REAL-TIME SIGN DETECTION TO TEXT CONVERSION

HarrishKumar L
Department of CSE
Sathyabama Institute of Science and Technology
Chennai, India
Harrishkumarl2020@gmail.com

Hashwanth Kumar H
Department of CSE
Sathyabama Institute of Science and Technology
Chennai, India
hashwanthprince@gmail.com

Karunya K
Assistant professor,
Department of CSE
Sathyabama Institute of Science and Technology
Chennai, India
karunsai.ifs@gmail.com

Poornima D
Assistant professor, Department of CSE
Sathyabama Institute of Science and Technology
Chennai, India
Poornima.d.cse@sathyabama.ac.in

Gowri Manohari V
Assistant Professor,
Department of CSE
Sathyabama Institute of Science and Technology
Chennai, India
Gowrimanohari.cse@sathyabama.ac.in

Abstract- Sign language is an essential communication aid for the hard of hearing. Rather than being real-time, Machine learning is an essential component of today's Indian Sign Language Recognition systems techniques that consider motions made with both one and two hands. We offer an identification system for speaking with signs in real-time that uses TensorFlow and OpenCV, two cutting-edge technologies, to convert gestures from sign language to text in a quest to narrow the communication gap that exists between the hearing and deaf communities. In this project working, we suggest a technique to use a webcam to generate an Indian Sign Language dataset, which can subsequently be utilized in the TensorFlow model training process during the transferring of learning to build an Real time Sign Language Recognition system. Even with a limited amount of data, this approach can achieve a reasonable degree of accuracy.

Key phrase – Sign-Language Detection, Hand Gestures, Close the Communication Gap, TensorFlow

I. Introduction

One of the most popular uses of computer vision is face detection. It is a basic among the pattern recognition and computer vision. Several techniques for detecting facial features have been introduced in the past ten years. Convolutional neural networks (CNN) and deep learning have demonstrated remarkable progress in recent years, enabling extremely accurate face identification systems. Computer technology known as "face detection" locates and measures a person's face in digital photos. The objective of facial recognition is to identify faces in a picture and to return the bounding box of each face that is identified (see object detection). In the computer image, other items such as bodies, buildings, and trees are omitted. Finding the locations and sizes of every item in an image that belongs to a particular class is the problem of object-class detection, which may be thought of as a special instance of face detection.

II. Proposed system

A real-time sign language identification system that uses TensorFlow and OpenCV, two cutting-edge technologies, to translate sign language motions into text. In this research, we propose a technique to utilize

a camera to construct an Indian sign-language database, after which a TensorFlow model is trained with transfer learning to develop a real-world system for interpreting language understanding. The algorithm can get a respectable degree of accuracy even with a little dataset. A real-time recognition system based on Indian Sign Language is being designed. Webcam feeds are gathered for gathering data utilizing OpenCV and Python. This project combines computer vision and natural language processing to develop a system that is capable of deciphering and understanding real-world sign language scenarios.

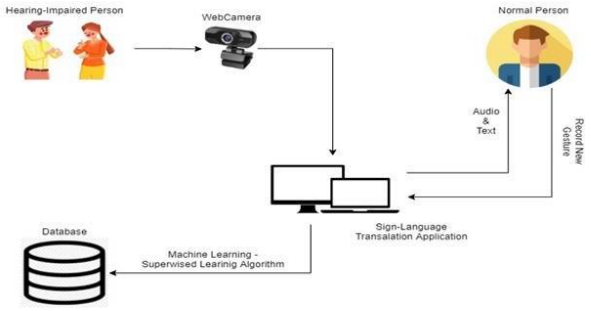

Fig 2.1 System Architecture

A. Data acquisition

There are several methods for obtaining data, but the most popular ones involve the use of sensory devices and vision-based approaches.

1) By the use of the sensory devices

It can offer information on our hand motions with the aid of other IOT devices and electromagnetic sensors. Data gloves will be used to do this. They are not very user-friendly and rather pricey.

2) By vision-based approach

The computer camera is an input device in this strategy. Therefore, additional sensors or data gloves are not needed in order to retrieve the data. As a result, the subject of the camera will become aware that it is an ordinary dialogue with the computer. The primary library utilized in this instance is OpenCV, which stands for Open-source computer vision. The primary difficulty with computer vision is that the system must adjust to the pace of hand motions and ensure that variations in a person's color do not cause confusion in the actions.

