to speed Precision- True Precision/(True Precision + False Precision) High precision value means more accurate will be the Model

4.2. Algorithm used:

- Convolutional Neural Network:

A Convolutional Neural Network (CNN) is a Deep Learning system that can take an input picture and assign importance (learnable weights and biases) to various aspects/objects in the image, as well as differentiate between them. The amount of pre- processing required by a Conv Net is much less than that required by other classification techniques. Conv Nets can learn these filters/characteristics with adequate training, whereas simple techniques need hand-engineering of filters. Conv Nets are multilayer artificial neural networks designed to handle 2D or 3D data as input. Every layer in the network is made up of several planes that may be 2D or 3D, and each plane is made up of numerous independent neurons composition, where nearby layer neurons are linked but same layer neurons are not. A Conv Net can capture the Spatial and Temporal aspects of an image by applying appropriate filters. Furthermore, reducing the number of parameters involved and reusing weights resulted in the architecture performing better fitting to the picture collection. Conv Net's major goal is to make image processing easier by extracting relevant characteristics from images while preserving crucial information that is must for making accurate predictions. This is highly useful for developing an architecture that is not just capable of collecting and learning characteristics but also capable of handling massive volumes of data

Chapter 5: MODEL ANALYSIS

Figure5.1 Different images

Step 1: Collecting and labelling images

In Collecting the images define the signs in images collection and setup the folder then after run python file to capture for label image.

Label image is a graphical image annotation tool.

It is written in Python and uses Qt for its graphical interface.

Annotations are saved as XML files in PASCAL VOC format, the format used by ImageNet.

For further steps there should be downloaded and installed object detection API from TensorFlow model zoo.

Figure5.2 Labelling the image

Figure5.3 Labelled image

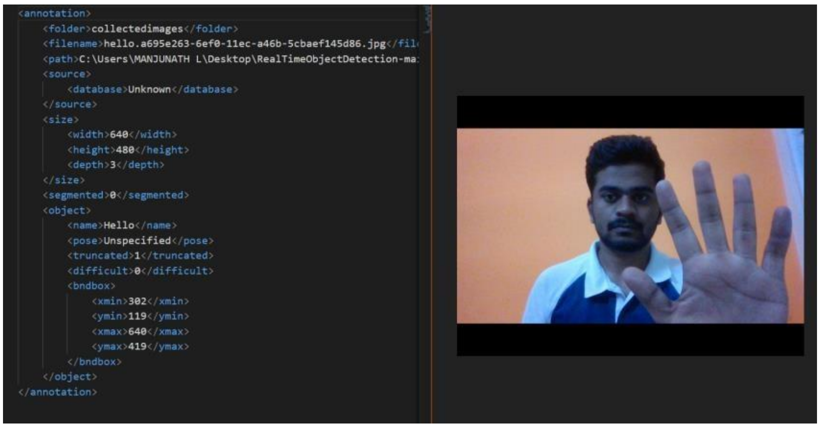

Figure 5.4 Labelled image and it’s xml code

STEP 2: Creating the TF records

TF Record are binary file format for storing data. Using TF Records helps speed up the training for your custom object detection model.

This is an end-to-end example of how to read and write image data using TF Records. Using an image as input data, you will write the data as a TF Record file, then read the file back and display the image.

This can be useful if, for example, you want to use several models on the same input dataset. Instead of storing the image data raw, it can be preprocessed into the TF Records format, and that can be used in all further processing and modelling.

