International Journal of Advances in Engineering and Management (IJAEM)
Volume 3, Issue 7 July 2021, pp: 1261-1267 www.ijaem.net ISSN: 2395-5252

Sign Language Auto Completer

Sujay ingle, Yash N Bhuse, ShubhamJagtap, Viraj Gore

Submitted: 25-06-2021 Revised: 04-07-2021 Accepted: 07-07-2021

ABSTRACT - Sign language is one of the means of communication through body moments, with the use of hands and arms, especially when spoken or verbal communication is not possible. Sign language is mostly being used for people who have hearing and speaking disabilities called Deaf and Mute. Even normal people face difficulties while talking with the deaf and dumb community as they are unaware of the meaning of various gestures that are part of sign language. Hence, a intelligent system is required to be developed and be taught so to make the deaf and dumb community people feel part of the society by helping them communicate freely.

Key words:Python,IDE, Numpy, OpenCV,Keras,TensorFlow

I. INTRODUCTION:

In the system the unable or dumb person should provide a gesture or sign image to the system, The software recognizes the sign gesture made in front of the screen using deep learning and try to make similar type of word related to the letters recognized.

The system has an auto-completer which will automatically complete the entire word for the user. The software helps the dumb person to complete the sentence as well as pronouns it in various other language like Hindi. The software thus helps the person to communicate with others in their own language even if they don't know one of the most abundantly spoken languages 'English'. The aim of this system is to provide an application to society to establish the ease of communication between deaf and mute people by making use of image processing algorithms.

II. LITERATURE REVIEW:

1. An Efficient Framework for Indian Sign Language Recognition Using Wavelet Transform :

Feature extraction and classification are the two important modules in the following ISLR gesture recognition technology. The joint use of Discrete Wavelet Transform (DWT) based feature extraction and nearest neighbour classifier is used to recognize the sign language. While using cosine distance classifier, the proposed hand gesture recognition system achieves maximum 99.23% accuracy.

2. Hand Gesture Recognition System For Dumb People:

Using digital image processing, the authors presented the static hand gesture recognition system. Vector SIFT algorithm is used for hand recognition features. Edges which are invariant to scaling, rotation, addition of noise are subject to computation of SIFT features.

3. Hand Gesture Recognition for Sign Language Recognition:

Authors presented various methods of hand gesture and sign language recognition proposed in the past by various researchers. Sign language is the only way of communication for people with hearing and speaking problems. These physically impaired people express their emotions and thoughts to other people with the help of sign language.

4. SignPro-An Application Suite for Deaf and Dumb :

Author presented application that helps the deaf and dumb person to communicate with the rest of the world using sign language. The real time gesture to text conversion is the most rewarding feature of this system. Gesture extraction, gesture matching and conversion to speech are its fundamental steps. Use of various image processing techniques such as histogram matching, bounding box computation, skin colour segmentation and region growing are a part of Gesture extraction. Feature point matching and correlation based matching are techniques applicable for Gesture matching. Voicing out of text and text to gesture conversion are the other features of the application.

5. A Review on Feature Extraction for Indian and American Sign Language:

Paper presented the recent research and development of sign language based on manual communication and body language. Pre-processing, feature extraction and classification are three steps elaborated by the sign language recognition system. Neural Network(NN), Support Vector Machine(SVM), Hidden Markov

International Journal of Advances in Engineering and Management (IJAEM)
Volume 3, Issue 7 July 2021, pp: 1261-1267 www.ijaem.net ISSN: 2395-5252

Models(HMM), Scale Invariant Feature Transform(SIFT),etc. are some of the methods used for recognition.

III. PROPOSED METHODOLOGY:

3.1 Technology Used & its Function:

3.1.1 Convolutional Neural Networks (CNN):

In deep learning, a convolutional neural network is a class of deep neural networks, most commonly used to analyze visual imagery. They are also known as space invariant artificial neural networks(SIANN), based on the shared-weight architecture of the convolutional that slides along input features. CNNs are regularized versions of multilayer perceptrons. CNNs use relatively little pre-processing compared to other image classification algorithms. This means that the network learns to optimize thefilters (or kernels) through automated learning, whereas in traditional algorithms these filters arehand-engineered. This independence from prior knowledge and human intervention in feature extraction is a major advantage.