B. Data preprocessing

Threshold-based color detection with background removal is required for hand detection using the OpenCV package. OpenCV requires threshold-based color detection with background removal in order to recognize hands. Every color has a distinct value that is used to set it apart from the others. The OpenCV utilizes a filter called Gaussian blur for this. This filter is used to downscale the image's frames in order to extract as much details as possible The Gaussian pyramid has three as its default value. We tried utilizing color segmentation algorithms to manually segment each image, but as the study report points out, lighting has an enormous effect on skin tone and hue. As a result, the segmentation we attempted to do did not yield very good results We also determined that rather than segmenting the hand based on skin color, we would maintain the hand's background as a stable single color. This is because we have plenty of signs to be trained in my project, more of which have similar gestures, such as the "V", "2" symbols. This would enable us to achieve improved outcomes.

C. Gesture classification

A rapid and effective method for identifying static hand motions is used: the Naïve-Bayes classifier. Its foundation is the categorization of various movements based on geometric-based invariants that are extracted from segmented visual data. The movements are captured with a static background from every frame in the video. The objects of interest must first be divided, labeled, and their geometric invariants must be extracted. After that, a distance weighting strategy and the K nearest neighbor method (KNN) are used to classify the gestures in order to provide relevant information for a selectively loaded naive Bayesian classifier..

III. Methodology

A. Data gathering

This module generates a numerical value for each action done by a person using sign language gesture recognition..

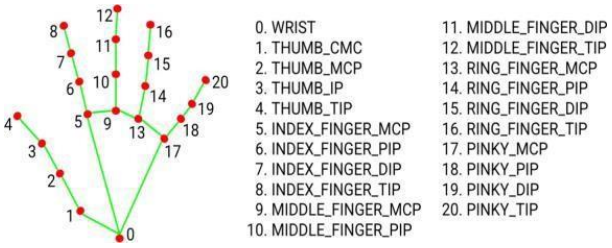

Fig3.1 Module Collecting Hand Gesture

B. Algorithm used

Here, the CNN algorithm is employed. The series of data points is processed by a Convolutional Neural Network algorithm, which then forecasts the sign language motions being made. The CNN system can accurately anticipate the movements being done because a collection of annotated sign language movements served as its training source.

C. Data pre-requisitions

1) Python

Python is popular among programmers due to its versatility, flexibility, and ease of use. Python is the best programming language for machine learning as it is widely used in the programming community and can run on its own platform. The subfield of artificial intelligence called machine learning aims to eliminate the need for precise instructions by enabling computers to learn by making mistakes and repeating tasks. But "machine learning" is the process of teaching computers to detect visual and auditory cues, understand speech, interpret language, and ultimately make important decisions on their own, commonly known as "artificial intelligence" (AI). Accomplishing a process that is difficult to execute without expertise requires further development of expertise because intelligent solutions must be adapted to real situations. Give good answers to the questions the real world needs to evolve in this way. Python is a popular programming language that is considered by many to have the best technology for this type of work. Python is simpler and similar to other programming languages. Additionally, a strong Python community allows employees to easily discuss current projects and offer suggestions for improvements.

2) OpenCV

A collection of projects called OpenCV (Open Source Computer Vision Collection) focuses mainly on real-time computing. Apache 2 is a free and open source cross-platform library. OpenCV has been providing GPU acceleration for real-time tasks since 2011. Its main applications are image processing, video recording and analysis, including object and face recognition. Its main interface is built in C++, but interfaces for Python, Java and MATLAB / OCTAVE are also available.

3) TensorFlow

This open-source AI system uses data streams to build models. Developers can use it to create multi-layered, large-scale neural networks. The main applications of TensorFlow are design, understanding, classification, detection and prediction. Google created it to perform tasks using machine learning, deep learning, and other statistics and predictions. TensorFlow is a tool that can be used to create models for a range of uses, including partial differential modeling, picture recognition, text recognition, and natural language processing.

4) Labellmg

The drawing tool used to label images is called LabelImg, which identifies the boundary boxes of the picture's objects. Using graphic imagery, we can define specific areas of the image and ensure that characters are clearly identified.

5) Mobile SSD net V2

The mobile net SSD model is an SSD network that does multi-shot detection attempts to identify and classify objects by identifying pixels falling in the joint box. Unlike the old model, the design of this model is founded on the idea of the reversed residue model, in which the residue's input and output are limited processes. Additionally, nonlinearities in the middle layers were minimized by using deep integration. This model is part of the TensorFlow object recognition API.

D. Proposed model testing and performance evaluation

This is an important step in evaluating how well your machine-learning model is working. Measure your model's performance using the experimental methods used to train your system. Performance indicators for categorization tasks include F1 scores, recall, accuracy, and precision. You should also examine the confusion matrix to determine in which class the model is most confusing. In the classification problem, there are four terms such that,

True positives (TP): Positive predictions that come true.