STEP 3: Train the Model

The open-source framework, TensorFlow object detection API makes it easy to develop, train and deploy an object detection model. They have their framework called the TensorFlow detection model zoo which offers various models for detection that have been pre-trained on the COCO 2017 dataset. The pre-trained TensorFlow model that is being used is SSD Mobile Net v2 320x320. The SSD Mobile Net v2 Object detection model is combined with the FPN-lite feature extractor, shared box predictor, and focal loss with training images scaled to 320x320. Pipeline configuration, i.e., the configuration of the pre-trained model is set up and then updated for transfer learning to train it by the created dataset figure [5.1]. For configuration, dependencies like TensorFlow, contiguity, pipeline_pb2, and text format have been imported. The major update that has been done is to change the number of classes which is initially 90 to 26, the number of signs (alphabets) that the model will be trained on. After setting up and updating the configuration, the model was trained in 10000 steps. The hyper-parameter used during the training was to set up the number of steps in which the model will be trained which was set up to 10000 steps. During the training, the model has some losses as classification loss, regularization loss, and localization loss. The localization loss is mismatched between the predicted bounding box correction and the true values.

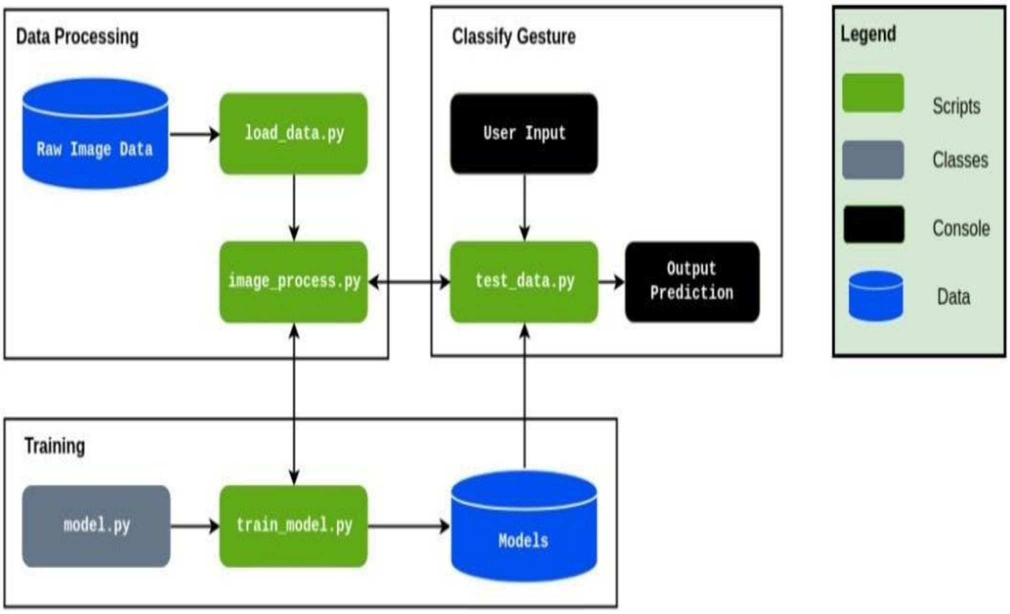

Figure 5.5 Block diagram of overflow of training model

STEP 4: Evaluate the Model

Once we trained a model, ideally we want to take a look at how well its performing. We normally take a look at

Precision: TP/(TP+FP)

What proportion of my detections were correct?

Recall: TP/(TP+FN)

What proportion of the actual objects did I capture?

Loss: A method of evaluating how well the model is performing against the data provided

FP->False Positive TN->True Negative TP->True Positive FN->False Negative

Below shown Figure[5.6-5.9] are the cases for True Positive.

