Journal of Physics: Conference Series

PAPER • OPEN ACCESS

Sign Language Recognition Using Convolutional Neural Network

To cite this article: A Sunitha Nandhini *et al* 2021 *J. Phys.: Conf. Ser.* **1916** 012091

Sign Language Recognition Using Convolutional Neural Network

A Sunitha Nandhini [1], Shiva Roopan D [1], Shiyaam S [1], Yogesh S [1]
[1]Computer Science Department, Sri Krishna College of Technology, Coimbatore, India.
shivaruby46@gmail.com

Abstract. Sign language is a lingua among the speech and hearing-impaired community. It is hard for most people who are not familiar with sign language to communicate without an interpreter. Sign language recognition appertains to track and recognize the meaningful emotion of human-made with head, arms, hands, fingers, etc. The technique that has been implemented here, transcribes the gestures from sign language to a spoken language which is easily understood by the listening. The gestures that have been translated include alphabets, words from static images. This becomes more important for the people who completely rely on gestural sign language for communication tries to communicate with a person who does not understand the sign language. Most of the systems that are under use face a recognition problem with the skin tone, by introducing a filter it will identify the symbols irrespective of the skin tone. The aim is to represent features that will be learned by a system known as convolutional neural networks (CNN), which contains four types of layers: convolution layers, pooling/subsampling layers, nonlinear layers, and fully connected layers.

Keywords: Sign language, cnn, training, dataset, filters

1. Introduction

A sign language interpreter is a significant step toward improving contact between the deaf and the general population. Sign language is a natural language used by hearing and speech impaired people to communicate. It uses hand gestures instead of sound to convey messages or information. Sign language can vary from one part of the world to another. Due to this, people find difficulty in communicating with normal people because normal people cannot understand sign languages. There arises a need for sign philological translators, which can translate sign language to spoken language. However, the availability of translators is limited when considering the sign language translators and these translators have many limitations. This led to the development of a sign language recognition system, which can automatically translate sign language into the text as well as a speech by effective pre-processing and accurate classification of the signs. According to recent developments in the area of deep learning, neural networks may have far-reaching implications and implementations for sign language analysis. In the proposed system, Convolutional Neural Network (CNN) is used to classify images of sign language because convolutional networks are faster in feature extraction and classification of images over other classifiers.

The environment may also recognise a sign as a compression technique for information transmission, which is then reconstructed by the receiver. The signs are divided into two categories: static and dynamic signs. The movement of body parts is frequently included in dynamic signs. Depending on the meaning of the gesture, it may also include emotions. Depending on the situation of the context, the gesture may be widely classified as:

- Arm gestures
- Facial / Head gestures
- Body gestures

2. Existing System

This research suggested the use of filters in the sign language translation algorithm because the existing system has low accuracy as it faced issues with skin tone identification. Sign language conversion can reach a maximum of 96% of accuracy but achieving that can be a tedious task. The current system failed to obtain this accuracy as it lagged to identify the skin tone under the low light areas.

3. Objectives

The key goal is to recognize the sign with maximum accuracy apart from different light,dark conditions must be developed

4. Proposed System

In this article, the filtering of images plays an important role. It improves the accuracy of identifying the symbols even in low light areas. Before the process of saturation and grey scaling the image is sent to the filtering system where it tries to find the symbol shown in the hands, after recognizing the symbol the image is further processed and final result which is the word is obtained.

5. Background And Related Work

Building efficient sign language processing systems necessitates both a knowledge of Deaf culture and a knowledge of sign languages in order to construct systems that account for sign languages' diverse linguistic aspects.This paper outlines the context and addresses recent reviews of sign language processing that do not take a systematic approach to the issue.

6. Image Recognition

The software requirements of this system are python, Open Source Computer Vision Library (OpenCV), TensorFlow, and NumPy. As python is the fastest when compared to other languages, it is used by this system. TensorFlow is an open-source machine learning tool that is used to train the sign images from start to finish. This framework makes use of OpenCV is a free and open-source software library for computer vision and machine learning. Numpy is a Python library that adds support for big, multi-dimensional arrays and matrices, as well as a wide range of high-level mathematical functions that can be used for them. [1]. The proposed system not only recognizes the digits and alphabets but also recognizes the words of sign language. Background elimination is done by giving a range that lies between the color range of the human hand. Thus this dynamically recognizes the hand region. The lighting condition problem is corrected by using the color of the human hand.

