International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 06 Issue: 05 | May - 2022 Impact Factor: 7.185 ISSN: 2582-3930

Sign Language Recognition Using Tensor Flow Object Detection

Prashanth Venkatesan[1], Ashwin Nair[2], Nachiket Dhopavkar[3], Prof. Savita Lohiya[4]

[1234]*Department of Information Technology, SIES Graduate School of Technology, Nerul.*

Abstract - Sign Language is really crucial for people with physical disabilities like hearing impairment or the ones who are unable to speak due to some chronic issue. They usually communicate through hand movements and gestures, various hand shapes to pass on the information. In this paper, we demonstrate a model comprising of American Sign Language (ASL) as it has shown higher spatial ability among students as well as proved to have higher cognitive benefits. The system makes use of major Python modules like OpenCV and TensorFlow to train and detect various signs which are used universally. Due to machine learning algorithms, deep learning techniques such as CNN (Convolutional Neural Network), training and testing modules, we are now able to revolutionize the industries like medical research and care, schools/universities, self-automated cars like Tesla and more.

Key Words: sign language, american sign language, tensorflow, opencv, python, hearing impairment

1. INTRODUCTION

Sign language has been widely used over many decades for communicating with people who have difficulty while speaking or listening. But the key thing is not many people are so used to communicating with signs and gestures which develops the gap between deaf/dumb people and the average ones in the society. This paper gives more insight about how this gap is nurtured, so that everyone can work and communicate together without much hassle.

Thanks to technological advancements over the years, we can now integrate neural network, artificial intelligence, machine learning with sign language and bring a shine to the orthodox way of communication. In spite of many existing models already developed for the similar cause, there are many flaws in the system especially in segmentation of moving hands which slows down the process. By bringing in Tensorflow object detection, we can correct the flaws in the current system. It is used for tracing and locating an object in a multimedia (photo/video).

According to various studies over the years, people with vocal disabilities are more likely to develop depression due to lack of communication with the close ones, which pushes them into being lonely. Depression and feeling of loneliness are really common among the ones who have hearing impaired issues, especially if they are submerged in the atmosphere full of vocals.

Therefore, we need a system which breaks this communication gap and also has top tier technological advancements which makes the sign language detection model the epitome of flawlessness.

2. LITERATURE REVIEW

Maryam Pahlevanzadeh et al. [1] showed a basic method for recognizing sign language which consisted of 2-layer classifier. First layer for recognizing signs in accordance with the movements of the hand and the second one for recognizing group of hand movements based on the shape of their hand And Karishma Dixit et al. [2] showed an approach which recognizes gestures based on ISL (Indian Sign Language) which then gets converted into a readable text. Priyanka C Pankajakshan et al. [3] presented a system for sign language recognition which uses Artificial Neural networks (ANN). ANN in this case is used for feature extraction. Anup Kumar et al. [4] showed the difference between dynamic and static gestures. It showed an improvised method to pull out gestures from a video format. Ashish S. Nikam et al. [5] demonstrated a project which could detect number and finger points using convex hull algorithm And Tülay Karayılan et al. [6] presented a project based on American Sign Language (ASL) which used Backpropagation Neural Network Algorithm to recognize signs. Arun C S et al. [7] used OpenCV and Jarvis algorithm to recognize alphabets and numbers. Noor Tubaiz et al. [8] used Modified k-Nearest Neighbor (MKNN) for detecting arabic sign language via glove-based technique. Hermann Hienz et al. [9] developed a video-based model for recognizing German Sign Language (GSL) and explained basics of Hidden Markov Models (HMM). Kumud Tripathi et al. [10] proposed an Indian Sign Language (ISL) system using gradient based key frame extraction method and showed the classification results for each sign. Vaishali S. Kulkarni et al. [11] developed a system for detecting static gestures of ASL. They did trial and error among three feature extraction methods and chose the one which showed better results with use of Artificial Neural Network (ANN). Akira Iwata et al. [12] presented a study which made use of histogram to deal with problems caused by posture of signs and also projected comparison between different rotation invariants

3. PROPOSED SYSTEM

The proposed system would bridge the gap between general people and differently abled people by the use of real time object detection system made of python modules (TF, OpenCV, etc) and understand the input given by gestures and then dynamically convert them to a normal recognizable output. This eases the communication as it translates signs in real time and efficiently build communication between the hearing impaired and the general public.