False positives (FP): Positive predictions that turn out to be negative.

True negatives (TN): Negative predictions that end up turning out to be negative.

False negatives (FN): Positive predictions that end up turning out to be negative.

There will always be confusion with these terms. Therefore, it is preferable to display these in a matrix format for improved visibility, which also improves understanding.

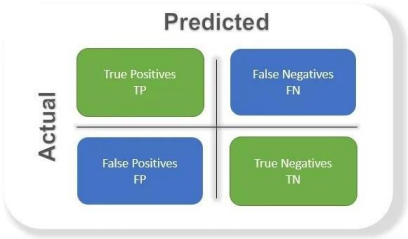

Fig3.2 Confusion Matrix

Accuracy: This tells you the percentage of all samples that the classifier successfully identified, or the overall accuracy of the model. To calculate accuracy, use

=(TN+TP) / (FP+TP+FN+TN)

Precision: The percentage of positive cases out of every positive outcome that was anticipated to happen. The positive prediction of the model over the entire dataset is used as the denominator in this case. Put it in place of figuring out "the level to which the system is true when it appears to be true."

=TP / (FP+TP)

Recall: The percentage of successful cases in relation to all actual successful cases. Thus, in this example, the denominator (FN + TP) represents the actual number of positive results in the dataset. Think of it as an effort to figure out "how many more correct ones the model did not display when it displayed the correct ones."

=TP / (FN+TP)

F1 score: The mean of recall as well as precision is what it is.. Since the F1 score accounts for both contributions, the higher the better. Note that if an F1 score is low due to the product in the numerator, the final score drops significantly. As a result, a model does well in the F1 score if it predicts positives accurately (precision), doesn't miss any positives, and rather predicts negatives (recall).

$$\frac{2}{\frac{1}{precision}+\frac{1}{recall}} = \frac{2+precision+recall}{precision\ +\ recall}$$

Negative is the opposite and truth is just as important. So depending on our application we will need a higher return and the F1 score will not be a good metric. Therefore, it would be useful to look at the PR or ROC curve based on weighted F1 scores.

IV. Result and conclusion

This module reads the individual's hand gestures and responds in actual time via sign language.

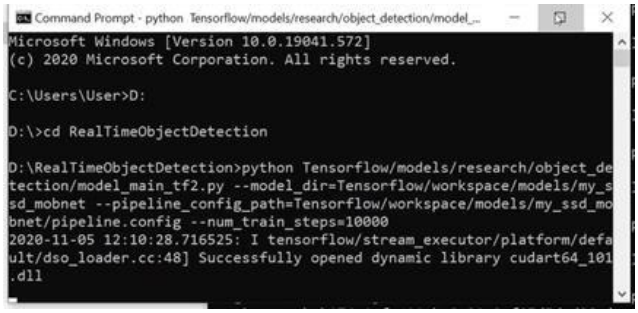

Fig4.1 Install TensorFlow

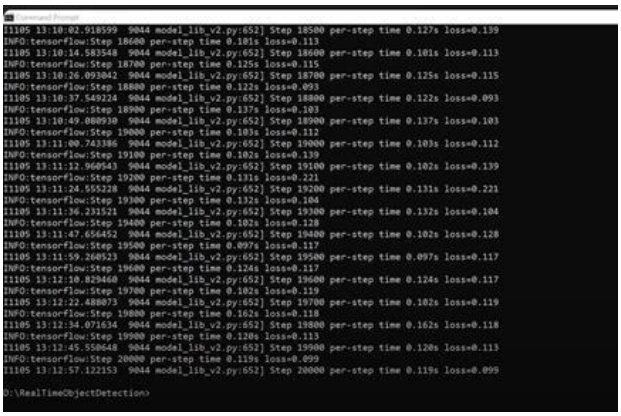

Fig4.2 Install Object Detection

In summary, creating a system that converts instant recognition into text is a complex task that can have significant consequences. For language users, this approach leads to better communication and allows them to connect with the larger community. The information will be expanded in the future to keep the system aware of new aspects. It is also possible to use a different model instead of the TensorFlow model. Language translation of the system is possible by modifying the data collection. We are employing the meaning of Indian sign language in our model; however, we might use another language, such as Spanish or French, in the future.

