International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 05 Issue: 12 | Dec - 2021 ISSN: 2582-3930

Custom Hand Gesture Recognition

Anuj Jain, Chandan Ranga and Hitesh Garg

Department of Computer Science, Maharaja Agrasen Institute of Technology

Computer Science and Engineering

Ms. Garima Gupta

ABSTRACT

Sign Language is mainly used by deaf (hard hearing) and dumb people to exchange information between their own community and with other people. It is a language where people use their hand gestures to communicate as they can't speak or hear. Sign Language Recognition (SLR) deals with recognizing the hand gestures acquisition and continues till text or speech is generated for corresponding hand gestures. Here hand gestures for sign language can be classified as static and dynamic. However, static hand gesture recognition is simpler than dynamic hand gesture recognition, but both recognition is important to the human community. We can use Deep Learning Computer Vision to recognize the hand gestures by building Deep Neural Network architectures (Convolution Neural Network Architectures) where the model will learn to recognize the hand gestures images over an epoch. Once the model Successfully recognizes the gesture the corresponding English text is generated and then text can be converted to speech. This model will be more efficient and hence communicate for the deaf (hard hearing) and dump people will be easier. In this paper, we will discuss how Sign Language Recognition is done using Deep Learning.

INTRODUCTION

Deaf (hard hearing) and dumb people use Sign Language (SL) as their primary means To express their ideas and thoughts with their own community and with other people with hand and body gestures. It has its own vocabulary, meaning, and syntax which is different from the spoken language or written language. Spoken language is a language produced by articulate sounds mapped against specific words and grammatical combinations to convey meaningful messages. Sign language uses visual hand and body gestures to convey meaningful messages. There are somewhere between 138 and 300 different types of Sign Language used around globally today. In India, there are only about 250 certified sign language interpreters for a deaf population of around 7 million. This would be a problem to teach sign language to the deaf and dumb people as there is a limited number of sign language interpreters exits today. Sign Language Recognition is an attempt to recognize these hand gestures and convert them to the corresponding text or speech. Today Computer Vision and Deep Learning have gained a lot of popularity and many State of the Art (SOTA) models can be built. Using Deep Learning algorithms and Image Processing we can able to classify these hand gestures and able to produce corresponding text. An example of “A” alphabet in sign language notion to English “A” text or speech.

American Sign Language Hand Gestures

In Deep Learning Convolution Neural Networks (CNN) is the most popular neural network algorithm which is a widely used algorithm for Image/Video tasks. For Convolution Neural Networks (CNN), we have made custom architecture where we can use this architecture to achieve the State of the Art (SOTA). By this, we can achieve an almost 100% accurate model which will recognize the hand gestures. This model will be deployed in web frameworks like Django or a standalone application or embedded devices where the hand gestures are recognized in the live camera and then converting them to text. This system will help deaf and dumb people to communicate easily and also help organization which deals in their own private gestures.

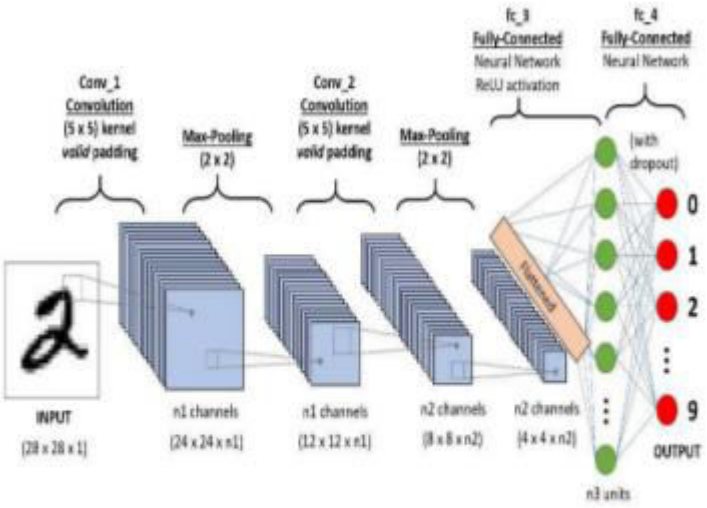


Fig. 2. Convolution Neural Networks

Features

- **Custom Gestures support.**
    1. **Create Custom Gesture –** A user will give a desired hand gesture as an input to the system with the text box available at the bottom of the screen where the user needs to type whatever he/she desires to associate that gesture with. This customize gesture will then be stored for future purposes and will be detected in the upcoming time.
    2. **Scan Single Gesture –** A gesture scanner will be available in front of the end user where the user will have to do a hand gesture. Based on Pre-Processed module output, a user shall be able to see associated label assigned for each hand gestures, based on the predefined American Sign Language (ASL) standard inside the output window screen.
- **Sentence Building Support –** A user will be able to select a delimiter and until that delimiter is encountered every scanned gesture character will be appended with the previous results forming a stream of meaning-full words and sentences.

- **Text to Speech (TtS) Support –** Whenever our application creates sentences they are saved in temporary file and whenever we access it our application recite those particular sentences.
- **Export to file –** A user would be able to export the results of the scanned character into an ASCII standard textual file format.
- **Trained on American Sign Language (ASL) –** This is an additional feature of our application which helps in recognizing the gestures of the American sign language for alphabets A to Z.

