JAYALAKSHMI INSTITUTE OF TECHNOLOGY ,THOPPUR

TEAM LEADER:P.DIVYA

TEAM MEMBER:D.ARTHI

SUJATHA

AMRUTHA

TUPDATED ON:2022 NOVEMBER 19
TEAM ID:PNT2022TMID40614

ABSTRACT

Deaf and mute people use sign language to communicate. Unlike acoustically conveyed sound patterns, sign language uses hand gestures, facial expressions, body language and manual communication to convey thoughts. Due to the considerable time required in learning Sign Language, people find it difficult to communicate with specially-abled people, creating a communication gap. Hence conventionally, people face problems in recognizing sign language. Moreover, different countries have their respective form of sign gesture communication which results in non-uniformity. The ISL (Indian Sign Language) used in India is largely different from the American Sign Language used in the US, mostly because of the difference in culture, geographical and historical context. Somewhere between 138 and 300 different types of sign language are currently being used throughout the world. Sign language structure varies spatially and temporally. We have identified these as a major barrier in communication with a significant part of society. And hence, we propose to design a system that recognizes different signs and conveys the information to people.

The component of any sign language consists of hand shape, motion, and place of articulation. When combined, these three components (together with palm orientation) uniquely determine the meaning of the manual sign. For sign language identification, sensor-based and vision-based methods are used. In vision-based gesture recognition technology, a camera reads the movements of the human body, typically hand movements and uses these gestures to interpret sign language, whereas in sensor-based methods, real-time hand and finger movements can be monitored using the leap motion sensor. We aim at developing a scalable project where we will be considering different hand gestures to recognize the letters and words. We plan to use different deep learning models to predict the sign. This may be developed as a desktop or mobile application to enable specially abled people to communicate easily and effectively with others. However, this project can later be extended to capture the whole vocabulary of ASL (American Sign Language) through manual and non-manual signs.

Keywords: *Sign language, ASL, ISL, Dynamic hand gesture recognition*

LIST OF ABBREVIATIONS

Abbreviation	**Description**
API	Application Programming Interface
ASL	American Sign Language
BSL	British Sign Language
HCI	Human Computer Interface
ISL	Indian Sign Language
OpenCV	Open-Source Computer Vision Library
SLR	Sign Language Recognition

LIST OF FIGURES

Page #

Chapter 3.

Figure 3.1. SSD MobileNet Layered Architecture 15

Chapter 4.

Figure 4.1. Detecting “Hello” in Real-time 19

Figure 4.2. Detecting “I Love You” in Real-time 20

Figure 4.3. Detecting “Yes” in Real-time 20

Figure 4.4. Detecting “No” in Real-time 21

Figure 4.5. Detecting “Sorry” in Real-time 21

Figure 4.6 Detecting “Please” in Real-time 22

Figure 4.7 Detecting “Thank You” in Real-time 22

CONTENTS

Chapter 1.	***Introduction***	**1 – 5**
1.1.	**Background**	2
1.2.	**Problem Statement**	2
1.3.	**Proposed Method**	3
	1.3.1. Method to Improve the Performance	3
	1.3.2. Evaluation Parameter	3
1.4.	**Objective and Motivation**	4
1.5.	**Project Organization**	4
	Summary	5
Chapter 2.	***Literature Review***	**6-11**
2.1.	**Introduction**	6
2.2.	**Scope of Work**	7
	2.2.1. Survey of Hand Gesture Recognition Methods in Sign Language Recognition	7
	2.2.2. Communication between deaf and mute people and normal people	8
	2.2.3. Intelligent sign language recognition using Image Processing	8
	2.2.4. Sign language recognition using Image Processing	9
	2.2.5. Sign Language Interpreter using Image Processing and Machine Learning	9
	2.2.6. Hand gesture recognition based on Digital Image Processing using MATLAB	10
	2.2.7. Gesture Recognition System	10
	Summary	11

Chapter 3.	***Methodology***	**12 – 18**
3.1. Introduction		12
3.2. Tools Used		12
3.2.1. LabelImg		12
3.2.2. Tensorflow Object Detection API		12
3.2.3. Transfer Learning		13
3.2.4. SSD MobileNet		14
3.2.5. OpenCV		15
3.2.6. Google Colab		17
Summary		18
Chapter 4.	***Experimental Results***	**19 – 23**
4.1. Introduction		19
4.2. Results and Discussion		19
Summary		23
Chapter 5.	***Conclusion and Future Scope***	**24-25**
5.1. Future scope		24
5.2. Conclusion		25
References		**26**
Appendix: List of Formulas Used		27

CHAPTER 1

INTRODUCTION

Communication is very crucial to human beings, as it enables us to express ourselves. We communicate through speech, gestures, body language, reading, writing or through visual aids, speech being one of the most used among them. The study by the World Health Organization (WHO) reports that more than 7% of the world’s population has hearing impairment. It is estimated that about 900 million people will experience hearing loss in 2050. According to WHO 2018 report in India, about 63 million people suffer from hearing impairment. Those suffering from speech and hearing loss find it impossible to communicate with the conventional world and vice versa. Use of sign language is the only way to communicate. But not all people understand sign language, and hence, visual aids or an interpreter are used for communication. However, these methods are rather cumbersome and expensive, and can't be used in an emergency. Sign Language mainly uses manual communication to convey meaning. This involves simultaneously combining hand shapes, orientations, gestures and movement of the hands, arms, or body to express the speaker's thoughts.

