SIGN LANGUAGE RECOGNITION & SPEECH CONVERSION

Md.Thousif Ahmed, Kartik Reddy, Rohit Reddy,Karthik and Tanmoy Hazra
Indian Institute of Information Technology, pune
{thousifahmed, karthikreddy}19@ece.iiitp.ac.in
{rohitreddy, kommalapatikarthik}19@cse.iiitp.ac.in
tanmoyhazra@iiitp.ac.in

Abstract. Inability to speak is considered a disability. People suffering with this disability often use different methods to communicate with other people. A number of methods are available for their communication, out of which the most common method of communication is through sign language. Developing sign language applications for deaf people can be very vital, as they will be able to communicate easily with even those who cannot understand sign language. According to the NAD, 18 million people are estimated to be deaf in India. They can communicate among themselves using sign language but having a conversation with others is a difficult task. Extensive work has been done on the American Sign Language(ASL) and by working on this project we want to translate the sign language into text and sound and enable these people to have an improved conversational experience. This project aims to bridge the communication gap between people and differently-abled people using sign language. The image dataset consists of 32 ASL gestures. Our model obtained an accuracy of 99.64% using Convolution Neural Network.

Keywords: ASL, Sign Language Character Recognition, Convolution Neural Network, Computer Vision, Machine Learning

1 Introduction

Sadly, in the fast changing society we live in, people with hearing impairment are usually forgotten and left out. They have to struggle to bring up their ideas, voice out their opinions and express themselves to people who are different to them. Sign language, although being a medium of communication to deaf people, still has no meaning when conveyed to a non-sign language user.So, we are putting forward a sign language recognition system. It will be an ultimate tool for people with hearing disability to communicate their thoughts as well as a very good interpretation for non sign language users to understand what the latter is saying.

1.1 Deep Learning

To know about Deep learning we need to first understand about artificial intelligence. Artificial intelligence (AI) is a very vast branch in Computer Science and it is used for constructing smart machines. The smart machines are capable of executing a task that requires human intelligence. Deep learning and machine learning are learning techniques in Artificial Intelligence. We have achieved many advancements in machine learning and deep learning. Now , as we are discussing Deep Learning. Deep learning is an AI function that works almost similar to the human brain in terms of performing tasks such as processing data and creating patterns and in the terms of decision making. Deep learning is a part of machine learning in artificial intelligence which consists of networks such as deep neural networks which are capable of learning unsupervised data.Consider a human brain: it consists of millions of neurons. A deep neural network (DNN) has several layers in between input and output layers. Each layer consists of several nodes and are similar to neurons of the human brain. To solve a task the system has to process layers of data between input and output. Stimulus must be passed at the input layer in order to execute a task.

1.2 Importance of Deep Learning

- Deep learning has a very significant importance in terms of handling big data.
- Deep learning includes a large number of features and its corresponding process.
- However , the access of the vast amount of data is required for deep learning algorithms to be effective.
- But deep learning models will be overfitted if data is too simple or incomplete.
- Deep learning is very useful for real life applications because of its algorithm's potential at learning.

2 Literature Survey

Identification of sign gestures is mainly performed by the following methods:

- **The Glove-based** method is in which the signer has to wear a hardware glove, while the hand movements are getting captured.
- **Vision-based method**, further classified into static and dynamic recognition. Statics deals with the detection of static gestures(2d-images) while dynamic is a real-time live capture of the gestures. This involves the use of the camera for capturing movements.

The Glove-based method, seems a bit uncomfortable for practical use, despite having an accuracy of over 90%.

Table 2; Literature Survey

s.no	Name of author(s)	Title	Content	Publish address	Publish year
1	K.Bantupalli and Y.Xie	American Sign Language Recognition using Deep Learning and Computer Vision	American Sign Language using Deep Learning and Computer Vision ● Consists of 31 American sign gestures. ● Inception(CNN) is used for Feature extraction. ● RNN is used to train on temporal features.	IEEE ACCESS	2018
2	Pigou	Gesture and Sign Language Recognition with Temporal Residual Networks	● Data set used CLAP 14. ● Consists of 20 Italian sign gestures. ● After preprocessing the CNN model having 6 layers is used for training. ● CNN is used for Feature Extraction while classification uses ANN. ● Achieved accuracy - 91.70% , Error rate - 8.30%.	ICCV-CVF + iEEE ACCESS	2019