LITERATURE SURVEY

1 Real-time sign language fingerspelling recognition using convolutional neural networks from depth map.

This works focuses on static fingerspelling in American Sign Language A method for implementing a sign language to text/voice conversion system without using handheld gloves and sensors, by capturing the gesture continuously and converting them to voice. In this method, only a few images were captured for recognition. The design of a communication aid for the physically challenged.

2 Design of a communication aid for physically challenged

The system was developed under the MATLAB environment. It consists of mainly two phases via training phase and the testing phase. In the training phase, the author used feedforward neural networks. The problem here is MATLAB is not that efficient and also integrating the concurrent attributes as a whole is difficult.

3 American Sign Language Interpreter System for Deaf and Dumb Individuals .

The discussed procedures could recognize 20 out of 24 static ASL alphabets. The alphabets A, M, N, and S couldn't be recognized due to the occlusion problem. They have used only a limited number of images.

RESEARCH/APPROACH

3. IMPLEMENTATION

3.1 Dataset We have used MNIST dataset and trained our model to achieve good accuracy.

3.1.1 ASL Alphabet

The data is a collection of images of the alphabet from the American Sign Language, separated into 29 folders that represent the various classes. The training dataset consists of 45500 images which are 200x200 pixels. There are 29 classes of which 26 are English alphabets A-Z and the rest 3 classes are SPACE, DELETE, and, NOTHING. These 3 classes are very important and helpful in real-time applications.

3.1.2 Sign Language Gesture Images Dataset

The dataset consists of 29 different hand sign gestures which include A-Z alphabet gestures, and also a gesture for space which means how the deaf (hard hearing) and dumb people represent space between two letters or two words while communicating. Each gesture has 1500 images which are 200x200 pixels, so altogether there are 29 gestures which means there 55,500 images for all gestures. Convolutional Neural Network (CNN) is well suited for this dataset for model training purposes and gesture prediction.

3.2 Data Pre-processing

An image is nothing more than a 2-dimensional array of numbers or pixels which are ranging from 0 to 255.Typically, 0 means black, and 255 means white. Image is defined by mathematical function f(x,y) where ‘x’ represents horizontal and ‘y’ represents vertical in a coordinate plane. The value of f(x, y) at any point is giving the pixel value at that point of an image.

Image Pre-processing is the use of algorithms to perform operations on images. It is important to Pre-process the images before sending the images for model training. For example, all the images should have the same size of 200x200 pixels. If not,the model cannot be trained

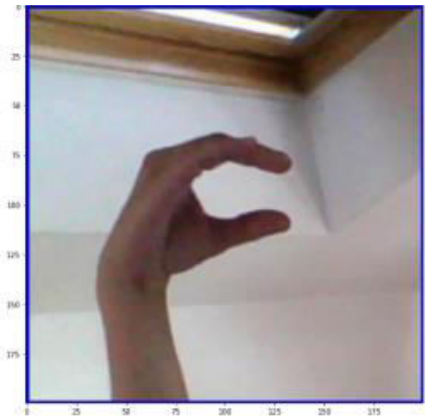


Fig. 3. Sample Image without Pre-processing

Fig. 4. Pre-Processed Image

The steps we have taken for image Pre-processing are:

- Read Images.
- Resize or reshape all the images to the same
- Remove noise.
- All the image pixels arrays are converted to 0 to 255 by dividing the image array by 255.

3.3 Convolution Neural Networks (CNN)

Computer Vision is a field of Artificial Intelligence that focuses on problems related to images andvideos. CNN combined with Computer vision is capable of performing complex problems

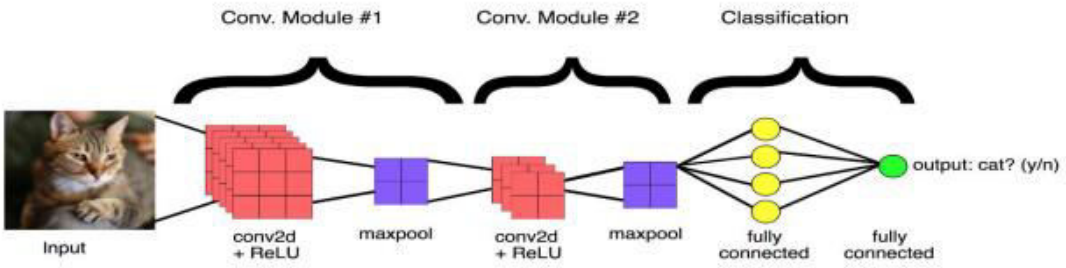


Fig. 5. Working of CNN

The Convolution Neural Networks has two main phases namely feature extraction and classification. A series of convolution and pooling operations are performed to extract the features of the image. The size of the output matrixdecreases as we keep on applying the filters. Size of new matrix = (Size of old matrix — filter size) +1 A fully connected layer in the convolution neural networks will serve as a classifier. In the last layer, the probability of the class will be predicted.

The main steps involved in convolution neural networks are:

1. Convolution

2. Pooling

3. Flatten

4. Full connection

3.3.1 Convolution

Convolution is nothing but a filter applied to an image to extract the features from it. We will use different filters to extract features like edges, highlighted patterns in an image. The filters will be randomly generated.

