Proceedings of the Fifth International Conference on Trends in Electronics and Informatics (ICOEI).
IEEE Xplore Part Number:CFP21J32-ART; ISBN:978-1-6654-1571-2

Hand Gesture Recognition for Sign Language Using Convolutional Neural Network

[1] Dr.J Rethna Virgil Jeny, [2]A Anjana, [2]Karnati Monica, [2]Thandu Sumanth ,[2] A Mamatha
[1] Professor, [2]UG Scholar
Department of Computer Science Engineering,
Vignan Institute of Technology and Sciences
[1] jeny.navagar18@gmail.com

ABSTRACT
Speech disorder is a condition which affects one's ability to speak and hear. Those people use the separate language that is sign language, using that they can communicate with normal people. In this paper we developed a system which can translate sign language to text and then to audio, thus can improve communication with sign language. The system takes image data using the webcam of the computer, then pre-processing of the image is done using masking technique where the hand is masked to recognize the signed alphabet. Using Convolutional neural network algorithm features are mapped and classification is performed to classify the images accordingly, now image that is given as input is predicted and the translation is done in the form of text, then converted to audio. English alphabet are used as the data set for this system that is all the 26 alphabet are taken in the form of masked images. We've used 45500 images for training and 6500 images for testing.

Key words
Hand Gesture Recognition, Feature Extraction, Pre Processing, Classification, CNN

1. INTRODUCTION

In this new era, science is immensely developed and people are expecting more automated products for their daily needs. Nowadays technology is developed, as our homes are mostly automated by the machines to reduce the work of man. Although some of us are unable to hear or speak unfortunately, for those people we have sign languages that they can be shown using their hands and can communicate with others, but most of us are not familiar with the sign language so to interact with the people like us it will be very difficult task to them so by using this system it will be convenient, for people who are deaf or dumb.

Hand gesture is a nonverbal communication form which is visible in the body language of a person. These hand gestures are common when we are speaking something to communicate some kind of important information. Normally Gestures includes moments of body, hands and even facial expressions. Gestures are bit different from sign language, although sign language can be shown using the gestures mainly using some hand moments. speech impairment problem. In this system, sign language is in the form of English alphabet, by those we can convert signs into alphabet and to text, further text can be converted to speech so that we can easily understand those people. Below is the alphabet that are shown using sign language.

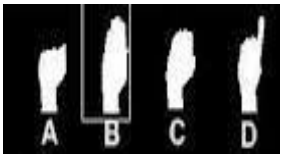

Fig.1. Hand Gestures for Sign Language

Interacting plays vital role in our lives, for the who are dumb they cannot hear to communicate with other people so they rely on the sign language interpreters but they may not be reliable. So computerized interpreters can fill that space by providing accuracy with cheaper price. The system that can convert the signs to text or audio can be very helpful in real time applications.

2021 5th International Conference on Trends in Electronics and Informatics (ICOEI) | 978-1-6654-1571-2/21/$31.00 ©2021 IEEE | DOI: 10.1109/ICOEI51242.2021.9453072

So naturally those people will expect some applications that can solve their communication problem from the computer engineers. As deaf and dumb can only rely on visuals so that's the reason the application is in the form of visual text and speech for the smooth communication. In this application input is given as some sign language hand gesture that undergoes some preprocessing technique then algorithm is applied and output will be translated in the form of text and audio.

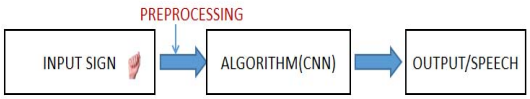

Fig.2. Flow of the Application

The drawback for the deaf is their employment issues. Communication has always a prominent part in getting the task solved this system can help in solving the issue of unemployment for deaf. To propose this type of system there should be alphabet that are more in the use, therefore we used American Sign Language (ASL). Video Capturing is processed using CNN and by using the CNN algorithm, features are mapped and summarized by the internal three layers of the algorithm, at last classes are formed based on their features and image is predicted this the internal function of the algorithm which gives the output.

The remaining sections in the paper is structured as –

Section 2 consists of related work that is done on hand gesture recognition for sign language. Section 3 will give brief idea about the data set that we have used in this paper. Section 4 explains the proposed system that we have developed and shown the results. Section 5 consists of conclusion and future work.