3	HCM Herath et al	Image based Sign Language Recognition System for Sinhala Sign language.	Image based Sign Language Recognition System for Sinhala Sign language. ● Low cost approach - using green screen in the background for preprocessing as subtraction would be easy. ● Mapping signs using centroid method.	Springer Science+Business Media, LLC, part of Springer Nature 2019	2019
4	M. K. Ahuja et al	Hand Gesture Recognition Using PCA	The authors proposed a scheme using a database-driven hand gesture recognition based on the skin color model approach and thresholding approach and an effective template matching using PCA.	IEEE ACCESS + MITE 2016	2016
5	Siming He et al.	Hand Gesture Recognition System For Dumb People .	Faster R-CNN with an embedded RPN module is used to improve accuracy, A 3D CNN is used for feature extraction.	The Author(s) 2017	2017

3 Methodology

The flow chart below explains about our project.

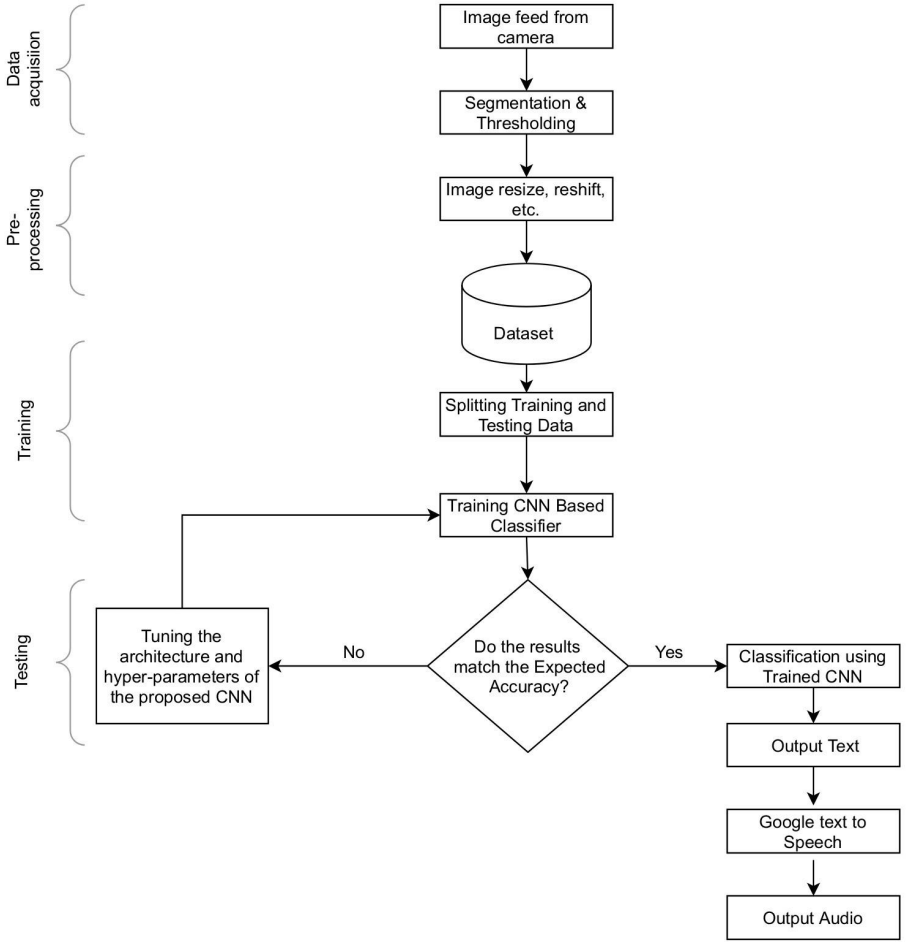


Figure 3 Flow Chart

3.1 Dataset Acquisition

We have created our own Dataset by following the American Sign Language for alphabet, special characters and also added a few custom gestures like YOLO , All the best, etc.. which can be further extended to add Indian Sign Language Vocabulary. We first segment the Hand region by using Background Subtraction Technique which uses the concept of running averages from a sequence of video frames. As the name suggests it is a way of eliminating the background from the image. To carry out this we extract the moving foreground from the static background. It has various use cases in day-to--day life, like object segmentation, counting the number of visitors, number of vehicles in traffic etc. It is able to learn and identify the foreground mask. That is why this technique is useful and effective for our project. Then we do thresholding for this image. By this we can get the contour of the hand region.