What this convolution does is, creates a filter of some size says 3x3 which is the default size. After creating the filter, it starts performing the element-wise multiplication starting from the top left corner of the image to the bottom right of the image.

The obtained results will be extracted feature.

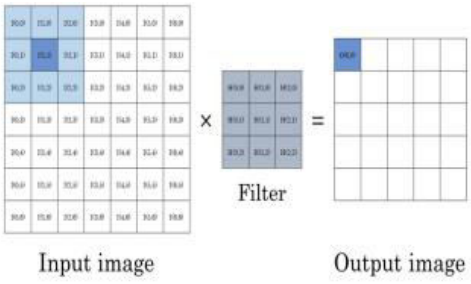


Fig. 6. Convolution

Fig. 6. Feature Extraction

3.3.2 Pooling

After the convolution operation, the pooling layer will be applied. The pooling layer is used to reduce the size of the image.

There are two types of pooling:

1. Max Pooling

2. Average Pooling

3.3.2.1 Max pooling

Max pooling is nothing but selecting the maximum pixel value from the matrix

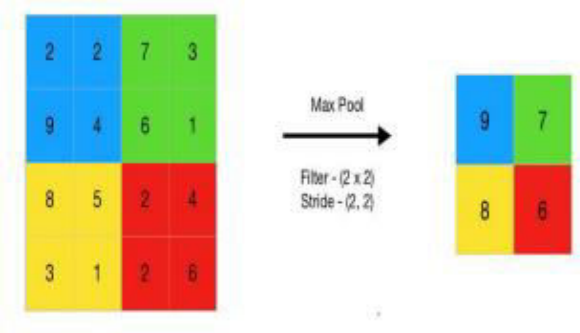


Fig. 7. Max Pooling

This method is helpful to extract the features with high importance or which are highlighted in the image.

3.3.3 Flatten

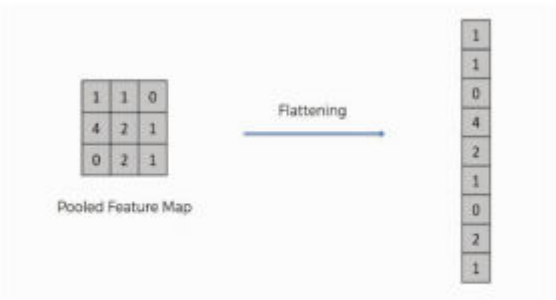


Fig. 9. Flatten

The obtained resultant matrix will be in muti-dimension. Flattening is converting the data into a 1-dimensional array for inputting the layer to the next layer. We flatten the convolution layers to create a single feature vector.

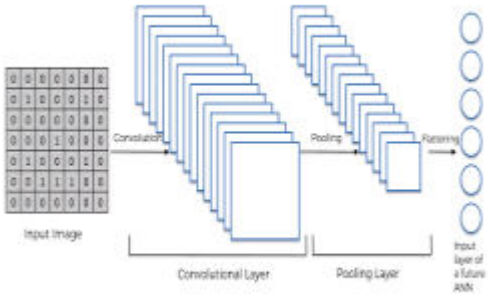


Fig. 10. Full Connection

A fully connected layer is simply a feed-forward neural network. All the operations will be performed and prediction is obtained. Based on the ground truth the loss will be calculated and weights are updated using gradient descent backpropagation algorithm.