2. RELATED WORK

The deaf people are normally less competent in writing a spoken language. For example, if there is an accident then deaf people cannot quickly communicate by writing so a translator is required. Designing a good algorithm is the most difficult part in designing a sign language converter to classify the static hand gesture images with good accuracy [1]. A speech recognition and movement capture together is used in a java program. So, in front of the movement sensor when a deaf person speaks in sign language, people behind the screen can easily understand even though they cannot speak the language [2]. The majority of the people in the places in united states and in Canada the people with speech defect mostly use American sign language(ASL). But it is difficult for people who are non-sign language speakers. By using deep learning, they the solution for this problem using ASL data set [3]. Before the input is given to Deep learning algorithms for recognition it goes through a sequence of pre-processing steps for converting a image (gesture) to a valid data by using Hand Track Point analysis. The existing Methods depend upon the external devices like movement sensor gloves or Microsoft Kinect to pick up the quality of finger movements [4]. CNN works like the human brain's visual cortex. To achieve this different convolutions are performed by using filters which are used as trainable weights. For each channel multiple filters can be applied and feature maps can be formed [5]. The alteration in the kernel size can be inspired by dataset which includes the background, preprocessing can be done to remove the background. The model is followed by a thought that the first layer with small size kernel would capture small features like outline of the hand, edges of the finger and its shadows. Hopefully the large size kernel captures the merger of these small features [6]. Bayesian networks also achieved high accuracies, they are very good at acquisition of temporal patterns, but the need very acutely predefined models that are defined before learning. A machine learning method called Transfer learning are usually trained on large data sets is used to prepare more definite data [7]. With given requirements explored by the nature of hand gesture recognition, to preserve the consistency in performance in

different situations a general invariant method is needed. Deep learning has been performing well in recent years mainly in the computer vision problems. In various topics it has shown its superiority in computer vision challenges. This high performance is somewhat due to recent advances in GPU architectures and its design [8]. For feature extraction HOG method is used. In this technique the shape or object inside the image can be described by edge directions or intensity distribution gradients. In this technique, the image is partitioned into small units and for each unit a histogram of gradient is calculated. It then generates a bin and the histogram of various samples based on angle and magnitude are grouped together [9]. To identify the image's ROI the determination of interest points present within the space of an image is required. A method is used to increase the number of interest points through FAST corner detection and SURF detection [10]. When sign objects are varying in both movement and shape in 3-dimensional space it is big challenge for sign language recognition, so from hand motions to detect the hand signs a sensor is used to detect 3 dimensional information. [11]. A computer vision system is used for Mexican sign language to automatically recognize the sign. The system has four reflectors (LED) and many number of lumens are used to enhance the quality of image acquisition [12]. Restricted Boltzmann Machine (RBM) which is a deep learning method is used to recognize sign language automatically from imaged data [13]. An enhanced classifier is used for the random forest to recognize the sign language and the classifier used is decision tree. [14]. A sensory glove with electronic device to collect hand motions that includes wrist, arms and fingers, is used for recognition of words of Italian sign language [15].

3. DATASET

In this paper we have used American Sign Language (ASL) for the sign language. The data set consists of English alphabet. Before training the model splitting is important. Generally, the dataset is split into 3 sets, training, testing and validation. First set is to make the model learn the features, second set is to test the model and see how accurately it is testing and the third step is for validating. In our case we split it into only 2 sets; we have taken 52000 images as the input data set which is partitioned into training and test data. 45500 images are taken as training data set and 6500 images are taken as testing data set. 1750 images of each alphabet is taken in the testing data. Here we basically used supervised learning approach where we trained our data to create a model and tested the model using testing data set to get the accuracy of the model. By using image masking method we converted the images into masked images.

Fig.3. English Alphabet In Sign Language

Using OpenCv module in python masking is performed on the image and then training and testing is done on the data set. Above are the alphabet signs using hand gestures.