Sign Language consists of fingerspelling, which spells out words character by character, and word level association which involves hand gestures that convey the word meaning. Fingerspelling is a vital tool in sign language, as it enables the communication of names, addresses and other words that do not carry a meaning in word level association. Despite this, fingerspelling is not widely used as it is challenging to understand and difficult to use. Moreover, there is no universal sign language and very few people know it, which makes it an inadequate alternative for communication.

A system for sign language recognition that classifies hand gestures and finger spelling can solve this problem. Various techniques are used, and their accuracies are recorded and compared in this report.

BACKGROUND

Sign language is defined as an organized set of hand gestures with specific meanings used by deaf people to communicate in their daily life. Being a visual language, it uses hand, face, and body movements as a means of communication. More than 300 different sign languages are available worldwide. Despite the diversity of sign language languages, the proportion of the population who knows them is low, making it difficult for people with special abilities to communicate freely with everyone. SLR is a method of communicating in sign language without prior knowledge of sign language. It recognizes gestures and translates them into a common language such as English. SLR is a very large research topic with a lot of work done but much more to be addressed. Machine learning technologies allow electronic systems to make decisions based on experience, or data. A classification algorithm requires two data sets: a training data set and a test data set. The training set gives the classifier experience, and the model is tested using the test set. Many authors have developed efficient methods for collecting and classifying data. Depending on the data collection method, previous work can be divided into two approaches: direct measurement methods and vision-based methods. Direct measurement methods are based on motion data gloves, motion capture systems or sensors.

PROBLEM STATEMENT

The Deaf and mute community can only communicate using sign language. Sign language involves simultaneously combining hand shapes, orientations, gestures and movement of the hands, arms, or body to express the speaker's thoughts. Because of cultural, geographic and historical differences, there exists over 300 different types of sign languages around the world. The ISL (Indian Sign Language) used in India is very different from the American Sign Language used in the United States. This causes inconsistency of sign languages around the world.

Moreover, learning sign language requires significant amount of time and effort. This makes it difficult for the conventional world to learn and hence interact with the deaf and mute community. According to a recent study, out of every thousand kids born, 2 to 3 of them are deaf or hard-of-hearing, and, as degrees of hearing loss go, there are 16 to 30 times more children who are identified as Deaf (having a Profound 91+dB hearing loss) than hard-of-hearing. For those deaf or hard of hearing children, only 10% of parents & family learn sign language to communicate with them.

We identify this as a major barrier in communicating with a significant part of the society.

PROPOSED METHOD

Method to Improve the Performance

This project aims at continuous improvement by targeting higher accuracy through continuous training of datasets and inclusion of new sign languages.

As of now, our model can recognize 7 sign languages – Hello, I Love You, Yes, No, Sorry, Thank You and Please.

To enable continuous improvement, we are aiming at two methods:

i) Changing hyperparameter values during training to get the optimal results: We tried training with 5000, 10000, 15000 and 20000 training steps, and found that we're getting the optimal results for 10000 training steps.

ii) Changing the dataset slightly and re-labelled the changed images to avoid overfitting issues: We re-trained the changed dataset and found that model is predicting better without overfitting.

Evaluation Parameter

The evaluation parameter of this project is the accuracy of predicting the class to which the given sign language poses belong.

On showing a given sign language, the model shows both the predicted class of sign language and accuracy simultaneously. This acts as the evaluation parameter which depicts the progress of the project.

OBJECTIVE AND MOTIVATION

The objective of our project is to bridge the gap and ensure the inclusion of deaf and mute community into the conventional society meanwhile ensuring an easy and effective mode of communication. We aim at designing a real time system that recognizes the sign language and expresses the same in an easy language, like English.

Currently, extensive work has been done on American sign language recognition, but Indian sign language differs significantly from American sign language. ISL uses two hands for communicating (20 out of 26) whereas ASL uses single hand for communicating. Using both hands often lead to obscurity of features due to overlapping of hands. In addition to this, lack of datasets and variance in sign language with locality has resulted in restrained efforts in ISL gesture detection.Our project aims at taking the basic step in bridging the communication gap between normal people and deaf and dumb people using Indian sign language. Effective extension of this project to words and common expressions may not only make the deaf and mute people communicate faster and easier with outer world, but also provide a boost in developing autonomous systems for understanding and aiding them.