7. Image Recognition System

The acquisition of sign data is the first step in the sign language recognition system. The data can be obtained in a variety of ways.

7.1 Data Acquisition

The method of taking visual photographs, such as those of a physical scene. Preprocessing: The purpose of preprocessing is to optimise the input image data by suppressing unnecessary noise or enhancing essential image features for more processing.Feature Learning: Feature learning constructs its derived features from the original evaluated data in order to be descriptive and facilitate subsequent learning. Generalization moves and theRecent and Innovative Trends in Computing and Communication is an international journal that focuses on recent and innovative trends in computing and communication. steps[2], and in certain cases, this has resulted in clearer individual interpretations. Classification is a process that is related to categorization, which is the process of distinguishing concepts and objects from one another.Recognition is concerned with finding trends and regularities in results. The method of classifying input data into objects or classes based on key features is known as pattern recognition[3].

7.2 Sign Language

Sign language is used to communicate with the 466 million people who are deaf and hard of hearing around the world. American Sign Language (ASL), Israeli Sign Language (ISL), Pakistani Sign Language (PSL), South Korean Sign Language (SKL), Taiwanese Sign Language (TWSL), Arabic Sign Language (ASL), and so on.

7.3 Rescaling

Some of the existing systems do not include the resizing step in pre-processing. Image resizing is important to increase or decrease the total number of pixels. In the proposed system, the images are resized by decreasing the number of pixels. As for size increases, the time it takes to process also increases. So, the reduction of size is being done in the first step of preprocessing

7.4 Colour Conversion

Some of the existing systems are processing color images which makes complex computation because the color image has more bits. So, the proposed system converts the resized image into the greyscale image in pre-processing. Take RGB values for each pixel and emit a single value representing the pixel's brightness when transforming an RGB image to greyscale[5]. It is more effective to use greyscale images rather than RGB images since greyscale images have less detail.

7.5 Detecting The Hand Region

Most of the existing systems have a limitation on background subtraction. In the proposed system, the hand region is being tracked for background elimination which gave better results over existing systems. As each person’s hand color and the lighting condition in each place varies, it is necessary to track the hand region by giving a range that lies between the color range of the human hand[6]. Thus this dynamically recognizes the hand region.

7.6 Obscure The Image

Some of the existing systems have limitations on finding the edges and features of images. To overcome this, the proposed system used a masking process to hide some portions of an image to reveal some portions which help to identify the edges and features of images easily. Thus the greyscale image is converted into a binary image.

7.7 Neural Network:

The Convolution layer is often the first layer. The picture is received by it (a matrix of pixel values). Assume the input matrix is read from the image's top left corner. After that, the programme selects a smaller matrix, known as a filter, to position there (or neuron, or core). Convolution is then generated by the filter, which passes along the input image.The task of the filter is to multiply its values by the pixel values from which they come. A number of these multiplications are combined. After that, a single amount is gathered. Since the filter has already read the picture in the upper left corner, it moves 1 unit to the right and repeats the process..A matrix is obtained after running the filter over all locations, but it is smaller than the input matrix.

7.8 Convolution Layer:

Convolution is a special operation that derives multiple characteristics from the data. It extracts low-level features such as edges and corners in the first step. Then upper-level layers extract functionality at a higher level. In CNNs, for the 3D convolution process. The input is N x N x D in dimension, and it is convolved with the H kernels, each of which is k x k x D in size. When one input is convolutioned with one kernel, one output feature is generated, and when H kernels are convolutioned independently, H features are produced. Each kernel is shifted from left to right, starting at the top-left corner of the input. If a kernel enters the top-right corner, it is shifted one element downward before being moved one element at a time from left to right. The procedure is repeated until the kernel hits the bottom-right corner of the screen[6]. A mathematical procedure that takes two inputs, such as an image matrix and a filter or kernel, is known as convolution. The image matrix is a digital representation of image pixels and the filter is another matrix that is used to process the image matrix. The kernel's size is much smaller than the image's, allowing us to process any aspect of the image. Apply a filter to the image matrix in this layer. This filter matrix is used in combination with the image matrix seen in Figure to perform convolution.Any number of convolution layers can be added based on the features to be removed. The convolution function needs four arguments: the first is the number of filters, the second is the structure of each filter, the third is the input shape, and the fourth is the image form and resolution.The triggering function to be used is the fourth argument. The activation mechanism determines which neuron can fire next.