3.1 PRE PROCESSING

Pre Processing is the process that is performed on the images before applying any algorithm on them. Main need for pre

processing an image is to improve the image by removing unwanted background and also for the enhancement of image features so we can apply deep learning algorithms on them. Pre Processing is done in many ways, in this project we used masking technique. Masking image means masking a particular part of the image to protect the important area, just as we use the tape during painting to protect a particular part of the wall. Masking an image can protect the area from the further changes that are made to the image.

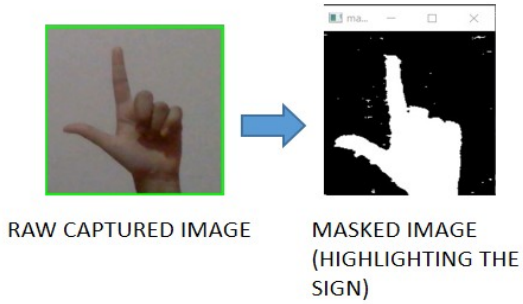

Fig.4. L Alphabet Sign Converting To Masked Image

For masking an image we used python module OpenCv where it has a function inRange, is used for masking. First two ranges of the skin colour is taken and kept each color range in an array, next BGR image is converted to HSV image and using the inRange() we mask the image with the help of HSV image. HSV stands for Hue Saturation image, this the colour scale which gives number readout of the image. HSV colour scale degree ranges from 0 to 360.

From the above image we can see the raw captured image is in a green box which is called Region of Interest (ROI). It is a part in the image that we wanted to use for some operations. From that region sign is taken and further processing done. In this system, the part of the sign in the image is filled with solid white colour and the remaining background is filled with solid black. The preprocessed image is the input to the convolution layer where input is in pixel format so that features can be mapped and summarized, then image can be mapped to the class that it belongs to. In the coming chapters we can get the brief idea about the whole process.

4. PROPOSED SYSTEM

In the proposed system we have mainly three steps they are Pre Processing, Feature Extraction and Pattern Matching or Sign Recognition. So, In the first step an image is taken as an input and pre-processed as explained in the above chapter using masking technique. Coming to the next step Feature Extraction is done using the CNN algorithm that means in the second stage we apply algorithm, it has three layers. From this step we can get the classified data to predict the image that is given and that Sign image is predicted and converted into the text form later this text is converted into audio. This application is mainly useful for the people who have some imperfections like deaf or dumb. So they can convey their thoughts to the other people by showing sign languages to the system which can convert to text and then to audio. This system is mainly for the people who have difficulty with hearing and speaking so there is no existing system that can be helpful to them for interaction. This proposed system will be very helpful to those people for the better communication.

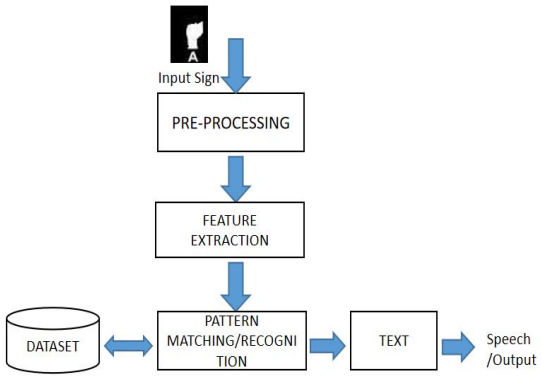

Fig.5. Flow Of The Proposed System

4.1 Algorithm

In the proposed system we have taken CNN algorithm for the classification as well as for the feature mapping. CNN stand for convolution neural network where it consists of three layers which perform

different task for extracting features from the image.

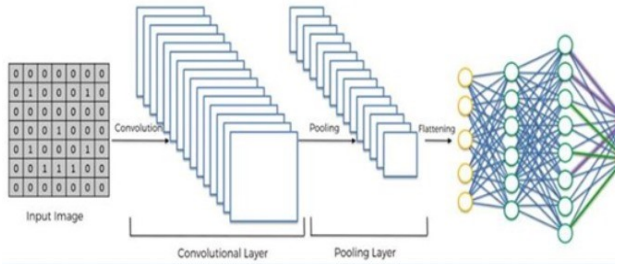

Fig.6. Steps Included In CNN Algorithm

From the above pic we got to know that CNN algorithm consists of three layers each layer's output is input to the next layer.