PROJECT ORGANIZATION

Chapter 1*"Introduction"* presents a general overview of the concept of Sign Language Recognition and other aspects involving it. It also contains the objective and motivation behind the work.

Chapter 2*"Literature review"* explains various other works and the technologies that are used for Sign Language Recognition.

Chapter 3 named as, *"Proposed Method",* includes the different methods and tools used in making this project.

Chapter 4 titled, *"Experimental Results",* describes the observations of different sign languages that our model is trained in.

Chapter 5 titled, *"Conclusion",* is the summary of the complete work carried out with a miniature part given to the society.

SUMMARY

In this chapter we discussed about:

- The basic concept and goals of the Real Time Sign Language Recognition System
- The purpose of the project and the motivation behind it.
- Parameter to better the project has also been discussed.

CHAPTER 2

LITERATURE REVIEW

INTRODUCTION

Chuan CH, Regina E, Guardino C (2014) American Sign Language recognition using leap motion sensor. In: 13th IEEE international conference on machine learning and applications (ICMLA), pp 541–544 Chuan et al. developed an American Sign Language recognition system using the leap motion sensor. The system was classified using K-Nearest Neighbor and Support Vector machine and accuracy of 72.78% and 79.83% were achieved respectively.

"Deep Convolutional Neural Networks for Sign Language Recognition" G.Anantha Rao, Guntur (DT) Extraction of complex head and hand movements along with their constantly changing shapes for recognition of sign language is considered a difficult problem in computer vision.

K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. 2D CNN is used to extract spatial features of input images while RNN is employed to capture the long-term temporal dependencies among input video frames. VGG16 pre-trained on ImageNet to extract spatial features and then feed the extracted features to a stacked GRU.

J. Carreira and A. Zisserman. Quo Vadis, action recognition? a new model and the kinetics dataset. CVPR, 2017. 3D convolutional networks are used which are able to establish not only the holistic representation of each frame but also the temporal relationship between frames in a hierarchical fashion. Inflate 2D filters of the Inception network trained on ImageNet, thus obtaining well-initialized 3D filters.

Recognizing American Sign Language Gestures from within Continuous Videos One of the main challenges is that in actions in continuous videos, the temporal boundaries of a specific movement are not very clear. This paper detects their temporal locations from within continuous videos, by collecting an ASL dataset that has been annotated with the time-intervals for each ASL word.

SCOPE OF WORK

The following are the summaries of each project that we have referenced:

A Survey of Hand Gesture Recognition Methods in Sign Language Recognition

Sign Language Recognition (SLR) system, which is required to recognize sign languages, has been widely studied for years. The studies are based on various input sensors, gesture segmentation, extraction of features and classification methods. This paper aims to analyze and compare the methods employed in the SLR systems, classification methods that have been used, and suggests the most promising method for future research. Due to recent advancement in classification methods, many of the recent proposed works mainly contribute on the classification methods, such as hybrid method and Deep Learning. This paper focuses on the classification methods used in prior Sign Language Recognition system. Based on our review, HMM-based approaches have been explored extensively in prior research, including its modifications.

This study is based on various input sensors, gesture segmentation, extraction of features and classification methods. This paper aims to analyze and compare the methods employed in the SLR systems, classifications methods that have been used, and suggests the most reliable method for future research. Due to recent advancement in classification methods, many of the recently proposed works mainly contribute to the classification methods, such as hybrid method and Deep

Learning. Based on our review, HMM-based approaches have been explored extensively in prior research, including its modifications. Hybrid CNN-HMM and fully Deep Learning approaches have shown promising results and offer opportunities for further exploration.

Communication between Deaf and Mute People and Normal People

Chat applications have become a powerful media that assist people to communicate in different languages with each other. There are lots of chat applications that are used different people in different languages but there is not such a chat application that has facilitated to communicate with sign languages. The developed system is based on Sinhala Sign language. The system has included four main components as text messages are converted to sign messages, voice messages are converted to sign messages, sign messages are converted to text messages and sign messages are converted to voice messages. Google voice recognition API has used to develop speech character recognition for voice messages. The system has been trained for the speech and text patterns by using some text parameters and signs of Sinhala Sign language is displayed by emojis. Those emojis and signs that are included in this system will bring the normal people closer to the disabled people. This is a 2-way communication system, but it uses pattern of gesture recognition which is not very reliable in getting appropriate output.