PROPOSED ARCHITECTURE

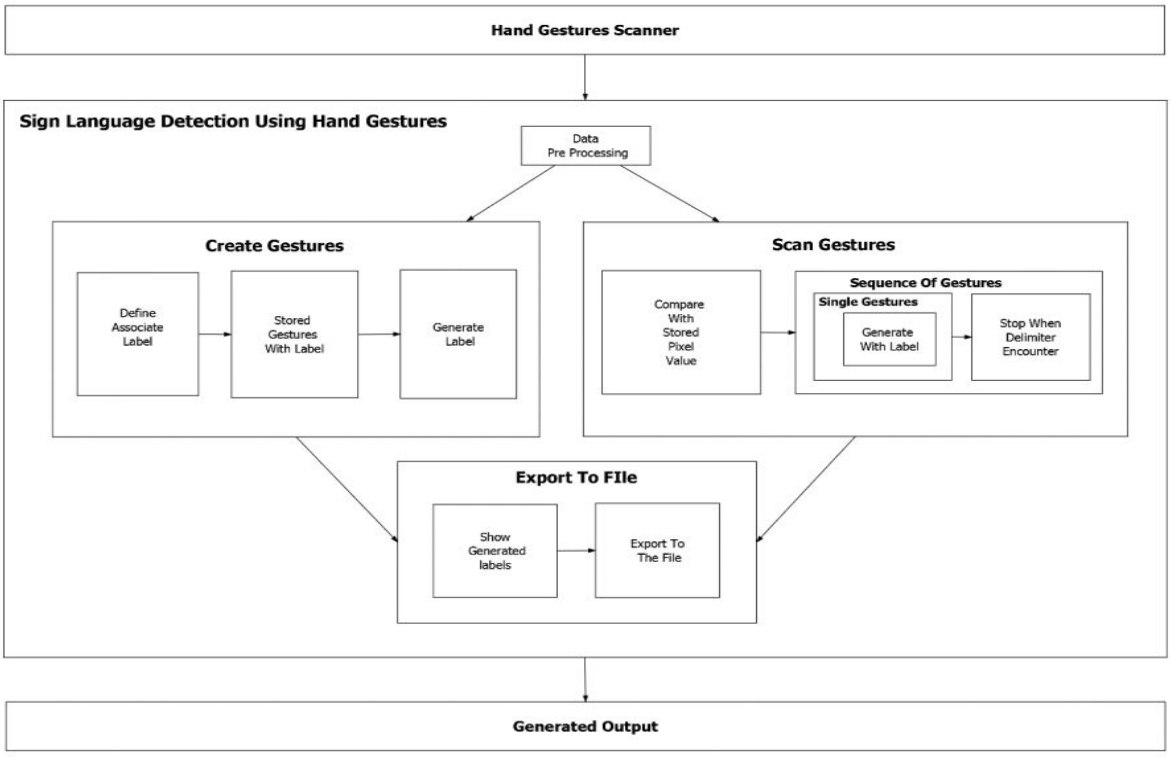
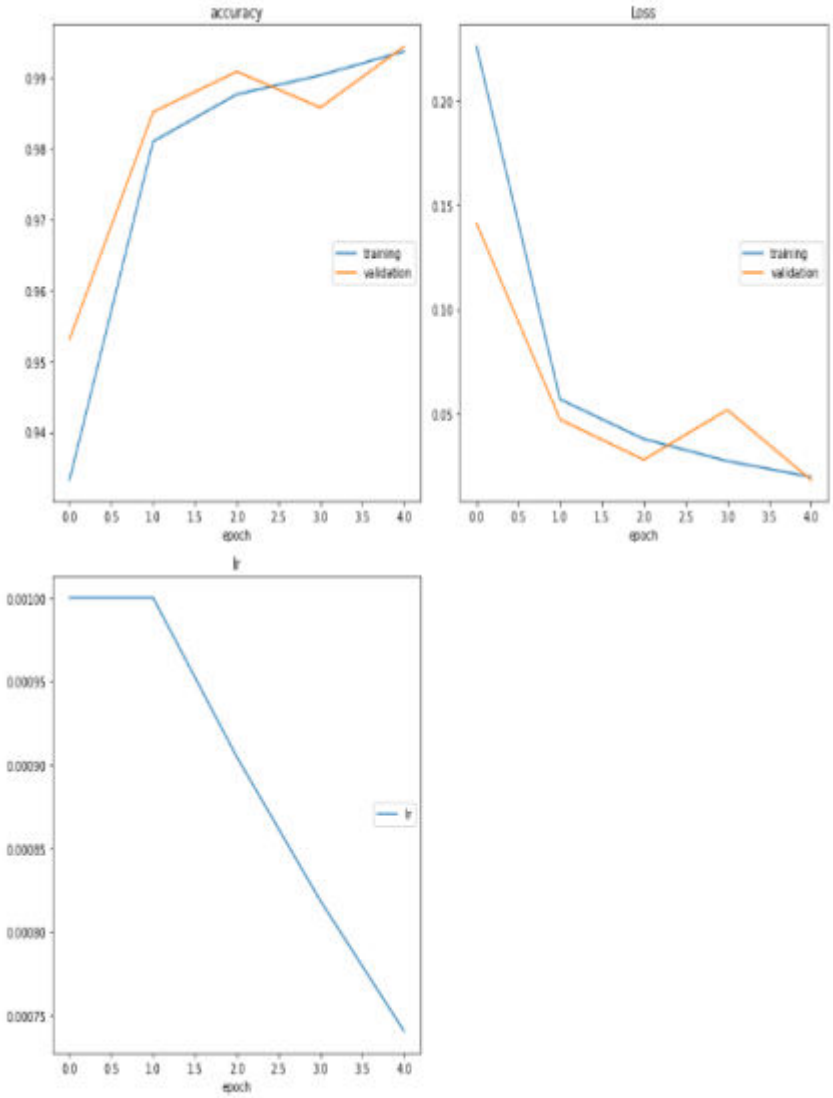


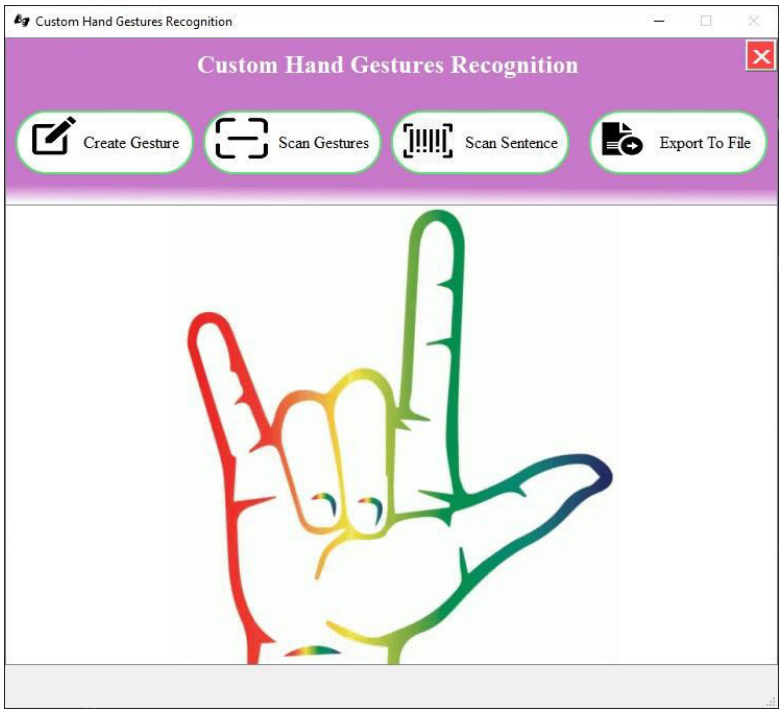
Fig 1: System Architecture for Sign Language Recognition Using Hand Gestures.