A) CONVOLUTIONAL LAYER

Firstly, input image is in the form array of pixels format, it is given to the first layer that is convolution layer, which applies filter to the input image that gives result as activation, by applying same filters repeatedly to the input gives the activation map that is called as feature map. The Feature map specifies the location of the identified feature in the image given as an input. The filters in the convolutional layer are called as convolutional kernel. Dot product is performed between the input image which is in the form of pixel and with filter to get the activation which results in Feature map.

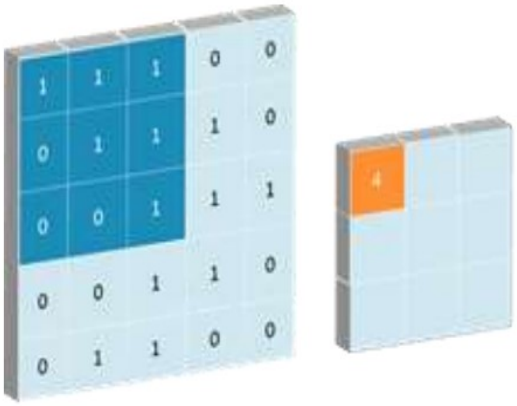

Fig.7. Convolutional 2D Layer

B) MAX-POOLING

The next layer is max pooling layer, responsibility of this layer is to down sample the feature maps by summarizing the features that are present in the patches of feature map. The output from the first layer is given as input to this layer. Feature map produced has sensitivity issue with the location of the feature to reduce this sensitivity max pooling layer performs down sampling of feature maps. Pooling layer performs independently with each feature map that are produced in the first layer.

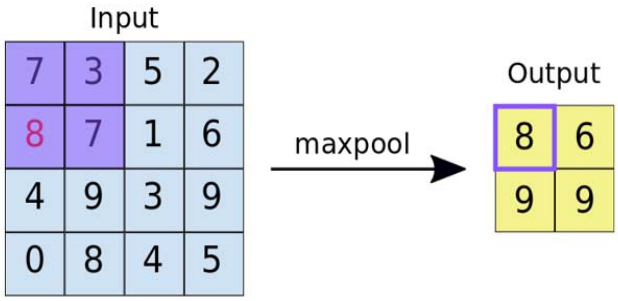

Fig.8. Max-Pooling Layer

C) FULLY CONNECTED LAYER

In fully connected layer, output of max pooling layer is given as input to this layer. Since the pooling layer gives 2D array and FC takes only 1D array, the 2D array is converted to 1D array by using flatten layer and then this 1D array is given to the FC layer. softmax function is utilized in the last layer, this gives the probability of that image belonging to that class, this process is known as classification. Then the image can be predicted to which class it belongs to. This is the main layer where classification is performed. By using dense function this layer is added to the algorithm. In fully connected layer there are some hidden nodes which are been set using the units parameter in the dense function. Then we get the output from this layer as prediction of the class that particular input images belongs to.

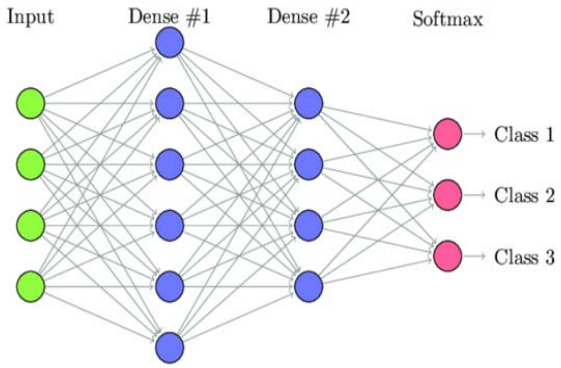

Fig.8. Fully Connected Layer

We conclude that in convolutional layer feature extraction and also classification of the images is done with respect to the class it belongs to. That means in our application sign images of alphabet are given as input and then feature map is done in the first layer and then classification of that image with sign is classified into its respective class by this we can predict the image sign. In the next chapter it will be clear about the implementation of our application.