3.1.1.1 CNN for SLAC: In SLAC, the CNN algorithm developed, captures the images in the form of input and assigns different weights and biases to it, so that it could be differentiated with other images. The CNN distinctively recognizes the images at better speed than the other algorithms of classification. This enables the
SLAC software to determine Sign Language
Imagery efficiently.

3.1.1.2 Challenges due CNN: During the development of SLAC, it is found that CNN requires a large dataset to perform and train the model. Hence, around 12,000 images were created to fulfill the requirement for CNN.

It was also found that the Computer should have a good GPU, because the CNN had several layers.

CNN performance also depends upon the quality of hardware devices used to capture the imagery.

3.1.2 TensorFlow:

TensorFlow is Google Brain's second-generation system. TensorFlow can run on multipleCPUsandGPUs (with optionalCUDA. TensorFlow only works in CUDA enabled GPUs. CUDA is a proprietary application programming interface made by Nvidia. TensorFlow is available on 64-bit Linux, macOS, Windows, and mobile computing platforms including Android and IOS.

Its flexible architecture allows for the easy deployment of computation across a variety of platforms (CPUs, GPUs,TPUs), and from desktops to clusters of servers to mobile and edge devices. Tensorflow enjoys a lot of community support. Tensorflow matches the industry standards as it is performance efficient.

3.1.2.1 Tensorflow for SLAC:

In SLAC, the Tensorflow framework enables the system to store the data which is in complex format in a much more closed-packed way.

3.1.2.2 Challenges due to Tensorflow:

While working on the SLAC project, it was found that Tensorflow works better in the LINUX environment than the WINDOWS.

It was found that as Tensorflow has a distinctive structure, it was challenging to find any bugs.

3.1.3 Keras:

Keras is anopen-sourcesoftware library that provides aPython interface forartificialneural networks. Keras acts as an interface for theTensorFlow library. Keras contains numerous implementations of commonly used neural-network building blocks such as
layers,objectives,activation
functions,optimizers, and a host of tools to make working with image and text data easier to simplify the coding necessary for writing deep neural network code.

3.1.3.1 Keras for SLAC:

As Keras has a user-oriented API, it was easy to code for the SLAC with very few lines of code. With the application of Keras, a convolutional neural network was developed for the SLAC.

It was found that Keras helped in feature-tuning and prediction of the sign-language in SLAC by providing sufficient pre-trained models.

3.1.3.2 Challenges due to Keras:

While working on the SLAC, it was found that using Keras framework is more compound and less pliable.

3.1.4 OpenCV:

OpenCV (Open Source Computer Vision Library) is alibraryofprogramming functions mainly aimed at real-timecomputer vision.Originally developed byIntel, it was later supported byWillowGarage then Itseez (which was later acquired by Intel). The library iscross-platform and free for use under theopen-sourceApache2License. Starting in 2011, OpenCV features GPU acceleration for real-time operations.

3.1.4.1 OpenCV for SLAC:

In SLAC, with the application of OpenCV library, the system was able to detect the sign gesture and pattern.

It helped the software to detect the hand sign gesture by its object identification application.

It was also found that OpenCV has a better run-time than MATLAB in the computer vision domain.

International Journal of Advances in Engineering and Management (IJAEM)
Volume 3, Issue 7 July 2021, pp: 1261-1267 www.ijaem.net ISSN: 2395-5252

It was also found that debugging was easy using the OpenCV library.

3.1.4.2 Challenges due to OpenCV:

While working on the SLAC, It was found that with a high illuminating background, the detection of sign language becomes complex.

3.1.5 Hunspell Library:

Hunspell is a library developed in C++ used to correct the errors in the spelling.