RESULT

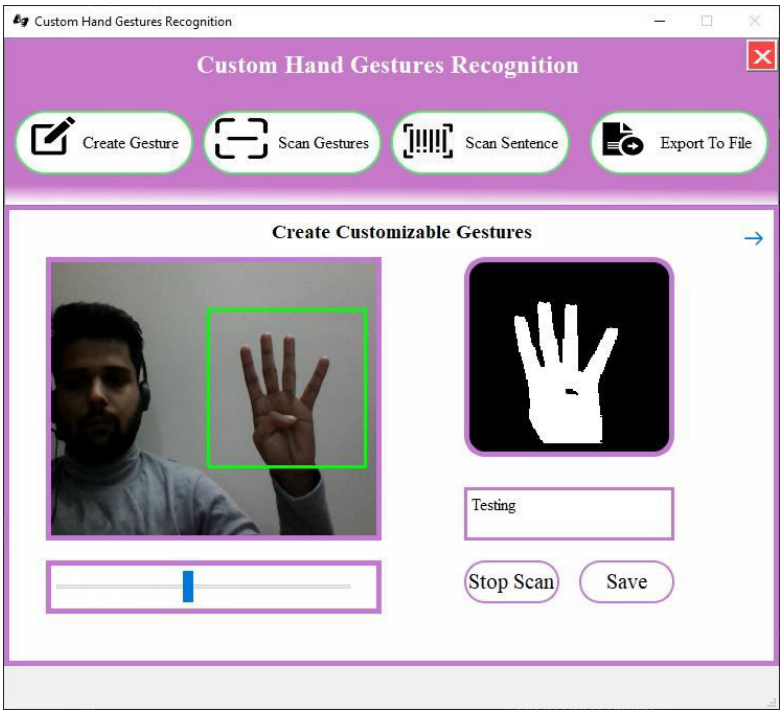


- We have trained our model for around 25 epochs with a batch size of 32.
- Our model performed well on the test cases. We have achieved almost 99% accuracy.
- We have used OpenCV to test our results in the live camera. This is a sample result on a live camera