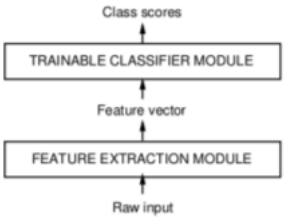

Fig.9. Classification Using CNN

4.2 IMPLEMENTATION

Implementation of this system is done with the python programming Language using the CNN algorithm and masking technique for image pre processing with the help of openCv module in python. Python is a language which is effective when dealing with the Machine learning and Artificial Intelligence. As python has number of modules to perform the particular tasks, Example: for image pre processing by just installing openCv module into the virtual environment we can perform the task. For implementation we have installed certain packages that are needed for the programs execution like openCv, numPy.. etc.

We have taken 3 Convolutional layers, 3 Max pooling layers, 2 Fully connected layers to build the CNN model. The input images given to first layer is of size (64,64) and taken 32 Filters each of size (3,3). It helps in detecting main features in the images and Trained with fully connected layer having 256 neurons. Selecting the number of neurons comes with many experiments and as the last fully connected layer is used for classification and since the no. of class labels are 26. It is built with 26 neurons and here softmax function is utilized for the classification. Next we have started prediction by using openCv. We have created a window to capture the live video frame by frame. While we give the gesture it needs to be preprocessed then only the sign will be recognized. So inside the live streaming we have created a simple region of interest (ROI) and when we placed the hand in that region a masked image is created. Now it is easy for the system to recognize the gesture as it is trained with these preprocessed images. Image pixels will be of 0 and 1 as it has black background and hand is of white color.

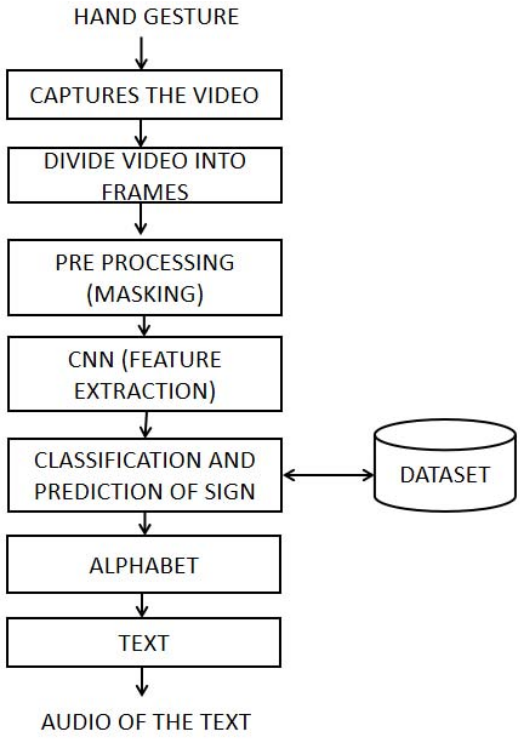

Fig.10. Sign Language Gesture Recognition and Converting To Speech

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 64, 64, 32)	896
max_pooling2d (MaxPooling2D)	(None, 32, 32, 32)	0
conv2d_1 (Conv2D)	(None, 32, 32, 32)	9248
max_pooling2d_1 (MaxPooling2	(None, 16, 16, 32)	0
conv2d_2 (Conv2D)	(None, 16, 16, 64)	18496
max_pooling2d_2 (MaxPooling2	(None, 8, 8, 64)	0
flatten (Flatten)	(None, 4096)	0
dense (Dense)	(None, 256)	1048832
dropout (Dropout)	(None, 256)	0
dense_1 (Dense)	(None, 26)	6682

Total params: 1,084,154
Trainable params: 1,084,154
Non-trainable params: 0

Fig 11. CNN Architecture

After it detected each alphabet we appended each alphabet together and formed into a text format and later to convert the text into audio we imported **pyttsx3** module and called init() method and created an object. And to play the audio call say() method and pass the text as a parameter. So we finally recognized the image and converted the signs into text and audio format. Our system achieved 98.55% accuracy, as mentioned earlier we have trained the system with 45,500 images and tested on 6500 images, and also we have experimented on 3 batch sizes 16, 32, 64. We got high accuracy when we take batch size 32 so we proceeded with the model. We have taken only 10 epochs with learning rate 0.1 and activation function we used is ReLU (Rectified linear unit).