7.9 Pooling Layer:

A pooling layer is another building block of CNN[8]. The main aim of pooling is to downsample the image matrix into a smaller matrix. Its function is to reduce the difficulty in computing the large image matrix obtained from the convolution operation. The pooling layer operates on each feature map independently. After the convolution operation, pooling the operation needs to be performed on the resultant matrix to reduce the complexity in processing these matrices. The primary aim of a pooling operation is to reduce the size of the matrix by downsampling it.

7.10 Flattening Layer:

Flattening is the method of transforming a pooled function map into a single column that can be transferred to a completely connected layer. Flattening is the process of transforming data into a one-dimensional sequence so that it can be passed on to the next layer. To make a single long function vector, flatten the contribution of the convolutional layers. It is attached to a totally connected sheet, which is the final classification model.

7.11 Fully Connected Layer:

The final pooling or convolutional layer's contribution, which is flattened before being fed into the completely connected layer. is the entry to the fully connected layer.In this layer, each neuron corresponds to weight and these weights are chosen randomly. Thus at each layer, this calculation takes place as g(W x+b) In Equation 4.1, x is the input vector, w is weight, b is bias and g is the activation function.[5] This calculation is repeated for each layer. To calculate the most reliable weights, the CNN network's completely linked component goes through its backpropagation process. Weights are allocated to each neuron that prioritise the most suitable mark. Finally, the neurons "vote" for one of the marks, and the assignment decision is taken depending on the winner of that vote.The final layer is used to get probability of the input being in a specific class after going through the Entirely Connected Layer connected layers.

8 Implementation

The above diagram is the basic architecture that was proposed for the system. This consist of the basic system modules such as image acquisition, pre-processing, feature extraction, and finally producing output based on pattern recognition

8.1 Preprocessing Model:

The primary purpose of pre-processing is to optimise image data by reducing unnecessary anomalies or improving image features in preparation for further processing. [number six] Cropping, resizing, and grey scaling are examples of pre-processing. It is an attempt to capture the essential pattern that reflects the uniqueness of data without noise or unintended data.

8.2 Feature Extraction:

The segmented images are evaluated after colour thresholding to determine the specific features of each symbol. Each symbol will feature a different combination of finger tip locations. As a consequence, this will be the only attribute necessary for recognition.The centroid of each frame is measured and placed into a 2 n list, which represents a set of X and Y coordinates, when the acquisition frames are read one by one Until being sent to the identification point, the 2 n array is transposed into a n 2 array.

8.3 Image Preprocessing(Filters):

Adjusting the skin tone(of the image) before feeding the image into the algorithm enhances the model's performance. To do this, we transform the video to HSV (hue, saturation, value), which eliminates the skin tone's fluctuation in performance [4].

8.4 Cropping:

Cropping is the process of removing unnecessary sections of an image in order to better frame the subject matter or adjust the aspect ratio[3].

8.5 Resizing:

Photos are resized to suit the available or reserved space. Resizing photographs is a technique for preserving the original image's quality [5]. The physical scale is affected by changing the resolution, but not the resolution is affected by changing the physical size.

9 Evaluation Metrics

- [7-8] True Positive (TP): A true positive is when the algorithm forecasts the positive class accurately. (If the expected and real signs are the same, the outcome is true positive.)
- False Positive (FP): In a false positive, the algorithm forecasts the positive class wrongly. (A false positive happens where the expected sign varies from the real sign.)
- True Negative (TN):A true negative is when the algorithm forecasts the negative class accurately. (If the expected sign is the same as the real sign, it is true negative.)