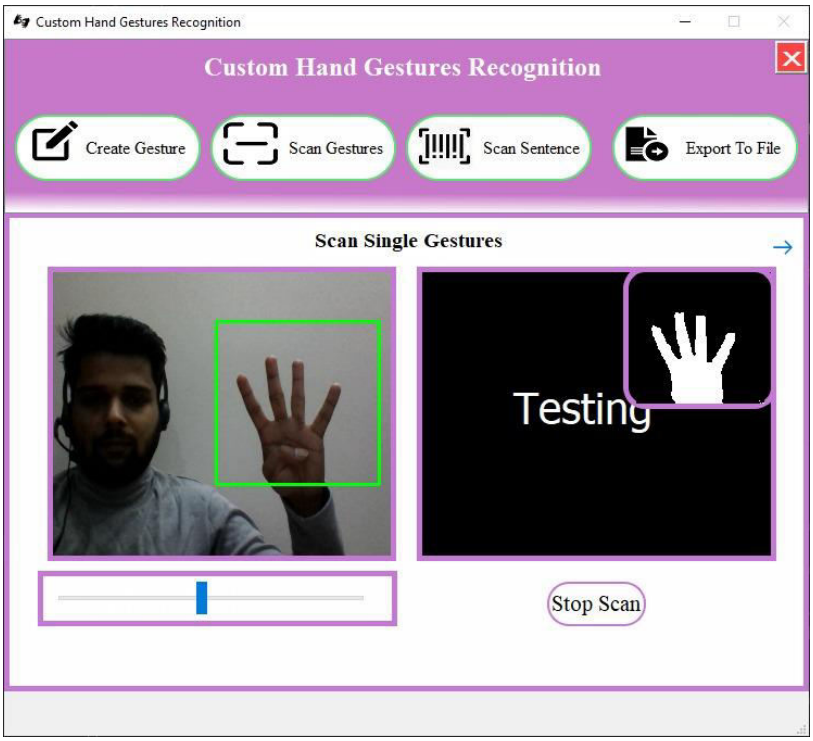
Snapshots



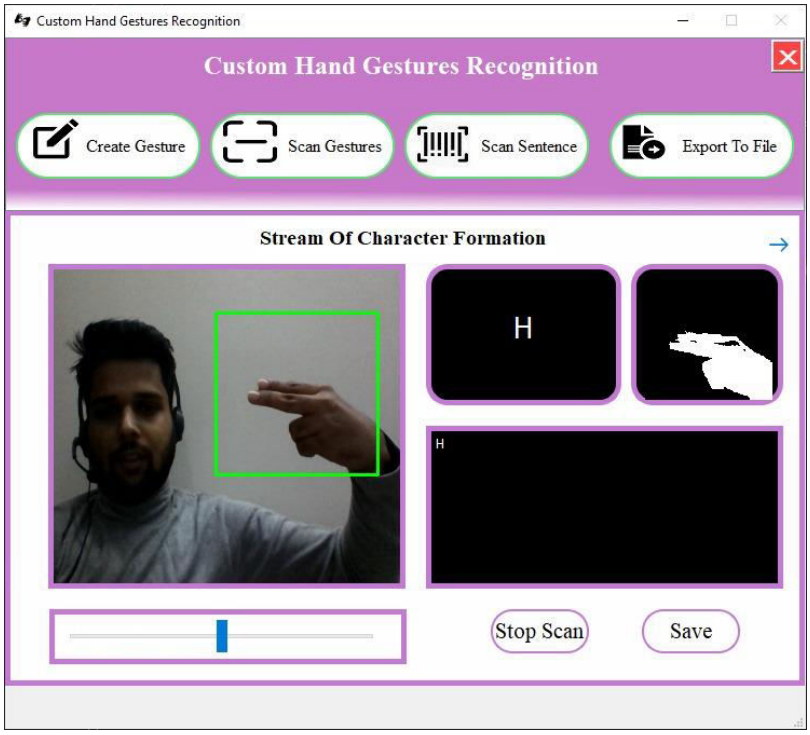
Opening Window



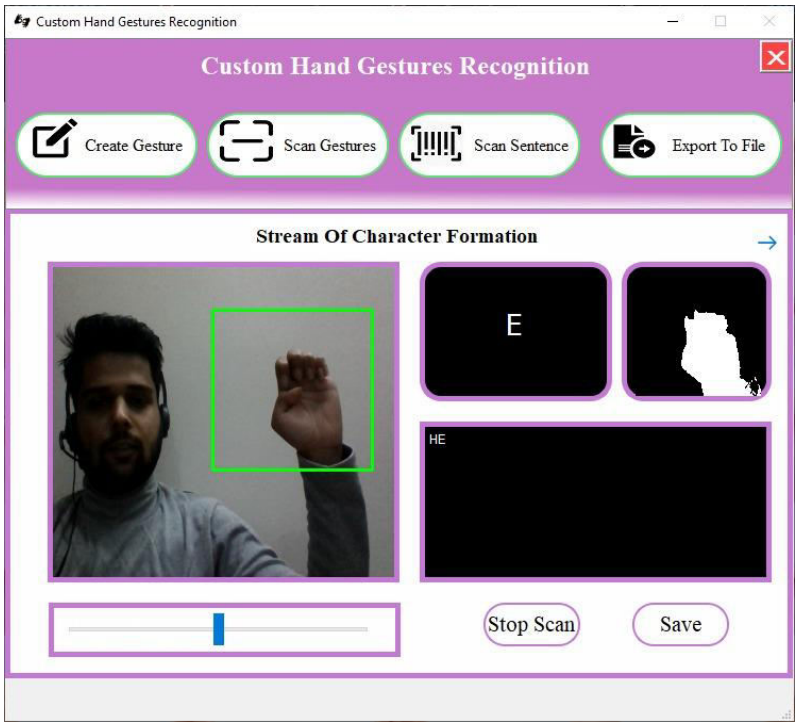
Creating Custom Gesture named "Testing"



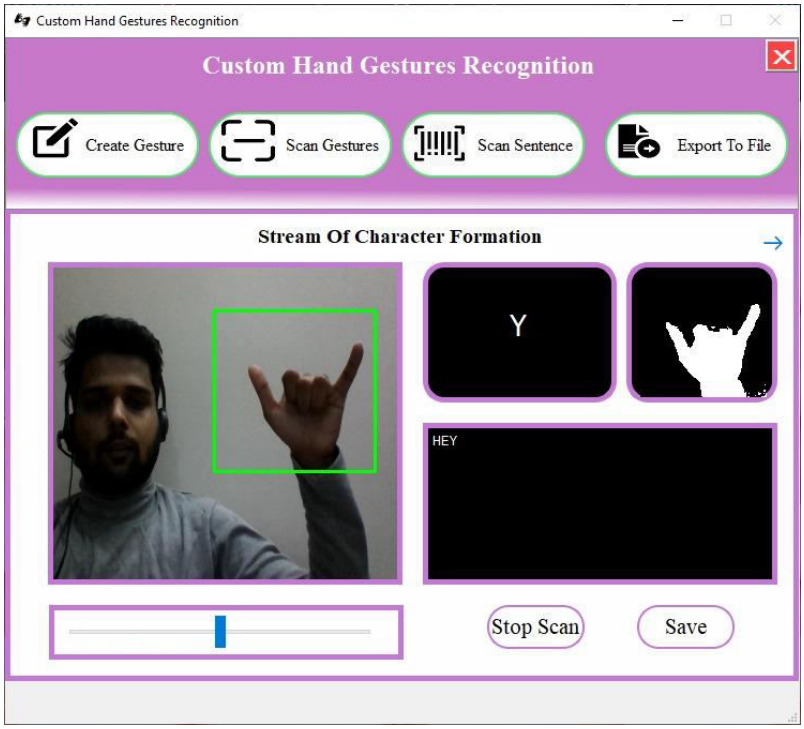
Scanning a custom gesture named "Testing"