Firstly, a virtual environment is created to install all the dependencies required like wget, opencv, tensorflow etc. Then a label map is created which contains all the basic information about the signs created and a tensorflow record set is also

created for the given signs. Then the model creation and training starts.

The modelling process is divided into three steps:

- Capturing images.
- Labelling images.
- Training the model.

A. Capturing images via OpenCV for to train our model.

- This sub module consists of an automated script which captures 16 images for every sign with an interval of 5 seconds between each image and an interval of 10 seconds between each recurring sign. Figure 1 shows the sign equivalent to "Yes" being captured using the written script. This step consumes a good chunk of resources (processor, RAM) but proved to be faster than traditional method of capturing images as the script once run, led to the whole process being completely auto-pilot.
- Each sign is then saved in their respective folders which will be used further for training and testing purpose.
- 90% of the images captured will be used for training and the rest will be used for testing.

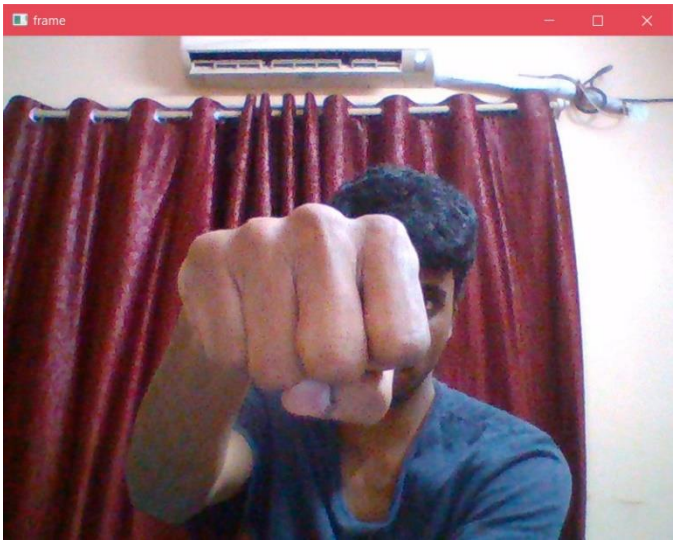

Fig -1: Curating images using OpenCV script.

B. Using a GitHub repository named labelImg to label our signs on the images captured in the earlier step for further evaluation and modelling.

- labelImg is basically an annotation tool written in Python to label any kind of images, a commonly used tool in Tensorflow Object Detection related projects.
- Repository was cloned and was further used to label each and every image individually.
- An xml file is created for all the images which contains information about the image resolution and the labelling which we did in the previous step.
- This is a very crucial step as it will be used for extracting the features of the given sign.

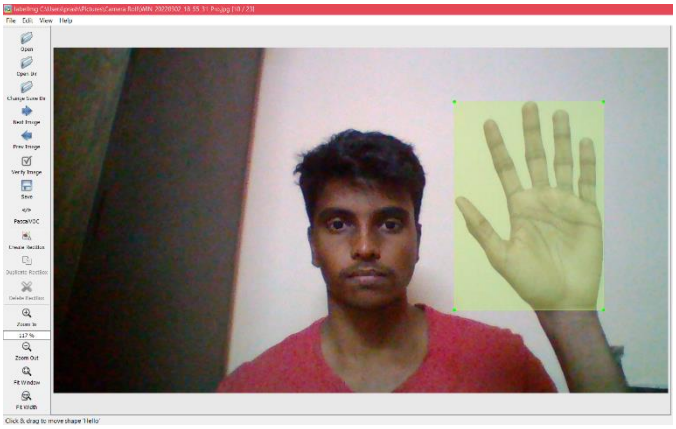

Fig -2: Labelling images for feature extraction.

C. Training the model with a pre-trained model (SSD MobileNet v2 320x320)

- Pre-trained model named SSD MobileNet v2 320*320 was chose as it showed great speed as well as good accuracy and is resource friendly as well.
- The training process was fastened by bringing graphic card into the play for faster processing and rendering speed.
- The overall training process took roughly about 36 minutes after using graphic card instead.

The trained model is able to detect about 15 ASL signs along with a custom sign which represents the developers.

4) WORKING

- Firstly, the system takes the input from a webcam using the help of OpenCV module. The system takes an input from the user via capturing the video of the user showing a certain sign.