Fig 3.1 Dataset

3.2 Data Preprocessing

Then we do thresholding for this image. By this, we can get the contour of the hand region. And save these gesture images separately in the directory with labels from 0 to 31. And after taking images of all the gestures we perform the data preprocessing using Keras preprocessing model. Here we are using Keras preprocessing model because it can happen that images in our dataset are all right handed so,some users who are left handed will use their left hand for communicating and at that time our model may not be able to recognise them so, to avoid this we are using keras model because it does the necessary resizing, shifting, flipping, orientations to certain degree, etc..which concludes our Final Dataset. We split the datasets into two parts. 70% for training data and 30% for testing data. And the next step is the feature extraction & classification,

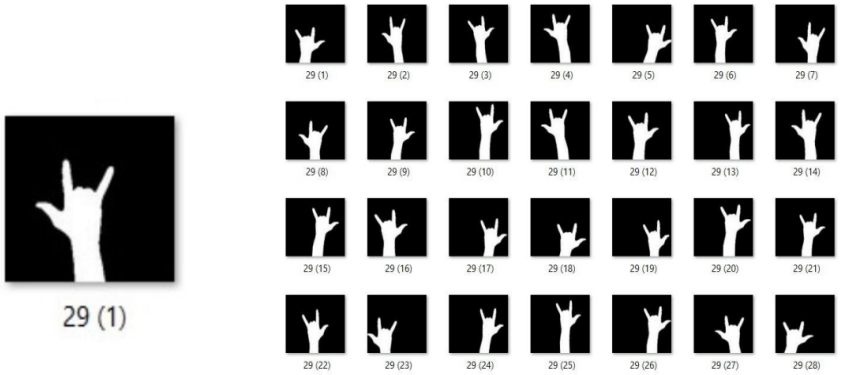


Fig 3.2 Data Preprocessing

3.3 Feature Extraction

3.3.1 Convolutional Neural Networks

As the name says this model uses Convolution in at least one of it's layers. Here in the first part, we use convolution operations for feature extraction. In this feature extraction, we extract important points or features from the image known as Filters.

Pooling: Pooling layers decrease the dimensions of the data by combining the outputs of neuron clusters at one layer into a single neuron in the next layer. It also helps with the position invariant feature detection which means, in the below image no matter where eyes, ears, nose, etc. are present, it will detect them. It also makes the model tolerant towards small distortions.

ReLU: It speeds the training of the model. It introduces non-linearity to the convolution network(CNN)

Padding: Padding means giving additional pixels at the boundary of the data.

Flattening: In this process, we convert the data into a one-dimensional array to give it to the final classification model.

Fully-connected layer: The last layer is a fully-connected layer Its purpose is to use features from previous layers for classifying the input image into various classes based on training data. By combining all these layers we create a CNN model