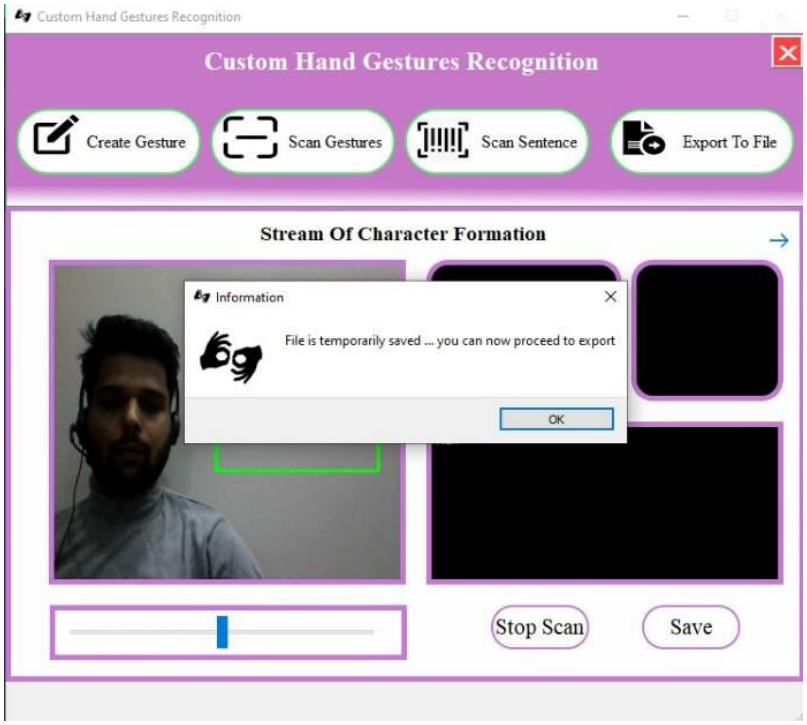
Scanning first letter of a sentence "HEY"



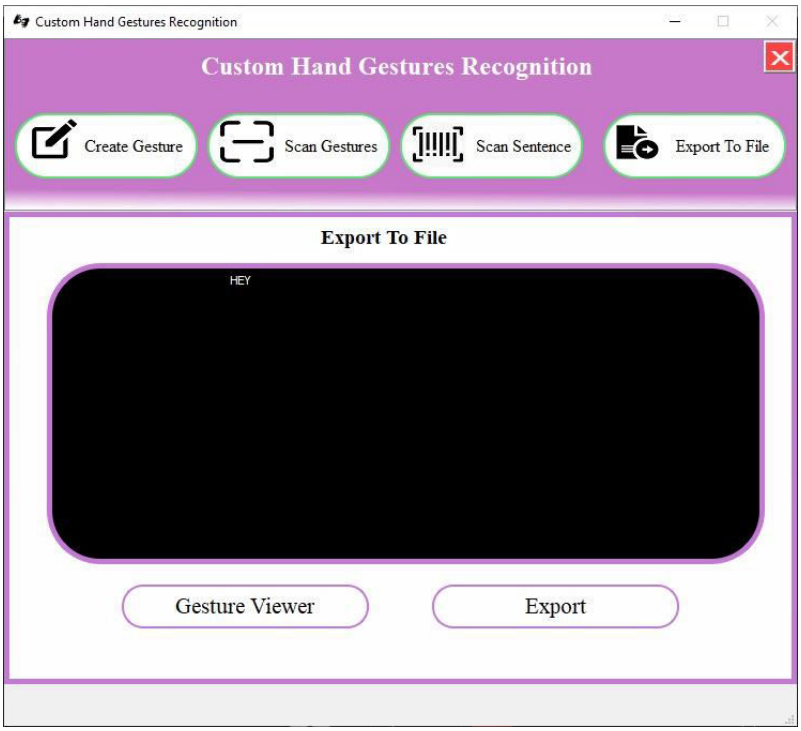
Scanning second letter of a sentence "HEY"



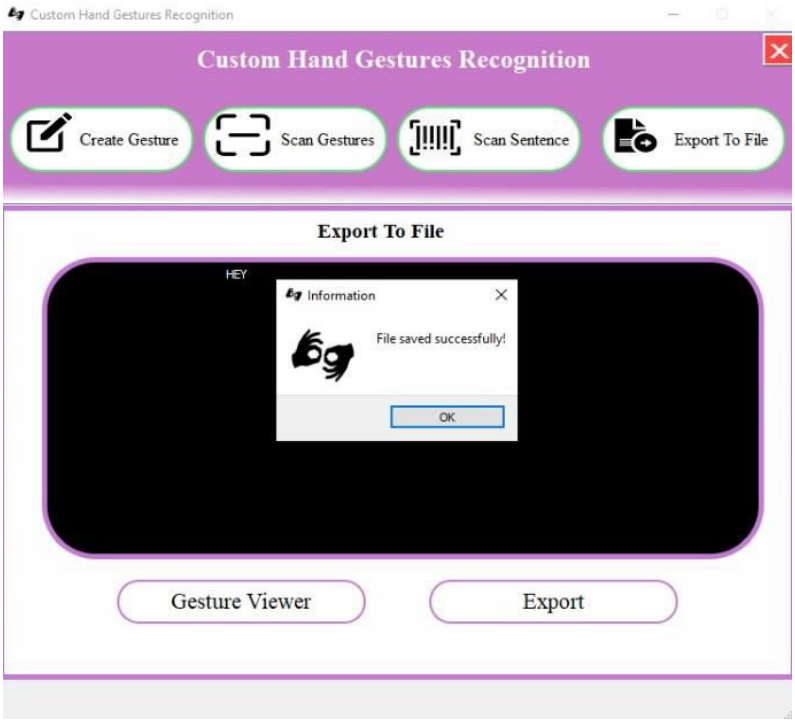
Scanning Third letter of sentence "HEY"