Fig -3: Process

- The input is then processed and the parameters are further recognized using the extraction and matching feature, taking use of the model which we trained. After matching features with the model trained, the input sign is detected and the output is displayed to the user and the signs are recognized in real-time. Detecting in real time is a tedious job as it takes a

lot of resources and the user needs to make sure he/she are sitting in a properly lit room preferably one using a ring light and a light plain background in order to improve the sign recognition and accuracy.

- We can further integrate this model into web applications or IOT projects like Raspberry Pi by converting the model into a TFJS or a TFLite file format.

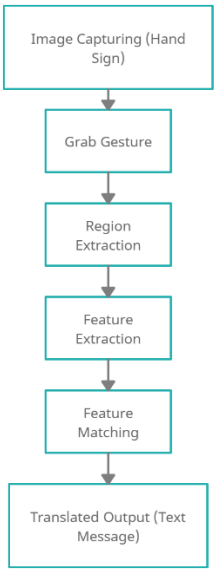

Fig -4: Case diagram for sign language recognition.

5) RESULTS AND DISCUSSIONS

Hearing-impaired sign users rely solely on sign language interpreters for communication. Therefore, you cannot rely on an interpreter every day. Interpretation costs are also very high and may not be available daily to all hearing-impaired people. This system helps to significantly improve the quality of life for hearing-impaired users by providing an auto-pilot system which would detect the ASL signs in real time in a fraction of second. People in general can also use this technology to learn new signs which will further enhance their communication with their loved ones who have a tough time communicating normally.

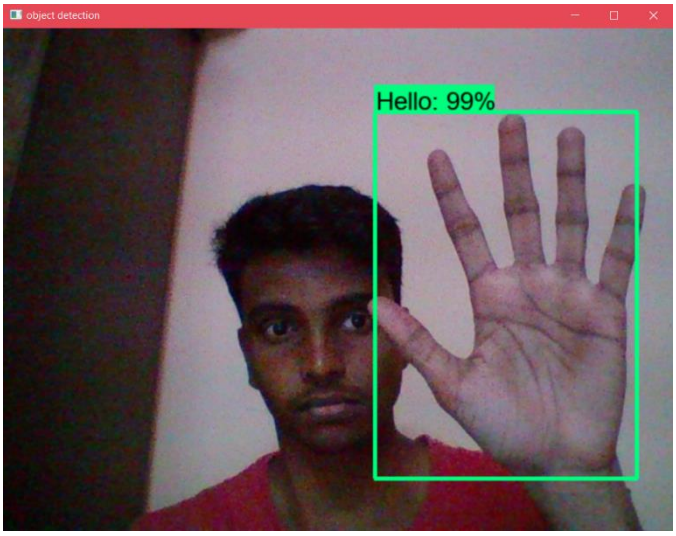

Fig -5: Result 1

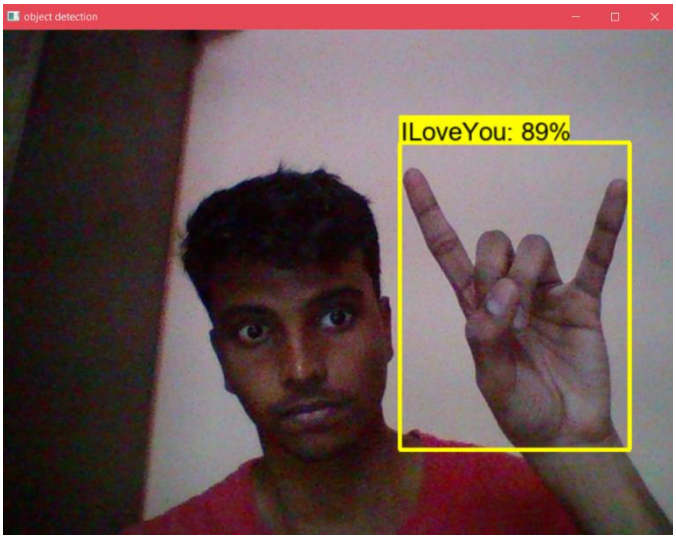

Fig -6: Result 2

We further evaluate our trained model using Tensorboard, to analyse the loss and evaluation metrics to make sure everything is proper and running and improve the model if necessary. Figure 7 shows the loss curve of the trained model.