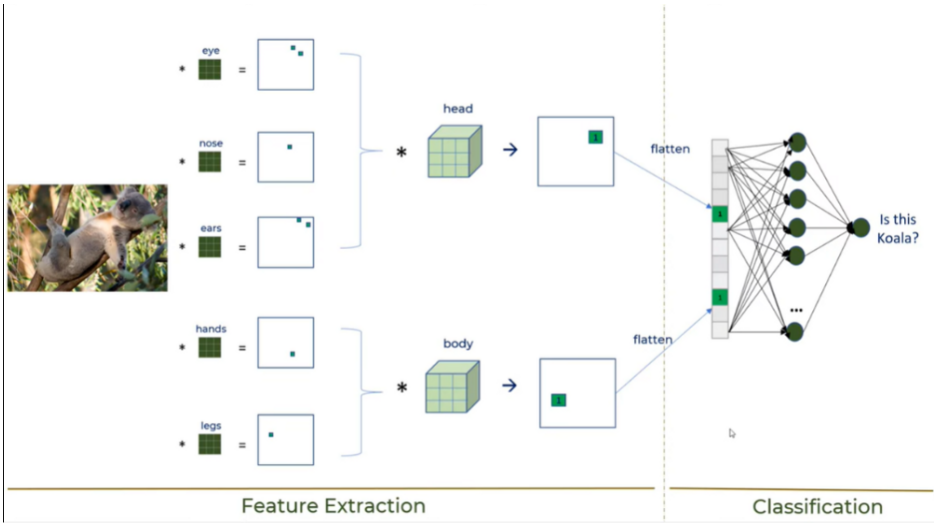


Fig 3.3.1 CNN

3.3.2 Principal Components Analysis

Principal Component Analysis, or PCA, is a dimensionality reduction technique that is often used to decrease the dimensionality of large data sets, by converting a large set of variables into a smaller set of variables without losing much information. After performing the preprocessing, Training of the model is performed using PCA. This is mainly achieved by calculating the Covariance matrix and extracting the eigenvector and eigenvalues from the Covariance matrix. The reduced transformation matrix is then calculated while projecting to the PCA space. The mean of all the images in the training dataset is computed to get the reduced transformation matrix. The mean image is subtracted from all the images in the training dataset for calculating the covariance matrix. The covariance matrix is computed from the training dataset, which is used to calculate the eigenvectors and eigenvalues. The eigenvectors and eigenvalues of the dataset images are arranged according to the index in decreasing order. The reduced transformation matrix is calculated and is then used to project the training dataset into the PCA space.

3.4 Model Architecture

3.4.1 CNN

The model architecture includes the Deep Convolutional Neural Network. The network consists of an input layer, followed by 7 convolutional and max-pooling layers alternately and followed by a soft max fully connected output layer to extract features. After the feature extraction, 2 layers of hidden neural networks are used for classification. Dropouts can help to avoid overfitting. We kind of saw that 50 iterations kind of trains the model well and there is no increase in validation accuracy along the lines so that should be enough. CNN belongs to the category of artificial neural networks usually designed to extract features from data and to classify given high-dimensional data. Pre-processed images are provided as input to the CNN model for training. Cross-validation method is used to arrive at the best model.The model achieves an accuracy of 99.64% on the validation dataset and the ratio of training set to validation set is around 210 : 90.

3.4.2 Support Vector Machine

"Support Vector Machine" (SVM) is a supervised machine learning algorithm that can be used for both classification or regression challenges. However, it is mostly used in classification problems. In the SVM algorithm, we plot each data item as a point in n-dimensional space (where n is a number of features you have) with the value of each feature being the value of a particular coordinate. Then, we perform classification by finding the hyper-plane that differentiates the two classes very well. If the given data is linear we can classify it simply by using line as a hyperplane, but if the data is non-linear we need to add suitable features i.e, add a new dimension. In the SVM classifier, it is easy to have a linear hyper-plane between these two classes. But, another burning question which arises is, should we need to add this feature manually to have a hyper-plane. No, the SVM algorithm has a technique called the kernel trick. The SVM kernel is a function that takes low dimensional input space and transforms it to a higher dimensional space i.e. it converts non separable problem to separable problem. It is mostly useful in non-linear separation problems. Simply put, it does some extremely complex data transformations, then finds out the process to separate the data based on the labels or outputs you've defined.