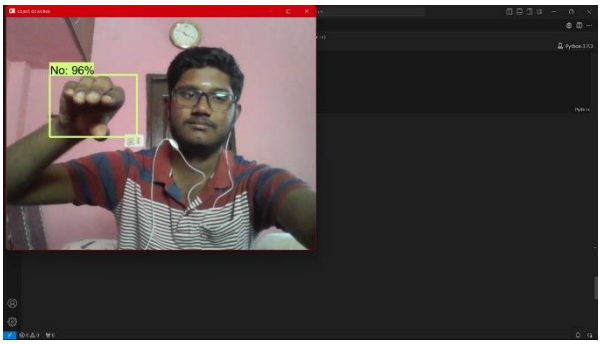

Fig4.3 “No” Sign

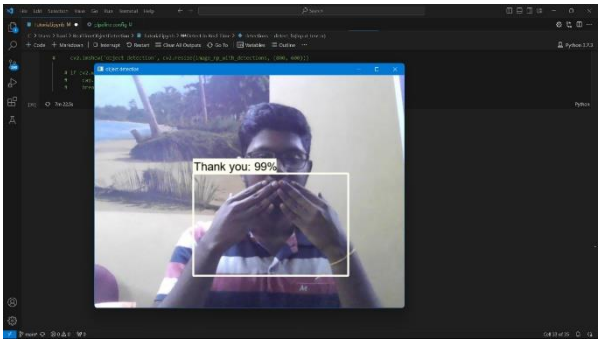

Fig4.4 “Thank You” Sign

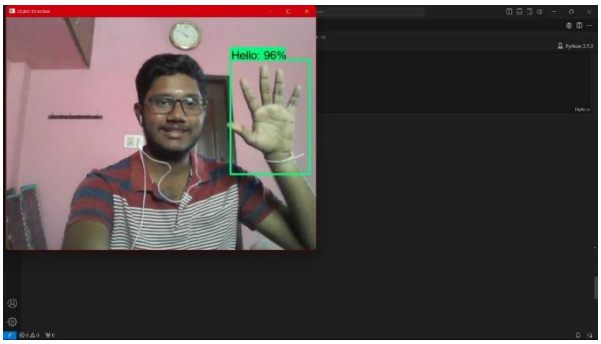

Fig4.5 “**Hello**” Sign

Reference

[1] Kapur, R.: The Types of Communication. MIJ. 6, (2020).

[2] Suharjito, Anderson, R., Wiryana, F., Ariesta, M.C., Kusuma, G.P.: Sign Language Recognition Application Systems for Deaf-Mute People: A Review Based on InputProcess-Output. Procedia Comput. Sci. 116, 441–448 (2017). https://doi.org/10.1016/J.PROCS.2017.10.028.

[3] Konstantinidis, D., Dimitropoulos, K., Daras, P.: Sign language recognition based on hand and body skeletal data. 3DTV-Conference. 2018-June, (2018). https://doi.org/10.1109/3DTV.2018.8478467.

[4] Dutta, K.K., Bellary, S.A.S.: Machine Learning Techniques for Indian Sign Language Recognition. Int. Conf. Curr. Trends Comput. Electr. Electron. Commun. CTCEEC 2017. 333–336 (2018). https://doi.org/10.1109/CTCEEC.2017.8454988.

[5] https://github.com/nicknochnack/RealTimeObject Detection .

[6] https://github.com/HumanSignal/labelImg

[7] Jamie Berke , James Lacy March 01, 2021 “Hearing loss/deafness| Sign Language” https://www.verywellhealth.com/sign-language- nonverbal-users-1046848

[8] “National Health Mission -report of deaf people in India”, nhm.gov.in. 21-12-2021.

[9] “Computer Vision” https://www.ibm.com/in-en/topics/computer-vision

[10] Stephanie Thurrott November 22,2021 “The Best Ways to Communicate with Someone Who Doesn’t Hear Well”

[11] https://www.forbes.com/sites/bernardmarr/2019/04/08/7-amazing-examples-of-computer-and-machine-vision-in-practice/?sh=60a27b1b1018

[12] Juhi Ekbote, M. Joshi Published 1 March 2017Computer Science 2017 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS) DOI: 10.1109/ICIIECS.2017.8276111 Corpus ID: 24740741 Indian sign language recognition using ANN and SVM classifiers

[13] By great learning team,“Real-Time Object Detection Using TensorFlow”, December 25 ,2021, https://www.mygreatlearning.com/blog/object-detection-usingtensorflow/

[14] Jeffery Dean, minute 0:47/2:17 from YouTube clip “TensorFlow: Open source machine learning”. Google 2015. Archived from original on November 11, 2021. “It is a machine learning software being used for various kinds of perceptual and language understanding tasks”

[15] Joseph Nelson “Labellmg for labeling object detection data” March 16,2020

https://blog.roboflow.com/labelimg