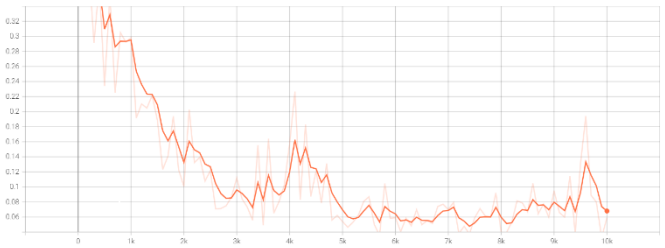

Fig -7: Loss metrics using Tensorboard.

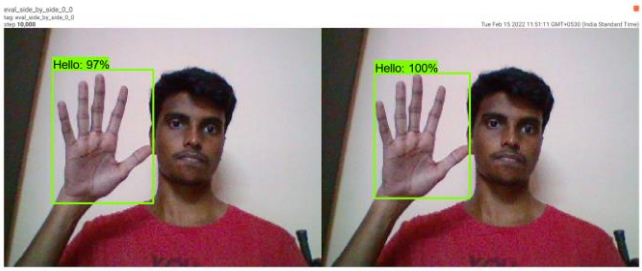

Fig -8: Evaluation using Tensorboard

After much trial and error, we found that capturing images for training in moderate lighting worked the best. If any signs didn't get detected properly, we captured the images of that particular sign in a noisy background which improved the performance. We took about 16 images for every sign and trained for about 10000 steps using pre-trained model SSD MobileNet v2 320x320. In order to rapidly render the output, we made sure to install the proper version of NVIDIA CUDA and cuDNN which will provide GPU accelerated support.

6. CONCLUSIONS

So, at the end, a full-fledged model has been created which is able to detect American Sign Language including some of the ones which is usually used by toddlers for communicating with their parents. After thorough analysis of TensorBoard analytics, the average accuracy for the collected signs comes out to be about 96%, and can be further achieved if the lighting of the room is better and less noisy

ACKNOWLEDGEMENT

The authors of this journal would like to thank all the faculties and friends for giving constant support throughout the journey of this project and the SIES Graduate School of Technology for their contribution towards the development of the project.

REFERENCES

1. Maryam Pahlevanzadeh; Mansour Vafadoost; Majid Shahnazi "Sign language recognition," 2007 9th International Symposium on Signal Processing and Its Applications.
2. Karishma Dixit; Anand Singh Jalal "Automatic Indian Sign Language recognition system, " 2013 3rd IEEE International Advance Computing Conference (IACC).
3. Priyanka C Pankajakshan; Thilagavathi B "Sign language recognition system," 2015 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS)
4. Anup Kumar; Karun Thankachan; Mevin M. Dominic "Sign language recognition," 2016 3rd International Conference on Recent Advances in Information Technology (RAIT).
5. Ashish S. Nikam; Aarti G. Ambekar "Sign language recognition using image based hand gesture recognition techniques," 2016 Online International Conference on Green Engineering and Technologies (IC-GET)
6. Tülay Karayılan; Özkan Kılıç "Sign language recognition," 2017 International Conference on Computer Science and Engineering (UBMK)
7. Arun C S; Geethu G Nath "Real Time Sign Language Interpreter" 2017 IEEE International Conference on Electrical, Instrumentation and Communication Engineering (ICEICE)
8. Khaled Assaleh; Noor Tubaiz; Tamer Shanableh "Glove-Based Continuous Arabic Sign Language Recognition in User-Dependent Mode" 2015 IEEE Transactions on Human-Machine Systems.
9. Hermann Hienz; Britta Bauer "Relevant Features for Video-Based Continuous Sign Language Recognition" 2000 Proceedings Fourth IEEE International Conference on Automatic Face and Gesture Recognition
10. Neha Baranwal; G. C. Nandi; Kumud Tripathi "Continuous Indian Sign Language Gesture Recognition and Sentence Formation" 2015 Eleventh International Multi-Conference on Information Processing (IMCIP)
11. Dr. S.D.Lokhande; Vaishali S. Kulkarni "Appearance Based Recognition of American Sign Language Using Gesture Segmentation" 2010 International Journal on Computer Science and Engineering (IJCSE)
12. Akira Iwata; Susumu Kuroyanagi; Simei G. Wysoski; Marcus V. Lamar "A Rotation Invariant Approach On Static-Gesture Recognition Using Boundary Histograms And Neural Networks" 2002 9th International Conference on