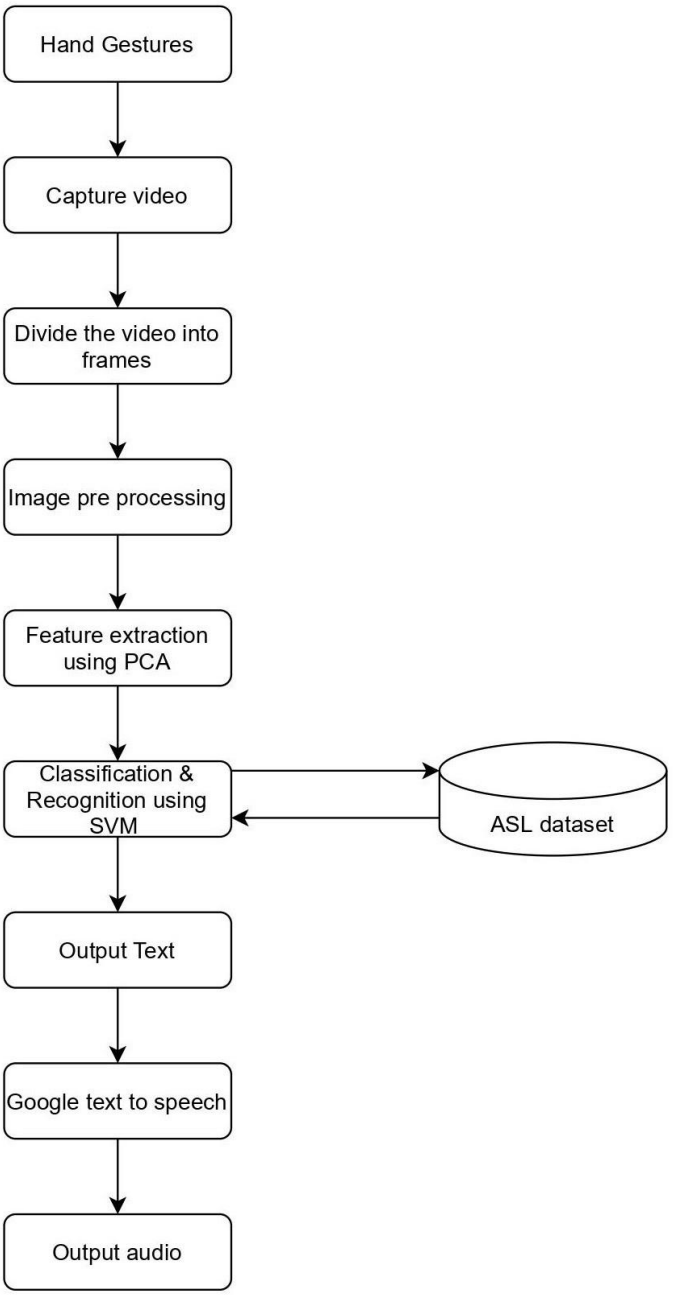


Fig 3.4.2 Flow chart

3.5 Real Time Prediction

The model takes a live sequence of video frames from the webcam, it fixes the background of our roi, extracts the hand region, and finally outputs a threshold image. This threshold image is given to our trained model as an input. The image is processed and our model gives two outputs - predictedClass , confidence.We take the maximum confidence class as the predictedClass. As we have created 32 distinct gestures so the model contains 32 predictedClasses consisting of alphabets, special characters and custom gestures.For improved communication, we have added the text-mode which makes it simple to convey messages between people using signs. We have also added the text to sound feature to make conversations easier.

3.6 Voice Conversion

After predicting the hand gestures the text will be converted to speech using Google Text-to-Speech. gTTS (Google Text-to-Speech)is a Python library and CLI tool to interface with Google Translate text-to-speech API. We will import the gTTS library from the gtts module which can be used for speech translation. The text variable is a string used to store the user's input.

4 Experiments and Results

4.1 Dataset Description;

We have created our own Dataset by following the American Sign Language for alphabet, special characters and also added a few custom gestures like YOLO , All the best, etc.. which can be further extended to add Indian Sign Language Vocabulary. We first segment the Hand region by using Background Subtraction Technique which uses the concept of running averages from a sequence of video frames.Then we do thresholding for this image. By this we can get the contour of the hand region.For each gesture we get a threshold image and all these images are pre-processed using Keras preprocessing model which does the necessary resizing, shifts, flips, orientations to a certain degree, etc.. which concludes our Final Dataset.

Figure 4.1 Dataset images

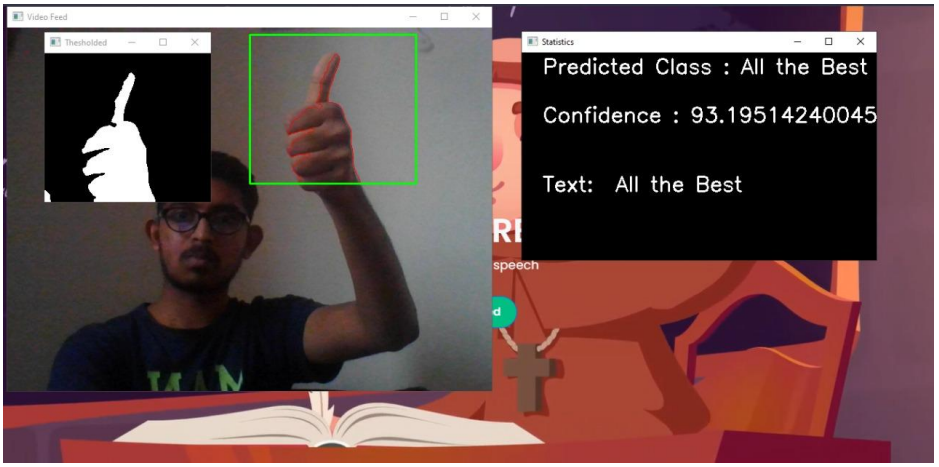
4.2 Parameter Setup

Training parameter	CNN
Learning Rate	0.001
Batch Size	64
Optimizer	Adam
Loss Function	Categorical cross entropy
Epochs	50