Intelligent Sign Language Recognition Using Image Processing

Computer recognition of sign language is an important research problem for enabling communication with hearing impaired people. This project introduces an efficient and fast algorithm for identification of the number of fingers opened in a gesture representing an alphabet of the Binary Sign Language. The system does not require the hand to be perfectly aligned to the camera. The project uses image processing system to identify, especially English alphabetic sign language used by the deaf people to communicate. The basic objective of this project is to develop a computer based intelligent system that will enable dumb people significantly to communicate with all other people using their natural hand gestures. The idea consisted of designing and building up an intelligent system using image processing, machine learning and artificial intelligence concepts to take visual inputs of sign language's hand gestures and generate easily recognizable form of outputs. Hence the objective of this project is to develop an intelligent system which can act as a translator between the sign language and the spoken language dynamically and can make the communication between people with hearing impairment and normal people both effective and efficient. The system is we are implementing for Binary sign language, but it can detect any sign language with prior image processing.

Sign Language Recognition Using Image Processing

One of the major drawbacks of our society is the barrier that is created between disabled or handicapped persons and the normal person. Communication is the only medium by which we can share our thoughts or convey the message but for a person with disability (deaf and mute) faces difficulty in communication with normal person. For many deaf and dumb people, sign language is the basic means of communication. Sign language recognition (SLR) aims to interpret sign languages automatically by a computer in order to help the deaf communicate with hearing society conveniently. Our aim is to design a system to help the person who trained the hearing impaired to communicate with the rest of the world using sign language or hand gesture recognition techniques. In this system, feature detection and feature extraction of hand 23 gesture is done with the help of SURF algorithm using image processing. All this work is done using MATLAB software. With the help of this algorithm, a person can easily train a deaf and mute.

Sign Language Interpreter using Image Processing and Machine Learning

Speech impairment is a disability which affects one's ability to speak and hear. Such individuals use sign language to communicate with other people. Although it is an effective form of communication, there remains a challenge for people who do not understand sign language to communicate with speech impaired people. The aim of this paper is to develop an application which will translate sign language to English in the form of text and audio, thus aiding communication with sign language. The application acquires image data using the webcam of the computer, then it is preprocessed using a combinational algorithm and recognition is done using template matching. The translation in the form of text is then converted to audio. The database used for this system includes 6000 images of English alphabets. We used 4800 images for training and 1200 images for testing. The system produces 88% accuracy.

Hand Gesture Recognition based on Digital Image Processing using MATLAB

This research work presents a prototype system that helps to recognize hand gesture to normal people in order to communicate more effectively with the special people. Aforesaid research work focuses on the problem of gesture recognition in real time that sign language used by the community of deaf people. The problem addressed is based on Digital Image Processing using Color Segmentation, Skin Detection, Image Segmentation, Image Filtering, and Template Matching techniques. This system recognizes gestures of ASL (American Sign Language) including the alphabet and a subset of its words.

Gesture Recognition System

Communication plays a crucial part in human life. It encourages a man to pass on his sentiments, feelings and messages by talking, composing or by utilizing some other medium. Gesture based communication is the main method for Communication 24 for the discourse and hearing weakened individuals. Communication via gestures is a dialect that utilizations outwardly transmitted motions that consolidates hand signs and development of the hands, arms, lip designs, body developments and outward appearances, rather than utilizing discourse or content, to express the individual's musings. Gestures are the requirement for hearing and discourse hindered, they pass on their message to others just with the assistance of motions. Gesture Recognition System is the capacity of the computer interface to catch, track and perceive the motions and deliver the yield in light of the caught signals. It enables the clients to interface with machines (HMI) without the any need of mechanical gadgets. There are two sorts of sign recognition methods: image- based and sensor-based strategies. Image based approach is utilized as a part of this project that manages communication via gestures motions to distinguish and track the signs and change over them into the relating discourse and content.

SUMMARY

In this chapter we discussed about:

- The different methods that have been in use to develop a sign language recognition system
- The use of different methods to recognize real-time video and their processing.

CHAPTER 3

PROPOSED METHOD

INTRODUCTION

Our proposed system is recognizing sign language using TensorFlow Object Detection API by performing transfer learning using SSD MobileNet. It recognizes seven hand gestures by capturing photos from real time using a webcam. Then the hand pixels are segmented and the image it obtained and sent for comparison to the trained model. Thus, our system is more robust in getting exact text labels of letters using LabelImg for labeling.

TOOLS USED

The following tools have been used in implementing the project:

LabelImg

LabelImg 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.

We have used LabelImg in local machine because it was not supported by Google Collab.

TensorFlow Object Detection API

TensorFlow Object Detection API, an open-source framework for object detection related tasks, was used for training and testing an SSD (Single-Shot Multibox Detector) with MobileNetmodel. The model was tested as a) pre-trained and b) with fine-tuning with a dataset consisting of images of seven different American sign language poses. It is a free and open-source software library for dataflow and differentiable programming across a range of tasks. It is a symbolic math library and is also used for machine learning applications such as neural networks. It is used for both research and production at Google.

Features: TensorFlow provides stable Python (for version 3.7 across all platforms) and C APIs; and without API backwards compatibility guarantee: C++, Go, Java, JavaScript and Swift (early release). Third-party packages are available for C#, Haskell Julia, MATLAB, R, Scala, Rust, OCaml, and Crystal. "New language support should be built on top of the C API. However, not all functionality is available in C yet." Some more functionality is provided by the Python API.