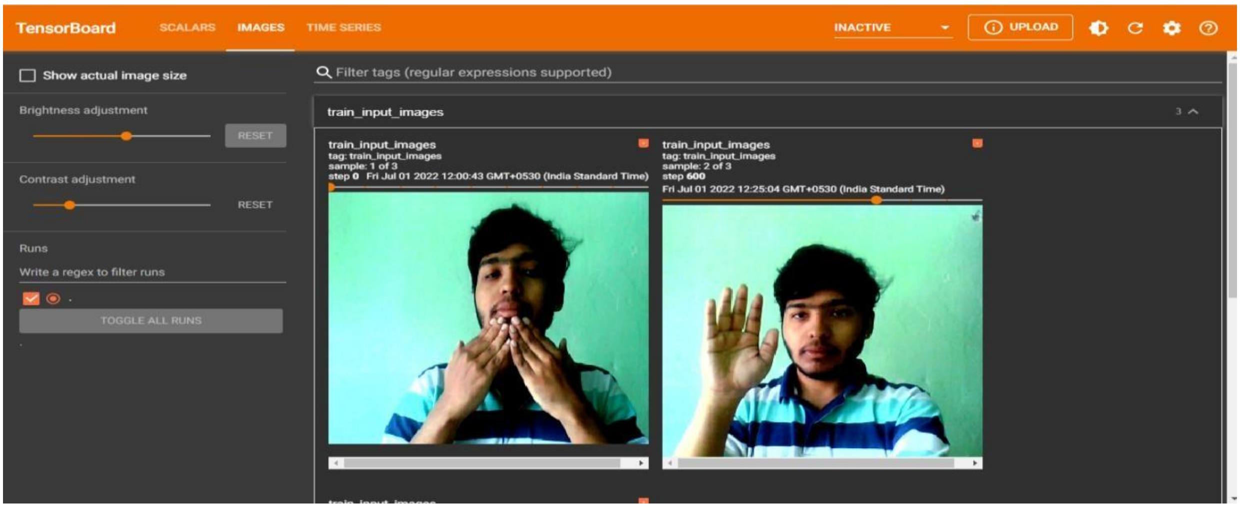

Fig 5.6 Train input images

Fig 5.7 Train input images

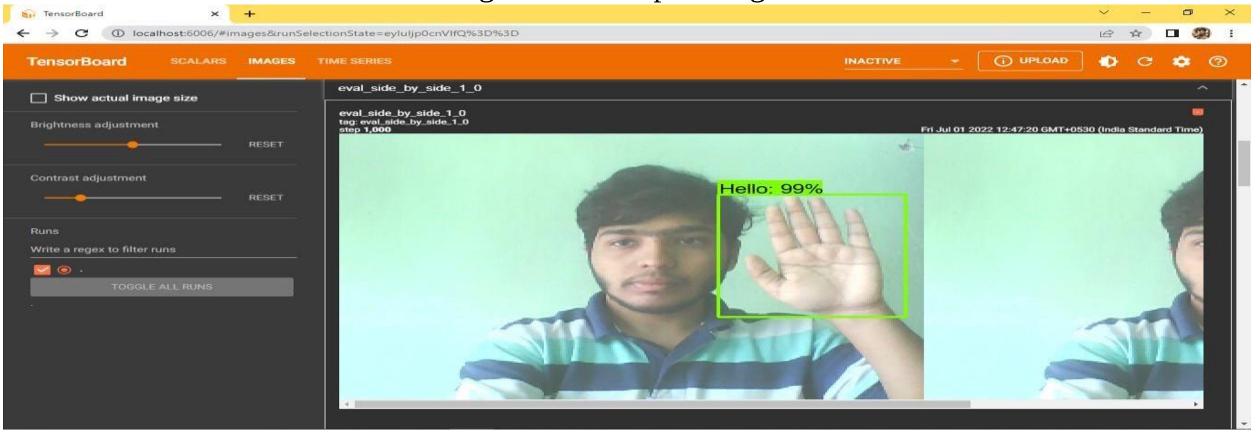

Fig 5.8 Evaluated Hello Image

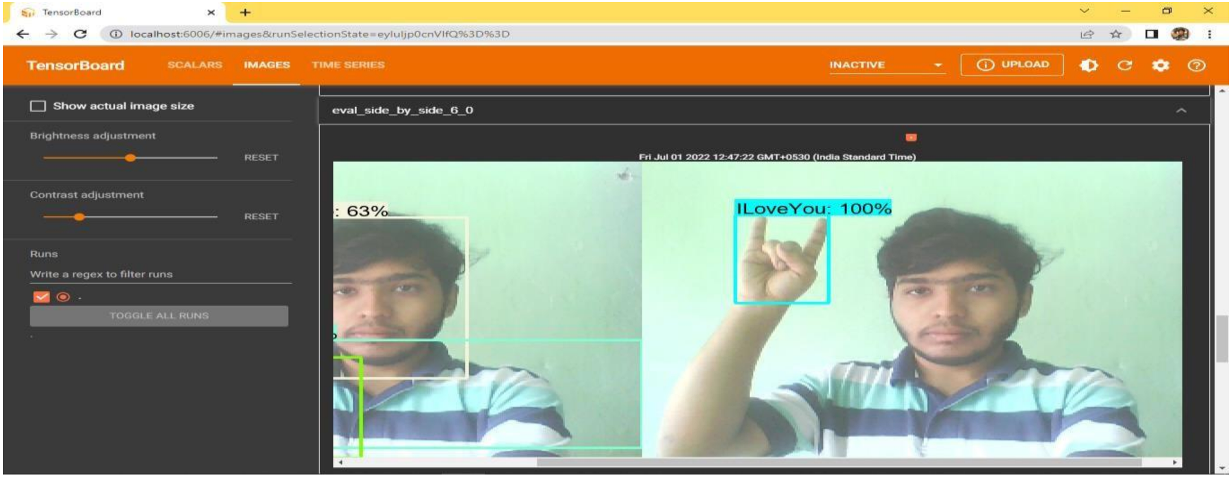

Fig 5.9 Evaluated I Love You Image

STEP 5: DETECTION

Detecting using Images

Sometimes you want to test our object detection model on a single image. This is particularly useful if you're training on a machine where a webcam isn't available.

Detection Using Webcam

Real time detection is where it's at, being able to detect objects in real time opens up a number of applications for the technology Fig 5.10.

```
detections['detection_classes'] = detections['detection_classes'].astype(np.int64)

label_id_offset = 1
image_np_with_detections = image_np.copy()

viz_utils.visualize_boxes_and_labels_on_image_array(
            image_np_with_detections,
            detections['detection_boxes'],
            detections['detection_classes']+label_id_offset,