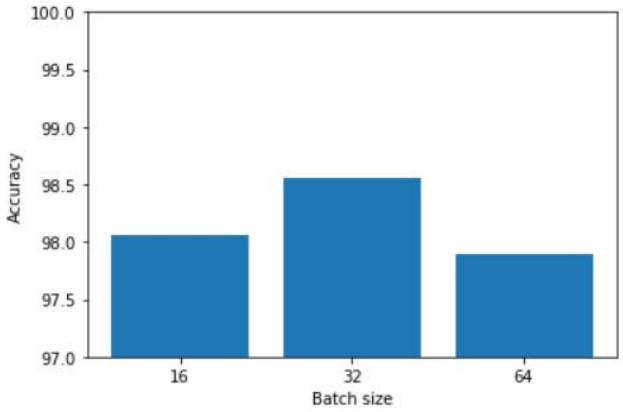

Fig.12. Bar Graph Between Batch Size And Accuracy

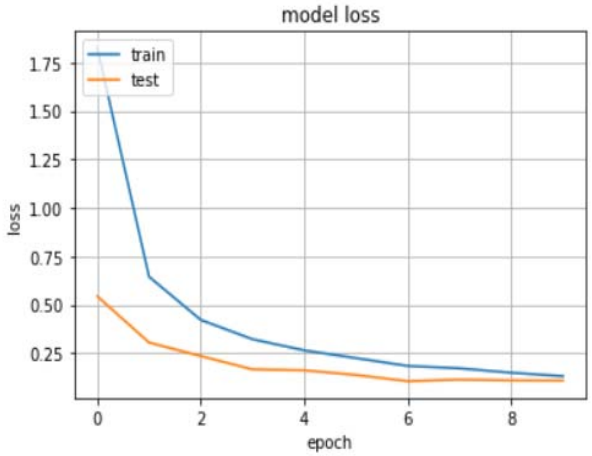

Fig.13. Graph Between Epoch And Model Loss

Figure 13 shows the loss for training and test data. As the epochs increases the loss percentage for both training and testing is getting decreased.

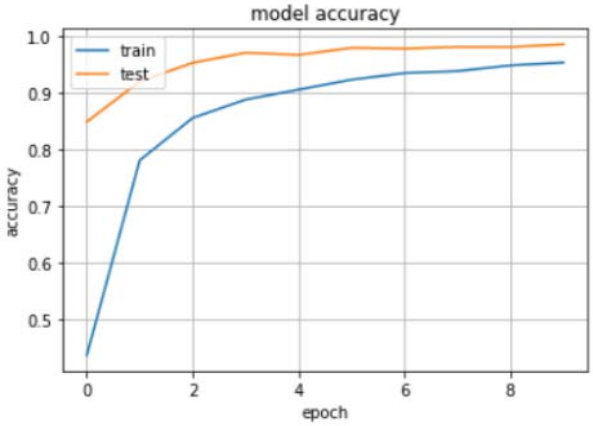

Fig.14. Graph Between Epoch And Model Accuracy

Figure 14 shows the accuracy for training and test data. As the epochs increases the accuracy for both training and testing is getting increased. Next we predicted the model by taking some live pictures and the system recognized the gestures and identified its class (‘A’,’B’ etc). Later we appended the

recognized classes and formed the text and also system reads out the text. Here are few images taken while experimenting.

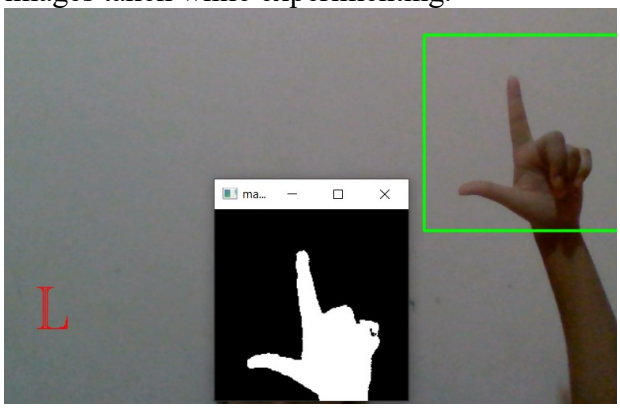

Fig.15. Alphabet Prediction

Fig.16. Output in the Form Of Text

5 CONCLUSION AND FUTURE WORK

The aim of this paper is to remove the conversation gap between the deaf-mute people and a normal person. We implemented and trained the convolutional Neural network with the American Sign language. This system allows us to give a gesture then it identifies the gesture, also it can form a sentence and speak out the sentence. The major difficult task is to remove the unwanted background of the image and noise that is captured in the region of interest so, to overcome this problem we used masking technique to mask the only hand gesture that is needed and the other unwanted background and noise is removed because of this pre processing technique only features can be extracted properly. Through this paper we are able tell that if we train the system with any other sign language it can recognize the language.