It is used to analyze different words and their parts. Hunspell library supports Chrome, Chromium, Firefox, etc. softwares. It is a free open source library license under GPL, LGPL and MPL tri-license.

3.1.5.1 Hunspell for SLAC:

Hunspell library auto-corrected the words and its letters detected by the software using the gesture recognition model.

It was found that the Hunspell library is the most effective spell checker to correct the language errors in the system.

3.1.6 Tkinter:

To develop the Graphical User Interface(GUI) in the most efficient way using python language, Tkinter is used.

3.1.6.1 Tkinter for SLAC:

While working on SLAC, Tkinter was used to create user-friendly GUI. As it has a very simple syntax, it was easy to work with its text widgets. It was also found that Tkinter can widely be ported.

3.1.6.2 Challenges due Tkinter

To make Tkinter useful, more toolkits had to be downloaded.

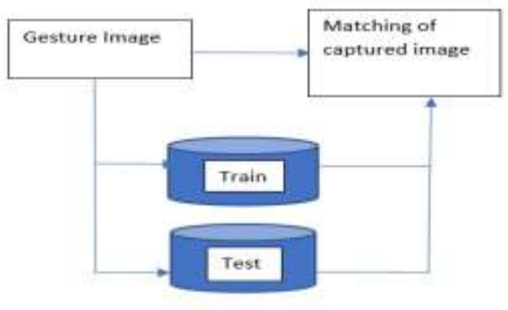
The debugging of code was a bit challenging while using Tkinter for SLAC.

Tkinter is not python-oriented completely.

3.2 Working of SLAC:

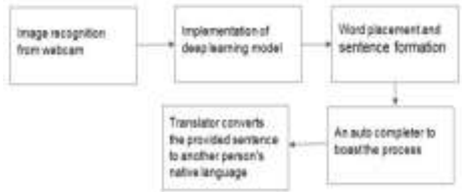
3.2.1 Overview:

The model will be implemented by a laptop with a web camera. Various signs implemented by the user will be captured by the camera and will be stored in the system. The user has to adjust according to the size of the frame so that the required gestures can be easily captured.

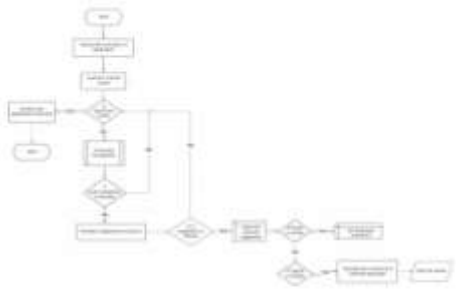


CONCEPTUAL FRAMEWORK When the camera has captured the gesture, it will compare the gesture with the stored gestures and the result will be displayed.