            detections['detection_scores'],
            category_index,
            use_normalized_coordinates=True,
            max_boxes_to_draw=5,
            min_score_thresh=.8,
            agnostic_mode=False)

plt.imshow(cv2.cvtColor(image_np_with_detections, cv2.COLOR_BGR2RGB))
plt.show()
```

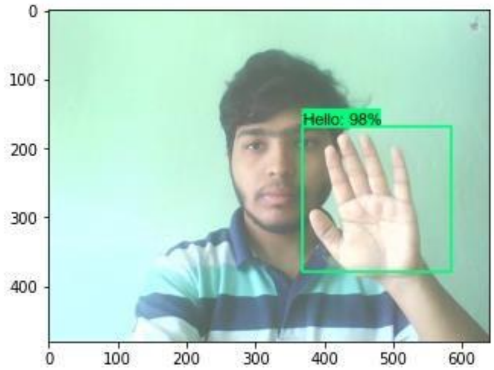

Fig 5.10 Hello sign detected using image

Chapter 6: RESULT AND ANALYSIS

Images used to train	True Result	False Result	Accuracy (%)
50	24	28	48
100	53	47	53
200	145	55	72.5
150	150	60	75.8
300	232	68	86.5

Table 6.3: Analysis of result

As a result, by expanding the dataset, the model may simply be scaled up to a vast size. Environmental issues such as low light intensity and an unmanaged backdrop are some of the model's limitations cause decrease in the accuracy of the detection. Therefore, we'll work next to overcome these flaws and also increase the dataset for more accurate results figure[6.1].

This chapter discusses about the results obtained.