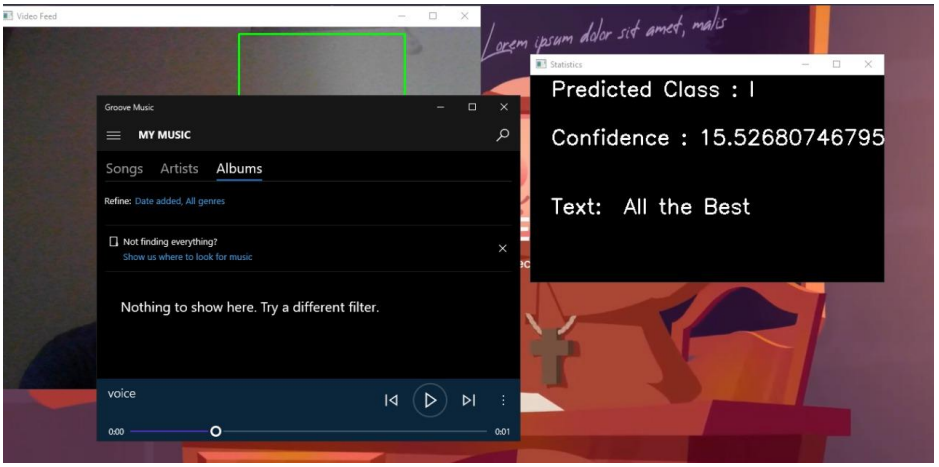
Table 4.2 "CNN Parameter Setup"

4.3 Results

Fig 4.3(1) Web page



4.3(2) Sign Language Prediction



4.3(3) Speech Conversion

4.3.1 Accuracy

Models	Training Accuracy	Testing Accuracy
CNN	99.93	99.64
SVM	91.57	84.18

Table 4.3.1 Accuracy

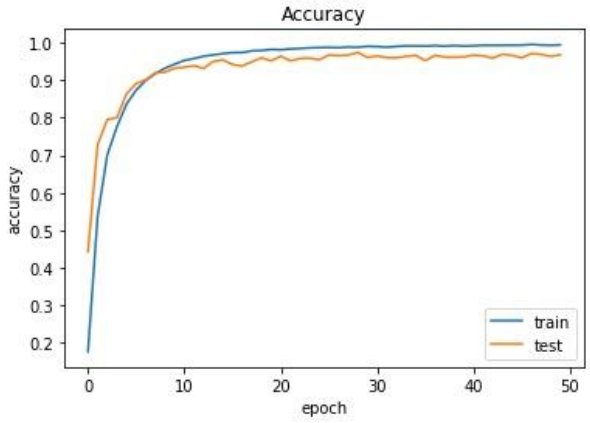


Fig 4.3.1 CNN Accuracy

5 Conclusion and Future Scope

Some of the limitations of our model are the model requires good lighting conditions and as background subtraction technique was used so we need to make sure that the background is stable otherwise the prediction may go wrong. As we are limited to make only a few sets of signs because of using our hands so maybe we cannot cover all ranges of signs. As we use text mode the size of the text is limited so we cannot form large messages at once. So we can improve the model in future to extend its effectiveness further and eliminate the limitations.

6 Acknowledgments

This project would not have been possible without the help and cooperation of many.I would like to thank the people who helped me directly and indirectly in the completion of this project work.First and foremost, I would like to express my gratitude to our beloved director, Dr.Anupam Shukla, for providing his kind support in various aspects. I would like to express my gratitude to my project guide Dr.Tanmoy Hazra, Assistant Professor, Head of the Department of CSE, for providing excellent guidance, encouragement,inspiration, constant and timely support throughout this B.Tech project. I would also like to thank all the faculty members in the Dept. of CSE and my classmates for their steadfast and strong support and engagement with this project.

References

[1] Sunitha K. A, Anitha Saraswathi.P, Aarthi.M, Jayapriya. K, Lingam Sunny, “Deaf Mute Communication Interpreter- A Review”, International Journal of Applied Engineering Research ,Volume 11, pp 290-296 ,2016.

[2] Madhavan Suresh Anand, Nagarajan Mohan Kumar, Angappan Kumaresan, “ An Efficient Framework for Indian SignLanguage Recognition Using Wavelet Transform” Circuits and Systems, Volume 7, pp 1874-1883, 2016.