Application: Among the applications for which TensorFlow is the foundation, are automated image-captioning software, such as DeepDream.

Transfer Learning

Transfer learning is used to improve a learner from one domain by transferring information from a related domain. We can draw from real-world non-technical experiences to understand why transfer learning is possible. Consider an example of two people who want to learn to play the piano. One person has no previous experience playing music, and the other person has extensive music knowledge through playing the guitar. The person with an extensive music background will be able to learn the piano in a more efficient manner by transferring previously learned music knowledge to the task of learning to play the piano. One person is able to take information from a previously learned task and use it in a beneficial way to learn a related task.

Looking at a concrete example from the domain of machine learning, consider the task of predicting text sentiment of product reviews where there exists an abundance of labeled data from digital camera reviews. If the training data and the target data are both derived from digital camera reviews, then traditional machine learning techniques are used to achieve good prediction results. However, in the case where the training data is from digital camera reviews and the target data is from food reviews, then the prediction results are likely to degrade due to the differences in domain data. Digital camera reviews and food reviews still have several characteristics in common, if not exactly the same. They both are written in textual form using the same language, and they both express views about a purchased product. Because these two domains are related, transfer learning can be used to potentially improve the results of a target learner. An alternative way to view the data domains in a transfer learning environment is that the training data and the target data exist in different sub-domains linked by a high-level common domain. For example, a piano player and a guitar player are subdomains of a musician domain. Further, a digital camera review and a food review are subdomains of a review domain. The high-level common domain determines how the subdomains are related.

As previously mentioned, the need for transfer learning occurs when there is a limited supply of target training data. This could be due to the data being rare, the data being expensive to collect and label, or the data being inaccessible. With big data repositories becoming more prevalent, using existing datasets that are related to, but not the same as, a target domain of interest makes transfer learning solutions an attractive approach

SSD MobileNet

By using SSD, we only need to take one single shot to detect multiple objects within the image, while regional proposal network (RPN) based approaches such as R-CNN series that need two shots, one for generating region proposals, one for detecting the object of each proposal. Thus, SSD is much faster compared with two-shot RPN-based approaches.
SSD is a single-shot detector. It has no delegated region proposal network and predicts the boundary boxes and the classes directly from feature maps in one single pass.

To improve accuracy, SSD introduces:

- small convolutional filters to predict object classes and offsets to default boundary boxes.
- separate filters for default boxes to handle the difference in aspect ratios.
- multi-scale feature maps for object detection.

SSD can be trained end-to-end for better accuracy. SSD makes more predictions and has better coverage on location, scale, and aspect ratios. With the improvements above, SSD can lower the input image resolution to 300 × 300 with a comparative accuracy performance. By removing the delegated region proposal and using lower resolution images, the model can run at real-time speed and still beats the accuracy of the state-of-the-art Faster R-CNN.

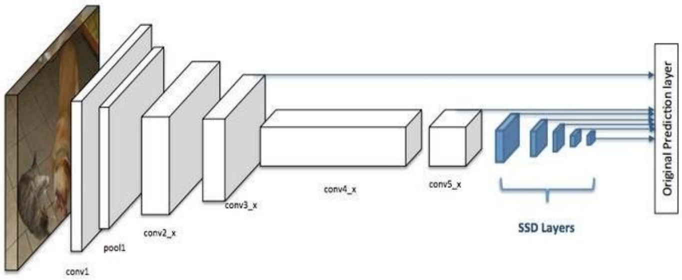

Figure 3.1. SSD MobileNet Layered Architecture

Open CV

OpenCV (Open-Source Computer Vision Library) is a library of programming functions mainly aimed at real-time computer vision. Originally developed by Intel, it was later supported by Willow Garage then Itseez.
The library is cross-platform and free for use under the open-source license.

OpenCV's application areas include:

- 2D and 3D feature toolkits
- Egomotion estimation
- Facial recognition system
- Gesture recognition
- Human–computer interaction (HCI)
- Mobile robotics
- Motion understanding
- Object identification
- Segmentation and recognition

Stereopsis stereo vision: depth perception from 2 cameras

- Structure from motion (SFM).
- Motion tracking
- Augmented reality

To support some of the above areas, OpenCV includes a statistical machine learning library that contains:

- Boosting
- Decision tree learning
- Gradient boosting trees
- Expectation-maximization algorithm
- k-nearest neighbor algorithm
- Naive Bayes classifier
- Artificial neural networks
- Random forest
- Support vector machine (SVM)
- Deep neural networks (DNN)

Image-Processing:

Image processing is a method to perform some operations on an image, in order to get an enhanced image and or to extract some useful information from it.
If we talk about the basic definition of image processing then "Image processing is the analysis and manipulation of a digitized image, especially in order to improve its quality".