• False Negative (FN): A false negative is a result in which the algorithm forecasts the negative class incorrectly. (If a projected sign varies from a negative sign (not a real sign), the consequence is a false negative.)
• Precision: Precision (P) equals the number of true positives (TP) divided by the number of true positives plus the number of false positives (FP).The percentage of applicable results is referred to as precision. Precision = TP/(T P+FP)
• Recall: The number of real positives (TP) divided by the number of true positives plus the number of false negatives equals recall (R) (FN). The proportion of overall related findings correctly classified is referred to as recall. TP/(T P+FN) = Recall
• Sensitivity: In certain fields, sensitivity (also known as true positive intensity, memory, or chance of detection) refers to the number of true positives that are correctly defined as such (e.g., the percentage of the predicted sign which correctly identified as the actual sign). Sensitivity = TP/(T P+FN) (6.3)
• Specificity:The proportion of real negatives that are accurately defined as such (e.g., the percentage of the expected sign that is correctly identified as a negative sign (not actual sign)) is determined through precision (also regarded as the true negative rate).
Specificity = TN/(FP+TN)
•F-Score: The F score is used to determine a test's accuracy, and it does so by combining precision and recalls. By combining accuracy and memory, the F score will offer a more accurate assessment of a test's results. F = 2;
For each sign a total of 200 images are taken for evaluation. Here Recall and Precision evaluation metrics are calculated for each sign. Table 1 the proposed system's test findings The number of True Positives (TP), False Positives (FP), True Negatives (TN), and False Negatives (FN) generated while measuring these signs are mentioned in the table below. 13 signs are shown here as an example. Table 2 shows the Recall and Precision that are calculated for different signs of the proposed system using the formulae mentioned above. The calculated values are shown for 13 signs which are part of the proposed system. Thus for each sign, the output precision can be known from this table and its sensitivity is also given.

10 Result

The device was put to the test with real-time video feedback, and the results were tallied. In any event, the SL recognition system's success is perfect. On all of the signs, the highest precision is 90%. This means that the machine is capable of understanding the plurality of signals. Figures 1-9 shows the results. Tables 1 and 2 shows the Sign names.

Software:
Operating System : Windows 7/8/10
Language : Python 3.7
Tools : Tensorflow,open cv ,pyttsx3,numpy,keras
Browser : Firefox / Chrome / Internet explorer

Hardware:
Processor : Intel Core i7
RAM : 8GB
Hard Disk : 1TB
Mouse : logical optical mouse
Keyboard : logical 107 keys
Motherboard : Intel
Speed : 3.3GHZ

11 Visual Results:

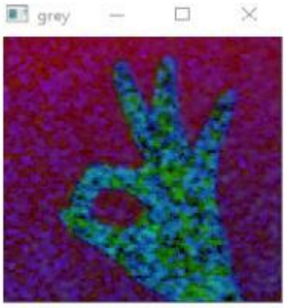
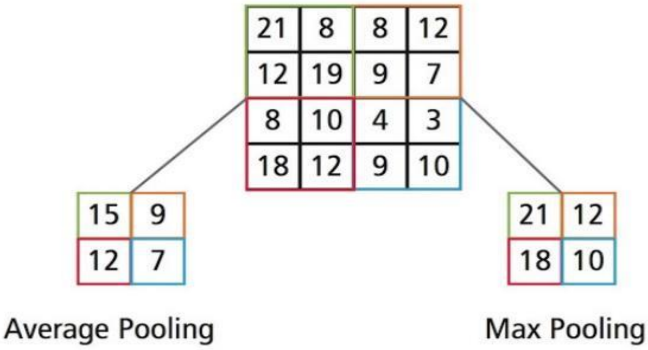