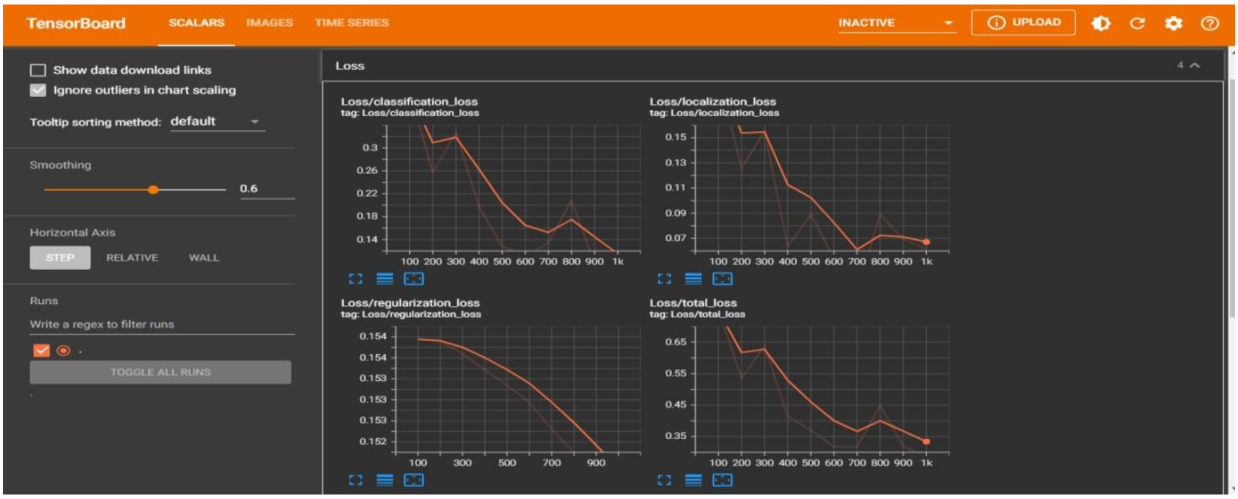

Fig 6.1 Loss curve

The developed system is able to detect Indian Sign Language alphabets in real-time. The system has been created using TensorFlow object detection API. The pre-trained model that has been taken from the TensorFlow model zoo is SSD Mobile Net v2 320x320.

It has been trained using transfer learning on the created dataset which contains 650 images in total, 25 images for each alphabet.

The developed system is able to detect Indian Sign Language alphabets in real-time. The system has been created using TensorFlow object detection API. The pre-trained model that has been taken from the TensorFlow model zoo is SSD Mobile Net v2 320x320.

It has been trained using transfer learning on the created dataset which contains 125 images in total, 25 images for each sign.