Digital-Image:

An image may be defined as a two-dimensional function f(x, y), where x and y are spatial(plane) coordinates, and the amplitude of fat any pair of coordinates (x, y) is called the intensity or grey level of the image at that point.

In another word An image is nothing more than a two-dimensional matrix (3-D in case of colored images) which is defined by the mathematical function f(x, y) at any point is giving the pixel value at that point of an image, the pixel value describes how bright that pixel is, and what color it should be.

Image processing is basically signal processing in which input is an image and output is image or characteristics according to requirement associated with that image.

Image processing basically includes the following three steps:

- Importing the image
- Analyzing and manipulating the image

Output in which result can be altered image or report that is based on image analysis

Google Colab

Colab is a free Jupyter notebook environment that runs entirely in the cloud. Most importantly, it does not require a setup and the notebooks can be created and simultaneously edited by multiple users. Colab supports many popular machine learning libraries which can be easily loaded in notebook.Colab is a free Jupyter notebook environment that runs entirely in the cloud.

The advantages of using Google Colab are:

- Pre-Installed Libraries: Anaconda distribution of Jupyter Notebook shipped with several pre-installed data libraries, such as Pandas, NumPy, Matplotlib, which is awesome. Google Colab, on the other hand, provides even more pre-installed machine learning libraries such as Keras, TensorFlow, and PyTorch.

- Saved on the Cloud: This feature makes all the work accessible from any device. The need of setting up a local machine does not arise. All the Gogle Colab notebooks are saved under Google Drive account and are just a login away.

- Collaboration: Another great feature that Google Colab offers is the collaboration feature. If you are working with multiple developers on a project, it is great to use Google Colab notebook. Just like collaborating on a Google Docs document, you can co-code with multiple developers using a Google Colab notebook. Besides, you can also share your completed work with other developers.

- Free GPU and TPU Use: Google Research lets you use their dedicated GPUs and TPUs for your personal machine learning projects. Sometimes for some projects, the GPU and TPU acceleration make a huge difference even for some small projects. This is one of the main reasons to code projects on Google Colab. Besides, since it uses Google resources, the neural network optimization operations do not mess with the processors, and my cooling fan doesn't get overused.

SUMMARY

In this chapter we discussed about:

- LabelImg, which is used to label the sign language poses after data collection.
- TensorFlow Object Detection API used to detect the sign language gesture.
- Transfer Learning which enables easy training of datasets
- SSD MoblieNet, which is needed to take one single shot to detect multiple objects within the image
- OpenCV, which was usedto display the result of the detected images
- Google Colab, which has been used as the substitute for local machine, and as the platform to setup the environment.

CHAPTER 4

EXPERIMENTAL RESULTS

INTRODUCTION

Our model performs transfer learning against the TensorFlow Object Detection API to train the object detector and finally uses JavaScript and python to detect the sign language gestures in real time. It detects seven significant American sign language gestures namely “Hello”, “Yes”, “No”, “I love you”, “Sorry”, “Please” and “Thank You”. Our model uses JavaScript to use the webcam from Google Colab to capture real-time data, it captures images and processes the frame to detect these sign language poses with up to 97% accuracy.

RESULTS AND DISCUSSION

After configuring the model and then training it for 10,000 steps, we the trained model from checkpoint 11 and began the real time detection process. These were the results we got:

i. Detecting “Hello” in real-time

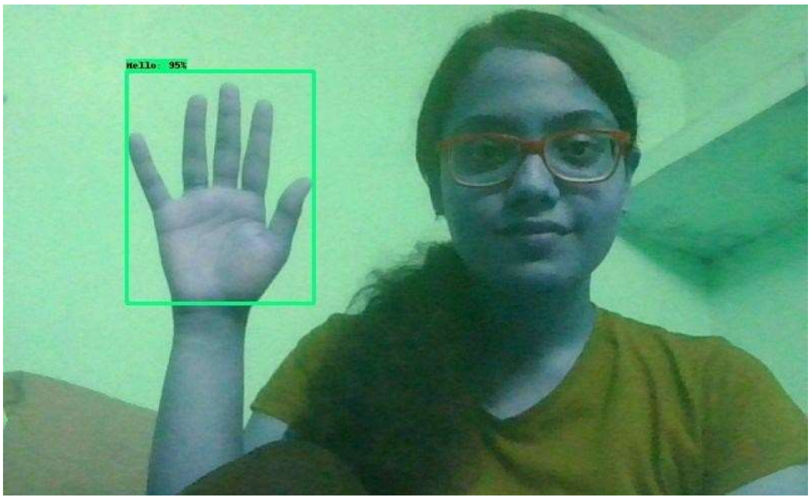

Figure 4.1

ii. Detecting “I love you” in real-time

Figure 4.2

iii. Detecting “Yes” in real-time

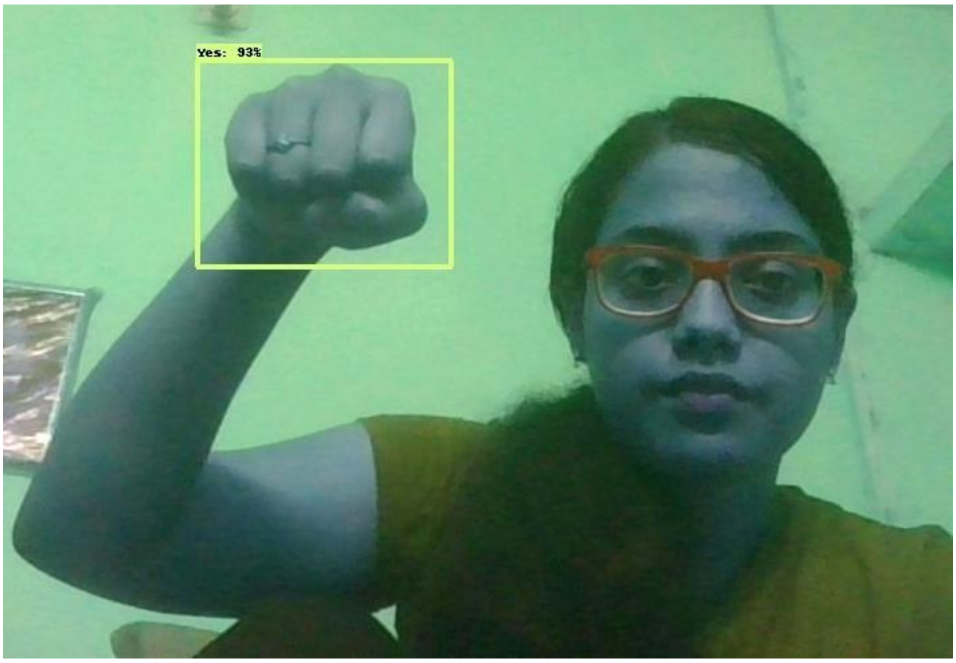

Figure 4.3

iv. Detecting “No” in real-time

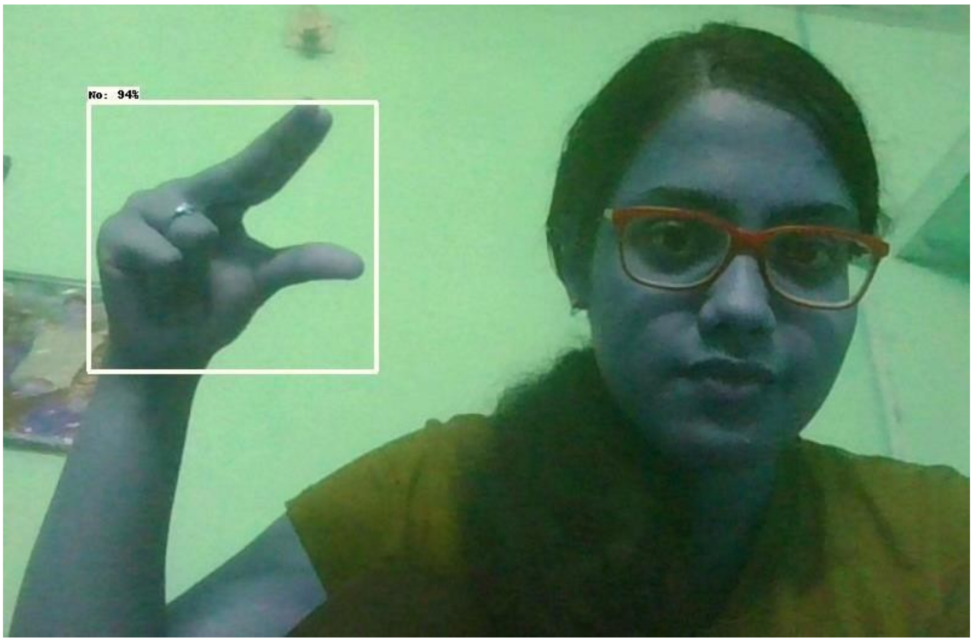

Figure 4.4

v. Detecting “Sorry” in real-time

Figure 4.5

vi. Detecting “Please” in real-time

Figure 4.6

vii. Detecting “Thank You” in real-time

Figure 4.7

SUMMARY

In this chapter we discussed about:

- The different sign language that is currently being recognized by our model.