Figure 1. Grey scale image

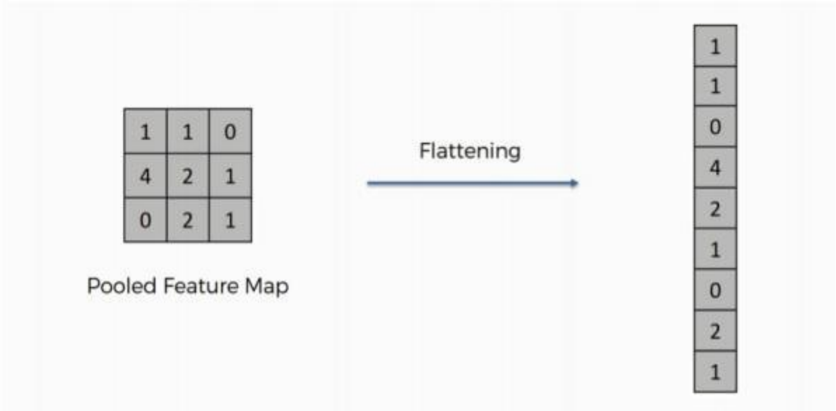


21	8	8	12
12	19	9	7
8	10	4	3
18	12	9	10

15	9
12	7

21	12
18	10

Figure 2. Pooling layer diagram



1	1	0
4	2	1
0	2	1

1
1
0
4
2
1
0
2
1

Figure 3. Flattening layer diagram

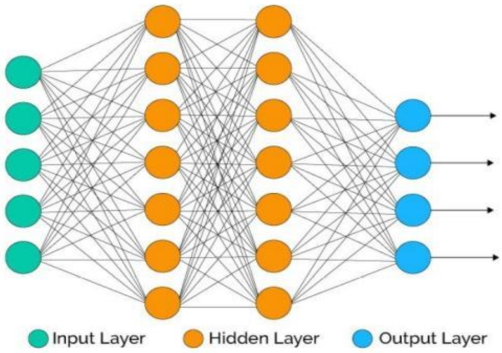


Figure 4. Fully connected layer diagram

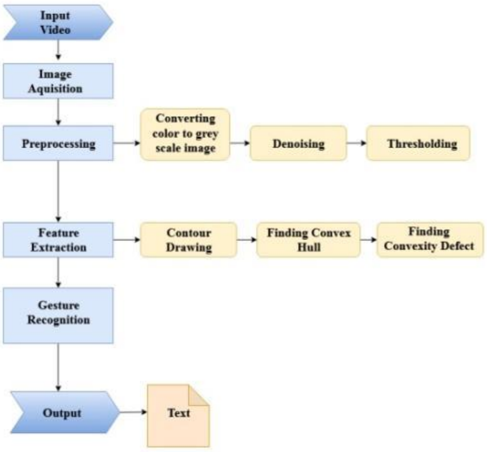


Figure 5. Proposed Architecture diagram

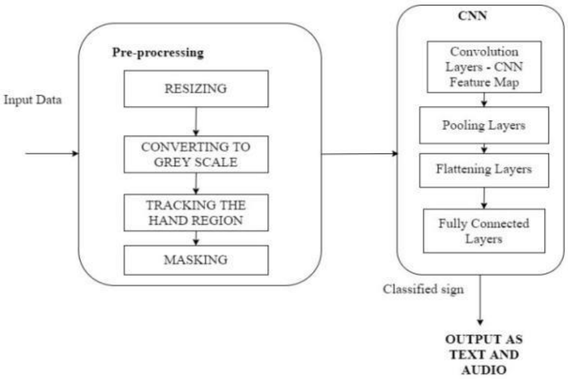
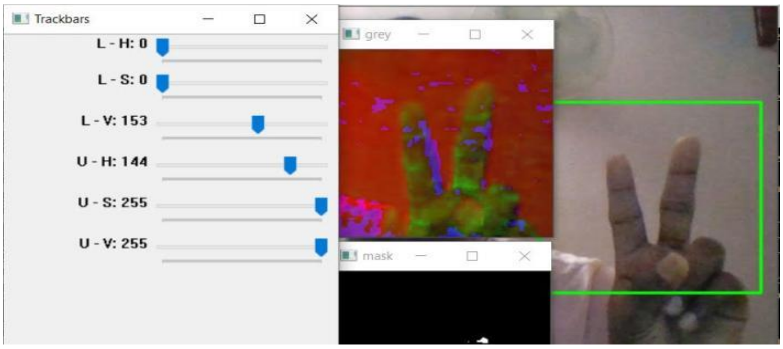