[3] Mandeep Kaur Ahuja, Amardeep Singh, “Hand Gesture Recognition Using PCA”, International Journal of Computer Science Engineering and Technology (IJCSET), Volume 5, Issue 7, pp. 267-27, July 2015.

[4] Sagar P.More, Prof. Abdul Sattar, “Hand gesture recognition system for dumb people”,

[5] International Journal of Science and Research (IJSR)

[6] Chandandeep Kaur, Nivit Gill, “An Automated System for Indian Sign Language Recognition”,International Journal of Advanced Research in Computer Science and Software Engineering.

[7] Pratibha Pandey, Vinay Jain, “Hand Gesture Recognition for Sign Language Recognition: A Review”,International Journal of Science, Engineering and Technology Research (IJSETR), Volume 4, Issue 3,March 2015 .

[8] Nakul Nagpal,Dr. Arun Mitra.,Dr. Pankaj Agrawal, “Design Issue and Proposed Implementation of Communication Aid for Deaf & Dumb People”, International Journal on Recent and Innovation Trends in Computing and Communication ,Volume: 3 Issue: 5,pp- 147 – 149.

[9] Neelam K. Gilorkar, Manisha M. Ingle, “Real Time Detection And Recognition Of Indian And American Sign Language Using Sift”, International Journal of Electronics and Communication Engineering & Technology (IJECET), Volume 5, Issue 5, pp. 11-18 , May 2014

[10] Neelam K. Gilorkar, Manisha M. Ingle, “A Review on Feature Extraction for Indian and American Sign Language”, International Journal of Computer Science and Information Technologies, Volume 5 (1) , pp- 314-318, 2014.

[11] Ashish Sethi, Hemanth ,Kuldeep Kumar,Bhaskara Rao ,Krishnan R, “Sign Pro-An Application Suite for Deaf and Dumb”, IJCSET , Volume 2, Issue 5, pp-1203-1206, May 2012.

[12] Priyanka Sharma,“Offline Signature Verification Using Surf Feature Extraction and Neural Networks Approach”, International Journal of Computer Science and Information Technologies, Volume 5 (3) , pp 3539-3541, 2014.

[13] F. K. H.Quek, “Toward a Vision-Based Hand Gesture Interface,” in Virtual Reality Software and Technology Conference, (1994) 17-31.

[14] F. K. H. Quek, “Eyes in the Interface,” Image and Vision Computing, 13 (1995).

[15] R. A. Bolt, “Put-that-there: Voice and Gesture at the Graphics Interface,” Proc. SIGGRAPH80, (1980).

[16] T. Starner and A. Pentland, “Real-Time American Sign Language Recognition,” IEEE Trans. on Pattern Analysis and Machine Intelligence 20, (1998) 1371–1375.

[17] C. W. Ng and S. Ranganath, “Real-Time Gesture Recognition System and Application,” Image Vis. Comput.,20(13–14) (2002) 993–1007.

[18] K. Abe, H. Saito, and S. Ozawa, “Virtual 3-D Interface System via Hand Motion Recognition from Two Cameras,”IEEE Trans. Syst., Man, Cybern. A, 32(4) (2002) 536–540.

[19] Md. Hasanuzzaman, V. Ampornaramveth, Tao Zhang, M.A. Bhuiyan , Y. Shirai and H. Ueno, “Real-Time Vision-Based Gesture Recognition for Human Robot Interaction”, in the Proc. of IEEE Int. Conf. on Robotics and Biomimetics, Shenyang China, (2004).

[20] Vuthichai Ampornaramveth and Haruki Ueno, “Software Platform for Symbiotic Operations of Human and Networked Robots”, NII Journal, 3 (2001) 73-81.
[21] J. Segen, “Controlling Computers with Gloveless Gestures,” in Proc. Virtual Reality Systems Conf., New York, (1993)66–78.
[22] D. Kortenkamp, E. Huber, and R. P. Bonasso, “Recognizing and Interpreting Gestures on a Mobile Robot,” in Proc. Artificial Intelligence (AAAI) Workshop, Portland, (1996) 915–921.
[23] Elena Sanchez-Nielsen, Luis Anton-Canalís and, Mario Hernández-Tejera , “Hand Gesture Recognition for Human-Machine Interaction”, Journal of WSCG, 12(1-3) (2003).