3.2.1.1 Application Block Diagram:

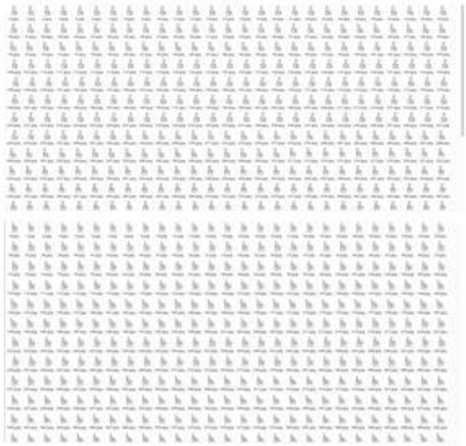


3.2.1.2 Application Flowchart:

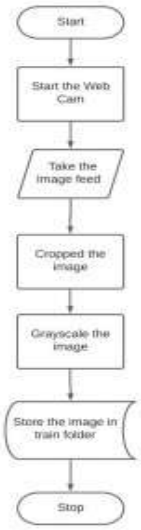


3.2.2 Image Preprocessing Module:

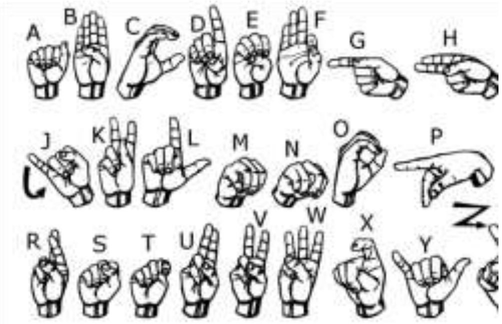
- Collection of datasets for the sign language recognition system was done by capturing the images and nearly 12000 images were captured.
- The images collected were tilted horizontally keeping in mind the left handed sign language users.
- The following are the collection of dataset of images:



Flowchart for Image Preprocessing:



There are all the classes of static alphabets right from A to Z.



The above image depicts the Hand Gesture Signs and the letter of alphabet they are associated from A to Z.

3.2.3 Data Processing and Model Building Module:

- To improve the accuracy of the model, we convert the images which are in RGB form to HSV colour space which refers to mainly: Hue, Saturation and Value.
- The HSV colour space is used because it corresponds better to how people will experience the colour than the RGB colour space does.

3.2.4 Model Training Module:

- A CNN model is used to extract features and effectively classify an image to its text.
- We have used keras and CNN which contain different layers for training the data.
- The model was trained through keras and Tensorflow Using GPU.
- The following are the steps by which the module was created:

1. The Model was compiled and created.
2. The preprocessed training and testing datasets were loaded.
3. The classifiers were trained.
4. The model was saved using json format, to make the model run on any platform. ● Flowchart for Model Training:

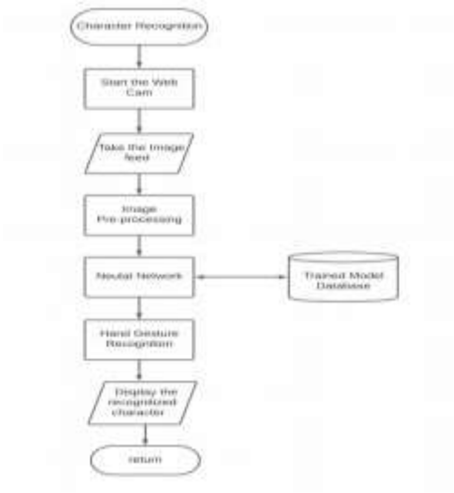
3.2.5 Video Capturing Module:

- OpenCV was used to capture the images in the 3-D array format.
- The captured images were converted into images which can be displayed on screen.

3.2.6 Character Recognition Module:

- The image was passed to the loaded model using it's predict method, which on processing the image returns the result as an array of probabilities for all 26 letters in the alphabet.
- This probability is that the current recognized character is that alphabet.
- Then a dictionary 'predictions', with keys as alphabets and values as their probabilities were created.
- 'Predictions' dictionary was sorted according to probabilities in descending order, so as to get a letter in the alphabet with the highest probability.
- Flowchart for Character Recognition Subroutine:

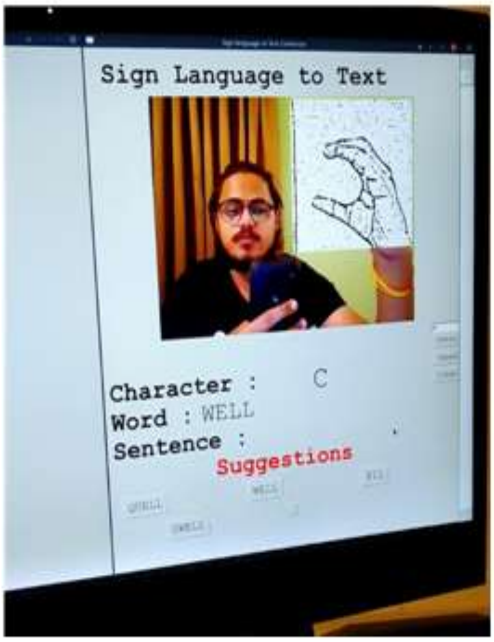
International Journal of Advances in Engineering and Management (IJAEM)
Volume 3, Issue 7 July 2021, pp: 1261-1267 www.ijaem.net ISSN: 2395-5252



3.2.7 Text-to-Speech Translation Module:

- googletrans library was used to translate the sentence to other languages.
- The audio object was generated using the 'Google Text To Speech' library to get the audio file.
- Finally the audio was played using the playsound library.