Figure 6. Pre-processing model diagram



L - H: 0
L - S: 0
L - V: 153
U - H: 144
U - S: 255
U - V: 255

Figure 7. Image processing with filter output

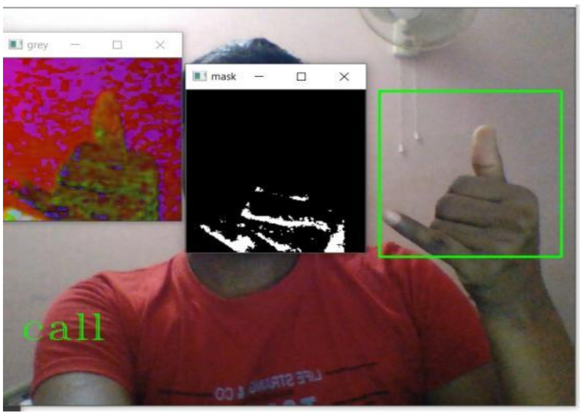


Figure 8. Sample output (sign language recognition)

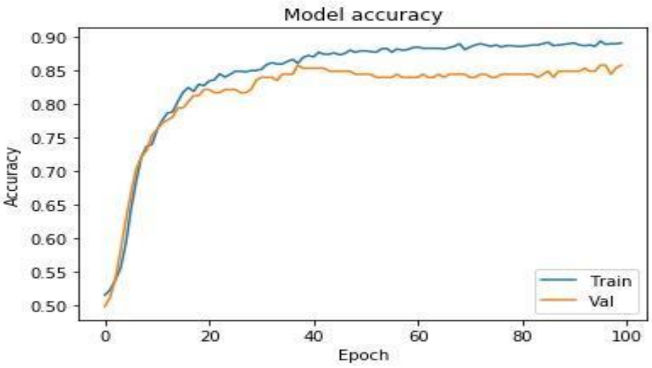
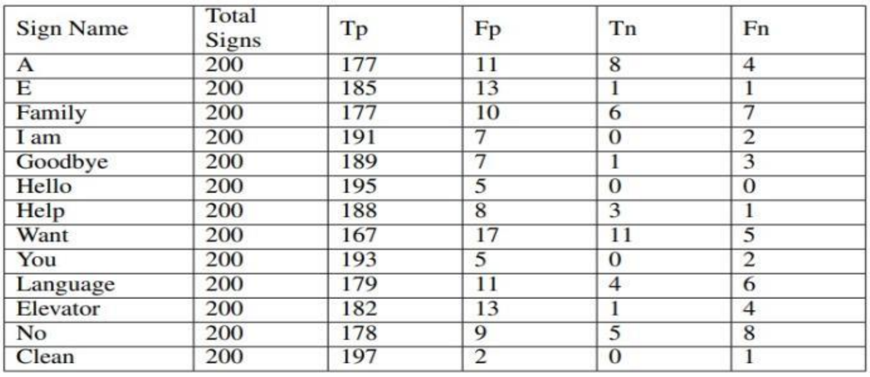