File is temporarily saved



Gestures used to made the sentence can be viewed



Sentence can be saved permanently into a file

CONCLUSION

From this project/application we have tried to overshadow some of the major problems faced by the disabled persons in terms of talking. We found out the root cause of why they can't express more freely. The result that we got was the other side of the audience are not able to interpret what these persons are trying to say or what is the message that they want to convey.

Thereby this application serves the person who wants to learn and talk in sign languages. With this application a person will quickly adapt various gestures and their meaning as per ASL standards. They can quickly learn what alphabet is assigned to which gesture. Add-on to this custom gesture facility is also provided along with sentence formation. A user need not be a literate person if they know the action of the gesture, they can quickly form the gesture and appropriate assigned character will be shown onto the screen.

Concerning to the implementation, we have used TensorFlow framework, with keras API. And for the user feasibility complete front-end is designed using PyQT5. Appropriate user-friendly messages are prompted as per the user actions along with what gesture means which character window. Additionally, an export to file module is also provided with TTS(Text-To-Speech) assistance meaning whatever the sentence was formed a user will be able to listen to it and then quickly export along with observing what gesture he/she made during the sentence formation.

FUTURE SCOPE

- It should be capable of converting normal language to sign language.

- It can be integrated with various search engines and texting application such as google, WhatsApp. So that even the illiterate people could be able to chat with other persons, or query something from web just with the help of gesture.
- This project is working on image currently, further development can lead to detecting the motion of video sequence and assigning it to a meaningful sentence with TTS assistance.

REFERENCES:

- https://www.researchgate.net/publication/354066737_Sign_Language_Recognition : To get our base idea and to do improvements.
- Brill R. 1986. The Conference of Educational Administrators Serving the Deaf: A History. Washington, DC: Gallaudet University Press.
- https://www.github.com : To get help and suggestion with our bugs and problems.
- https://www.wikipedia.com : To learn required things.
- https://www.techpedia.com : To learn required things.
- https://edu.authorcafe.com/ : To get project different methods.
- Brill R. 1986. The Conference of Educational Administrators Serving the Deaf: History. Washington, DC: Gallaudet University Press.
- Y. Lecun, L. Bottou, Y. Bengio and P. Haffner, "Gradient-based learning applied to document recognition," in Proceedings of the IEEE, vol. 86, no. 11, pp. 2278-2324, Nov. 1998, doi: 10.1109/5.726791.

- M. Sandler, A. Howard, M. Zhu, A. Zhmoginov and L. Chen, "MobileNetV2: Inverted Residuals and Linear Bottlenecks," 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018, pp. 4510-4520, doi: 10.1109/CVPR.2018.00474.

- L. K. Hansen and P. Salamon, "Neural network ensembles," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 12, no. 10, pp. 993-1001, Oct. 1990, doi: 10.1109/34.58871.

- Kang, Byeongkeun, Subarna Tripathi, and Truong Q. Nguyen. ”Real- time sign language fingerspelling recognition using convolutional neural networks from depth map.” arXiv preprint arXiv: 1509.03001 (2015).

- Suganya, R., and T. Meeradevi. ”Design of a communication aid for phys- ically challenged.” In Electronics and Communication Systems (ICECS), 2015 2nd International Conference on, pp. 818-822. IEEE, 2015.

- Sruthi Upendran, Thamizharasi. A,” American Sign Language Interpreter System for Deaf and Dumb Individuals”, 2014 International Conference on Control, Instrumentation, Communication and Computa.

- David H. Wolpert, Stacked generalization, Neural Networks, Volume 5, Issue 2, 1992, Pages 241-259, ISSN 0893-6080, https://doi.org/10.1016/S0893-6080(05)80023-1.

- Y. Liu, X. Yao, Ensemble learning via negative correlation, Neural Networks, Volume

12, Issue 10,1999, Pages 1399-1404, ISSN 0893-6080, https://doi.org/10.1016/S0893-6080(99)00073-8.

- Polikar R. (2012) Ensemble Learning. In: Zhang C., Ma Y. (eds) Ensemble Machine Learning. Springer, Boston, MA. https://doi.org/10.1007/978-1-4419-9326-7_1\

- MacKay D.J.C. (1995) Developments in Probabilistic Modelling with Neural Networks
  - Ensemble Learning. In: Kappen B., Gielen S. (eds) Neural Networks: Artificial Intelligence and Industrial Applications. Springer, London. https://doi.org/10.1007/978-1-4471-3087-1_37