For future work includes recognition of gestures that are made with both hands as our system recognizes gestures given by only one hand. In our system it first recognizes the alphabets and converts them into text it is sometimes difficult to form a long sentence, so we would like to use signs of common words so that it is easy to form a sentence.

REFERENCES

[1]. Beena, M. V., Namboodiri, M. A., & Dean, P. G. (2017). Automatic sign language finger spelling using convolution neural network: analysis. *Int J Pure Appl Math*, *117*(20), 9-15.

[2]. Arsan, T., & Ülgen, O. (2015). Sign language converter. *International Journal of Computer Science & Engineering Survey (IJCSES)*, *6*(4), 39-51.

[3]. N. Muthukumaran, 'Analyzing Throughput of MANET with Reduced Packet Loss', Wireless Personal Communications, Vol. 97, No. 1, pp. 565-578, November 2017.

[4]. N. Muthukumaran, R. Aiswarya, S. Anna Sankari & K. Divya Bharathi, Deep Learning Neural Network Based Human Emotion Classification with ANFIS Algorithm, Irish Interdisciplinary Journal of Science & Research, Vol.4, Iss.3, Pages 105-111, July-September 2020.

[5]. Grandhi, C., Liu, S., & Rahoria, D. American Sign Language Recognition using Deep Learning.

[6]. VP. Anubala, N. Muthukumaran and R. Nikitha, 'Performance Analysis of Hookworm Detection using Deep Convolutional Neural Network', 2018 International Conference on Smart Systems and Inventive Technology, pp. 348-354, 2018, doi: 10.1109/ICSSIT.2018.8748645.

[7]. Bajaj, Y., & Malhotra, P. (2020). American Sign Language Identification Using Hand Trackpoint Analysis. *arXiv preprint arXiv:2010.10590*.

[8]. N. Muthukumaran, N. R. G. Prasath and R. Kabilan, "Driver Sleepiness Detection Using Deep Learning Convolution Neural Network Classifier," 2019 Third International conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud), Palladam, India, 2019, pp. 386-390.

[9]. Pigou, L., Dieleman, S., Kindermans, P. J., & Schrauwen, B. (2014, September). Sign language recognition using convolutional neural networks. In *European Conference on Computer Vision* (pp. 572-578). Springer, Cham.

[10]. Jeeva. R, Muthukumaran. N, 'Identification of Fictitious Messages in Social Network using E-Hits and Newsapi', International Journal of Innovative Technology and Exploring Engineering, Vol. 9, Issue. 10, pp. 7- 11, August 2020.

[11]. A.Karthika, N. Muthukumaran, R. Joshua Samuel Raj, 'An Ads-Csab Approach for Economic Denial of Sustainability Attacks in Cloud Storage', International Journal of Scientific & Technology Research, Vol. 9, Issue. 04, pp. 2575-2578, April 2020.

[12]. Garcia, B., & Viesca, S. A. (2016). Real-time American sign language recognition with convolutional neural networks. *Convolutional Neural Networks for Visual Recognition*, *2*, 225-232.

[13]. J. Rebekah, D. C. J. W. Wise, D. Bhavani, P. Agatha Regina and N. Muthukumaran, "Dress code Surveillance Using Deep learning," 2020 International Conference on Electronics and Sustainable Communication Systems (ICESC), Coimbatore, India, 2020, pp. 394-397, doi: 10.1109/ICESC48915.2020.9155668.

[14]. Strezoski, G., Stojanovski, D., Dimitrovski, I., & Madjarov, G. (2016, September). Hand gesture recognition using deep convolutional neural networks. In *International conference on ICT innovations* (pp. 49-58). Springer, Cham.