Figure 9. Trained Modal Accuracy outpu

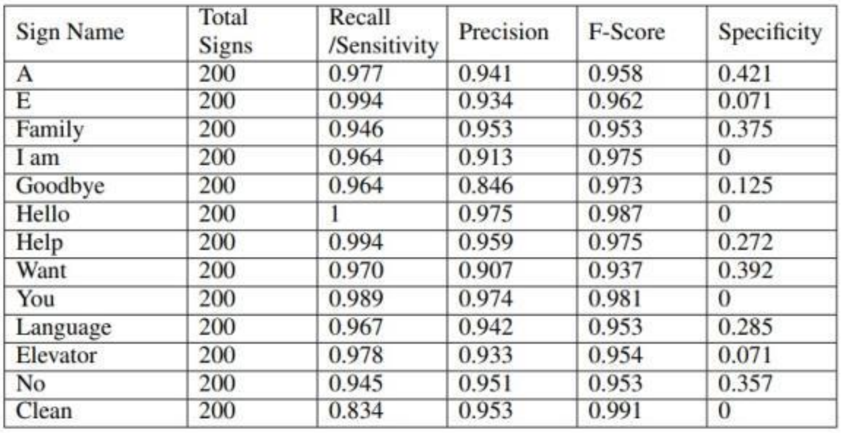
Table 1. Sign names



Sign Name	Total Signs	Tp	Fp	Tn	Fn
A	200	177	11	8	4
E	200	185	13	1	1
Family	200	177	10	6	7
I am	200	191	7	0	2
Goodbye	200	189	7	1	3
Hello	200	195	5	0	0
Help	200	188	8	3	1
Want	200	167	17	11	5
You	200	193	5	0	2
Language	200	179	11	4	6
Elevator	200	182	13	1	4
No	200	178	9	5	8
Clean	200	197	2	0	1

Number of True Positive (TP), False Positive (FP),True Negative (TN) and False Negative(FN) for some signs

Table 2. Sign names



Sign Name	Total Signs	Recall /Sensitivity	Precision	F-Score	Specificity
A	200	0.977	0.941	0.958	0.421
E	200	0.994	0.934	0.962	0.071
Family	200	0.946	0.953	0.953	0.375
I am	200	0.964	0.913	0.975	0
Goodbye	200	0.964	0.846	0.973	0.125
Hello	200	1	0.975	0.987	0
Help	200	0.994	0.959	0.975	0.272
Want	200	0.970	0.907	0.937	0.392
You	200	0.989	0.974	0.981	0
Language	200	0.967	0.942	0.953	0.285
Elevator	200	0.978	0.933	0.954	0.071
No	200	0.945	0.951	0.953	0.357
Clean	200	0.834	0.953	0.991	0

Recall/Sensitivity, Precision, F-Score, Specificity values for some signs

12. Conclusion

The proposed system successfully predicts the signs of sign and some common words under different lighting conditions and different speeds. Accurate masking of the images is being done by giving a range of values that could detect human hand dynamically. The proposed system uses CNN for the training and classification of images. For classification and training, more informative features from the images are finely extracted and being used. A total of 1750 static images for each sign is used for training to get the accurate output. Finally, the output of the recognized sign is shown in the form of text as well as converted into speech. The system is capable of recognizing 125 words including alphabets. Thus this is a user-friendly system that can be easily accessed by all the deaf and people.

References

[1] S. C. W. Ong and S. Ranganath, ―*Automatic sign language analysis: A survey and the future beyond lexical meaning*,‖ IEEE Trans. Pattern Anal. Mach. Intell., vol. **27**, no. 6, pp. 873– 891, Jun. 2005.

[2] L. Ding and A. M. Martinez, ―*Modelling and recognition of the linguistic components in American sign language*,‖ Image Vis. Comput., vol. **27**, no. 12, pp. 1826– 1844, Nov. 2009.

[3] D. Kelly, R. Delannoy, J. Mc Donald, and C. Markham, ―*A framework for continuous multimodal sign language recognition*,‖ in Proc. Int. Conf. Multimodal Interfaces, Cambridge, MA, 2009, pp. 351–358

[4] G. Fang, W. Gao, and D. Zhao, ―*Large vocabulary sign language recognition based on fuzzy decision trees*,‖ IEEE Trans. Syst., Man, Cybern. A Syst. Humans, vol. **34**, no. 3, pp. 305–314, May 2004.

[5] Haldorai, A. Ramu, and S. Murugan, Social Aware Cognitive Radio Networks, Social Network Analytics for Contemporary Business Organizations, pp. 188–202. doi:10.4018/978-1-5225-5097-6.ch010

[6] R. Arulmurugan and H. Anandakumar, Region-based seed point cell segmentation and detection for biomedical image analysis, International Journal of Biomedical Engineering and Technology, vol. **27**, no. 4, p. 273, 2018.

[7] N. Purva, K. Vaishali, *Indian Sign language Recognition: A Review*, IEEE proceedings on International Conference on Electronics and Communication Systems, pp. 452-456, 2014.

[8] F. Pravin, D. Rajiv, HASTA MUDRA *An Interpretation of Indian Sign Hand Gestures*, 3rd International conference on Electronics Computer technology, vol. **2**, pp.377-380, 2011