CHAPTER 5

FUTURE SCOPE AND CONCLUSION

Future Scope

In future works, the model can be improved upon the following points:

i) Upgrading the dataset and training to recognize combination of static gestures: Some words like "Bye" use multiple hand gestures to signify one word. Also, when speaking a sentence, a lot of gestures are used. The model can be efficiently trained to implement these features.

ii) Interpretation of facial expression by real-time system: A few words like "Congratulations" "Sorry", etc. use facial expressions to convey the meaning. Our system can only recognize hand-gestures for now. In future, inclusion of facial expressions will ensure addition of more words and better accuracy of result.

iii) Inclusion of different types of sign language: As previously stated, there are over 300 different types of sign languages used around the world. Our model recognizes ASL and ISL types only. But with training more datasets, the model can be advanced for use around the world.

iv) Integrating more words into the vocabulary: Currently, our model can accurately recognize only 7 sign language - Hello, I Love You, Yes, No, Sorry, Thank You and Please. With inclusion of more words into the vocabulary, the model can be widely used.

v) Use of advanced technologies: With time, the technologies advance, and in future, the different tools can be upgraded to include a more efficient tool and hence more accurate results. TensorFlow model that has been used can be interchanged with another model as well.

Conclusion

Sign languages are kinds of visual languages that employ movements of hands, body, and facial expression as a means of communication. Sign languages are important for especially abled people to have a means of communication. Through it, they can communicate and express and share their feelings with others. The drawback is that not everyone possesses the knowledge of sign languages which limits communication. This limitation can be overcome using automated Sign Language Recognition systems which will be able to easily translate the sign language gestures into commonly spoken language. In this paper, it has been done by TensorFlow object detection API. The system has been trained on the American Sign Language alphabet dataset. The system detects sign language in real-time. For data acquisition, images have been captured by a webcam using OpenCV which makes the cost cheaper. Though the system has achieved a high average confidence rate, the dataset it has been trained on is small and limited.

REFERENCES

[1] Daiyi Peng, Xuanyi Dong, Esteban Real, Mingxing Tan, Yifeng Lu Hanxiao Liu, Gabriel Bender, Adam Kraft, Chen Liang, Quoc V. Le, "PyGlove: Symbolic programming for automated machine learning", Advances in Neural Information Processing Systems, Vol. 33, pp. 96-108, 2020.

[2] Zhang Shujun and Qun Zhang, "Sign language recognition based on global-local attention", Journal of Visual Communication and Image Representation, Vol. 80, Article id: 103280, 2021.

[3] I. A. Adeyanju, O. O. Bello, and M. A. Adegboye, "Machine learning methods for sign language recognition: A critical review and analysis", Intelligent Systems with Applications, Vol. 12, Article id: 200056, 2021.

[4] Brandon Garcia, Sigberto Alarcon Viesca, “Real-time American Sign Language Recognition with Convolutional Neural Networks”, Stanford University Stanford, CA, 2016

[5] Sawant Pramada, Deshpande Saylee, Nale Pranita, Nerkar Samiksha, Mrs.Archana S. Vaidya, “Intelligent Sign Language Recognition Using Image Processing”, IOSR Journal of Engineering (IOSRJEN), Volume 3, Issue 2, PP 45-51, 2013

[6] Pratibha Pandey , Vinay Jain, “An Efficient Algorithm for Sign Language Recognition”, (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 6, 2015

[7] Anup Kumar, Karun Thankachan, Mevin, M. Dominic, “Sign Language Recognition”, IEEE-India, 2016

[8] Ashish S. Nikam, Aarti G. Ambekar, “Sign language recognition using image based hand gesture recognition techniques”, IEEE-India, 2016

[9] Nicholas Renotte, *‘RealTimeObjectDetection’*, 2020 [Online]. Available: https://github.com/nicknochnack [Accessed: 27-Feb-2022]

[10] Vaishshells. *‘Sign Language Recognition for Computer Vision Enthusiasts’*, 2021 [Online]. Available: https://www.analyticsvidhya.com/blog/2021/06/sign-language-recognition-for-computer-vision-enthusiasts/ [Accessed: 17-Jan-2022]

[11] João Carreira, Andrew Zisserman, “Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset”, IEEE-USA, 2017

[12] Ching-Hua Chuan, Eric Regina, Caroline Guardino, “American Sign Language Recognition Using Leap Motion Sensor”, IEEE-USA, 2014

[13] Data Flair, ‘*Sign Language Recognition Using Python and OpenCV’*, 2022 [Online]. Available: https://data-flair.training/blogs/sign-language-recognition-python-ml-opencv/ [Accessed: 12-Jan-2022]

[14] Yuancheng Ye, Yingli Tian, Matt Huenerfauth, and Jingya Liu, “Recognizing American Sign Language Gestures from within Continuous Videos”, The Graduate, The City College, Rochester Institute of Technology, Rochester, New York, USA, 2018

APPENDIX

LIST OF FORMULAS USED

1. Momentum Optimizer :

$$\nu_j \leftarrow \eta * \nu_j - \alpha * \nabla_w \sum_{1}^{m} L_m(w)$$
$$\omega_j \leftarrow \nu_j + \omega_j$$