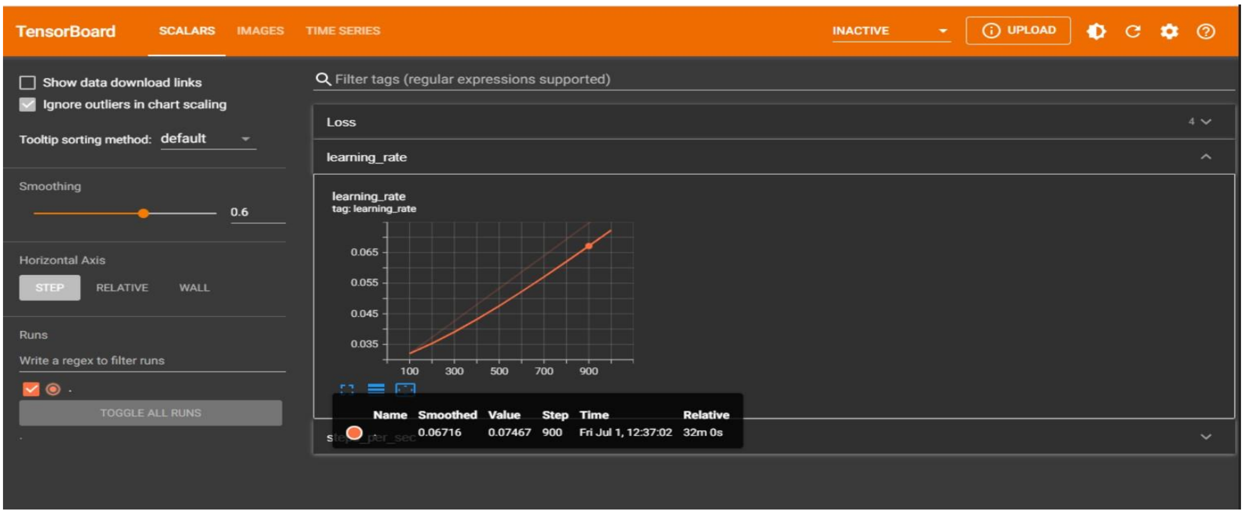

Fig 6.2 Learning curve

Chapter 7: CONCLUSION

7.1 Application

1) The dataset can easily be extended and customized according to the need of the user and can prove to be an important step towards reducing the gap of communication for dumb and deaf people.

2) Using the sign detection model, meetings held at a global level can become easy for the disabled people to understand and the value of their hard work can be given.

3) The model can be used by any person with a basic knowledge of tech and thus available for everyone.

4) This model can be implemented at elementary school level so that kids at a very young age can get to know about the sign language.

7.2 Future Scope

1) The implementation of our model for other sign languages such as Indian sign language or American sign language.

2) Further training with large dataset to efficiently recognize symbols.

3) Improving the model's ability to identify expression

7.3 Inference

The fundamental goal of a sign language detecting system is to provide a practical mechanism for normal and deaf individuals to communicate through hand gestures. The proposed system will be used with a webcam or any other in-built camera that detects and processes indicators for recognition. We may deduce from the model's findings that the suggested system can produce reliable results under conditions of regulated light and intensity. Furthermore, new motions may be simply incorporated, and more images captured from various angles and frames will supply the model with greater accuracy. As a result, by expanding the dataset, the model may simply be scaled up to a vast size. The model has some limitations, such as environmental conditions such as low light intensity and an unmanaged backdrop, which reduce detection accuracy. As a consequence, we'll attempt to fix these problems as well as expand the dataset for more accurate findings

From the result of the model, we can conclude that the proposed system can give accurate results under controlled light and intensity. Furthermore, custom gestures can easily be added and more the images taken at different angle and frame will provide more accuracy to the model. Thus, the model can easily be extended on a large scale by increasing the dataset. The model has some limitation such as environmental factors like low light intensity and uncontrolled background which cause decrease in the accuracy of the detection. Therefore, we’ll work next to overcome these flaws and also increase the dataset for more accurate results.

8.REFERENCES

- Jamie Berke , James Lacy March 01, 2021 "Hearing loss/deafness| Sign Language" https://www.verywellhealth.com/sign-language-nonverbal-users-1046848
- SAKSHI GOYAL, ISHITA SHARMA, S. S. Sign language recognition system for deaf and dumb people. International Journal of Engineering Research Technology 2, 4 (April 2013
- https://youtu.be/pDXdlXlaCco
  20.Bantupalli, K ., Xie, Y .: American Sign Language Recognition using Deep Learning and Computer Vision. Proc. - 2018 IEEE Int. Conf. Big Data, Big Data 2018. 4896– 4899
  (2019). https://doi.org/10.1109/BIGDATA.2018.8622141.

Appendix 1

BMS Institute of Technology and Management

AVALAHALLI, DODDABALLAPUR MAIN ROAD, BANGALORE-64

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

STUDENTS PROFILE

S.No	Name	USN	Photograph	Mail Id	Mobile Number
1	Manjunath L	1by19ec097		1by19ec097@bmsit.in	8073090980
2	Gowtham C	1by19ec056		1by19ec056@bmsit.in	8088792799
3	H P Viraj	1by19ec057		1by19ec057@bmsit.in	6366192546
4	Monish Kumar M	1by19ec104		1by19ec104@bmsit.in	8431931367