3.3 Output of the SLAC:



International Journal of Advances in Engineering and Management (IJAEM)
Volume 3, Issue 7 July 2021, pp: 1261-1267 www.ijaem.net ISSN: 2395-5252

This was the visual layout of the software.
It consisted of four sections, namely Character, Word, Sentence and Suggestions.
The user had to show the sign language gesture in the grayscale.
The system took about 5-7 seconds to predict the character, which is associated with the 4.captured sign language hand gestures section of the video feed.
The predicted character will now be displayed in the Word section.
The Hunspell library gave meaningful suggestions, which were shown in the Suggestions section.
Once the word had been formed partially, the user selected one of the suggested words, which got added to the Sentence section.
When the sentence got formed, the user5.selectedthedesiredlanguageandclicked on Speak to get the output in the form of Audio.

IV. CONCLUSION:

The purpose of developing SLAC was to fill the gap of communication between the people having speaking and hearing problems, and the regular people. This project helps us understand the importance of Computer Vision Technology and what kind of solution it offers in the real world problems scenario.

As there are more technological advancements in the coming future, this project holds a significant place as it deals with multiple domains of Computer Science.

The sensor-less sign language and gesture recognition system is a module which provides easy and satisfactory user communication for deaf and mute people. The module provides two way communications which helps in easy interaction between the normal people and the special people. The system is a novel approach to ease the difficulty in communicating with those having speech and vocal disabilities. The aim is to provide an application to the society to establish the ease of communication between the deaf and mute people by making use of image processing algorithms. Since it follows an image based approach it can be launched as an application in any minimal system and hence has near zero-cost.

V. FUTURE WORK:

A. Develop a mobile app based on this model.
B. Applying gesture recognition for accessing internet applications.
C. Enhancing the recognition capability for various lighting conditions.
D. Adding advanced widget features.
E. Using Augmented Reality application for visual screen output
F. Enhancing the sound output in the user's native accent which will make it more understandable.
G. Mobile based applications will be having AI based facial recognition which will be detecting the user's expression and giving the output in the audio-visual format.

REFERENCES:

[1]. KumudAlok, Anurag Mehra, Ankita Kaushik, AmanVerma, " Hand Sign Recognition using Convolutional Neural Network" International Research Journal of Engineering and Technology (IRJET) Jan 2020.
[2]. Lean Karlo S. Tolentino, Ronnie O. Serfa Juan, August C. Thio-ac, Maria Abigail B. Pamahoy, Joni Rose R. Forteza, and Xavier Jet O. Garcia." Static Sign Language Recognition Using Deep Learning" International
[3]. Journal of Machine Learning and Computing December 2019.

[4]. Suharjito, Ricky Anderson, Fanny Wiryana, Meita Chandra Ariesta, Gede Putra Kusuma," Sign Language
[5]. Recognition Application Systems for Deaf-Mute People: A Review Based on Input-Process-Output" 2nd
[6]. International Conference on Computer Science and Computational Intelligence 2017, ICCSCI 2017.
[7]. Ashok K Sahoo, GouriSankar Mishra and Kiran Kumar Ravulakollu, "Sign Language Recognition: State of the art" ARPN Journal of Engineering and Applied Sciences February 2014.
[8]. Shivashankara S, Srinath S," American Sign Language Recognition System: An Optimal Approach,"Modern Education and Computer Science press", August 2018.
[9]. Shaileshbachani, Shubhamdixit, Rohinchadha, Prof.AvinashBagul. "Sign Language Recognition using neural network" International Research Journal of Engineering and Technology (IRJET) Apr 2020.