SIGN LANGUAGE PREDICTOR

[1]GUTTA SRIDEVI, [2]RAMINENI BHAGYA SRI, [3]RAVELLA PRASANTH SAI,
[4]SHAIK DAVOOD ALI, [5]PRAVEEN KUMAR REDDY PILLI

[1,2,3,4,5]Seshadri Rao Gudlavalleru Engineering College,Gudlavalleru, India
E-mail: [1]guttasridevi144@gmail.com, [2]raminenibhagyasri02@gmail.com, [3]saip2066@gmail.com, [4]skdavoodali@gmail.com, [5]praveenprem369@gmail.com

Abstract - Since that sign language is not a common language, quite few people can comprehend it. The majority of hearing societies find it challenging to communicate with the deaf group because of this. One can anticipate that within a few decades, digital technology would have a significant impact on how people go about their daily lives and that everyone will communicate with machines either by gestures or speech recognition. If we are able to foresee such a future, we should consider the physically disabled and take action to help them. Hence, computerized recognition systems provide a novel technique to interpret deaf signals without the aid of a professional. The 26 hand gestures with in dataset correspond to the letters of the English alphabet, from A to Z. This paper took into account the Hand Gesture Recognition standard dataset from the Kaggle website. Convolutional Neural Network (CNN), the type of neural network and its pretrained Mobile Net model used in this study, improve the predictability of the alphabet in American Sign Language (ASL).

Keywords - ASL Gestures, Deep Learning, CNN, Mobile Net

I. INTRODUCTION

Unlike spoken languages, that utilize auditory-verbal capabilities for communication, sign languages are complete, natural languages that rely on visual-manual modalities. In contrast to spoken languages, sign languages possess distinct linguistic features and vocabulary items. Despite the similarities between sign languages, communication between signature language speakers might not always be possible. Interpreters believe sign languages to just be natural languages because they developed through time in an unconstrained and unstructured approach. Neither of these is sign language based on spoken languages, nor are they only a condensed version of them. Instead, they have evolved independently and have complex grammatical standards. Body language, a form of nonverbal communication that employs facial expressions, gestures, and postures to convey meaning, is distinct from sign language. Sign language is an entire language that has unique grammar, syntax, and vocabulary. The classification of sign languages may be done using their origins. Some of the most commonly used sign gestures, American sign gestures and French sign gestures, both emerged alongside the spoken languages of their respective nations. French sign gestures had an impact on the development of American sign gestures in the United States in the early 19th century, but French sign gestures originated in France in the late 18th century. There are other different sign languages that are utilized across the globe in addition to these, each having its own vocabulary and syntax. Nicaraguan Sign Gestures, which were formed in the 1980s by a group of deaf adolescents who gathered in a school for the deaf in Nicaragua, are one example of a sign language that has grown among tiny, isolated communities of the deaf. They created their own system of motions and gestures to interact, as they had no prior experience with sign language.

Sign languages have become effective means of communication for people who can't speak or hear. Deaf people use these languages to communicate with one another and with hearing people who have acquired the language. Local Deaf cultures have adopted sign languages as an essential component, and they differ depending on the area and the deaf group that utilises them. Along with those who are deaf or hard of hearing, those who cannot physically communicate, have challenges using spoken language due to their condition or have deaf family members in their family interact with each other with the help of gestures. Those who might not have access to other modes of communication can express themselves, engage with others, and fully participate in society by learning sign languages.

The variety of gesture languages used nationally and globally varies by nation, and most nations have their own native sign language, and some may have multiple languages. For instance, whereas British sign gestures are used in the United Kingdom, American sign gestures are used in the United States and some regions of Canada. Even among the close relatives of the deaf, these languages are frequently unable to be understood by the non-disabled despite their widespread usage. This may result in communication hurdles inside the deaf community as well as between the deaf and hearing populations. There is no universal sign language, unlike spoken languages. This implies that communication might be difficult and inefficient amongst people who use various sign languages. Although textual communication is an option, it can be time-consuming and not always useful, especially when travelling.

In this digital technology, system also have to understand what deaf people are saying in front of camera. Hence, a recognition system is needed. The two different sorts of approaches for recognising signs in sign language are sensor-based and image-based. The first technique depends on wearing certain gloves or localised sensors. This method's key advantage is its capacity to accurately convey information regarding signals or gestures, such as the movement, rotation, orientation, and positioning of the hands. Second method is built on image processing, which does not require sensors or other hardware. This method solely uses pattern recognition and a variety of image processing techniques.

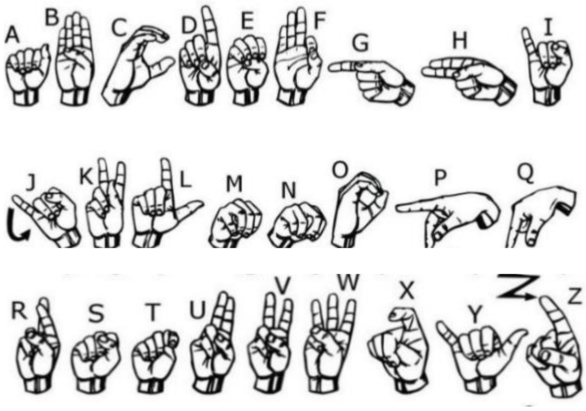


Figure 1: American Sign Alphabets

II. LITERATURE SURVEY

[1]The glove-based sign-to-text/voice converting system by for deaf and dumb is described in this research. The glove presents Arabic sign language characters as text on LCD and produces sounds through the speaker, reducing the communication gap between the deaf and the hearing. The significance of the study stems from its goal of assisting this group of nonverbal persons in communicating with others and enhancing their contributions to national development.[2]The project transforms user-performed American Sign Gestures signs into sentences of English language using a sensor glove.[3]The performance evaluation of the features and classifier is described in the work that Siddharth Kaslay, Tejal Kesarkar, and Kanchan Shinde have proposed. The features utilised in the proposed system are the Gray Level Cooccurrence Matrix (GLCM), Hu moment, and Color moment. To classify data, Support Vector Machine (SVM) is used.The suggested solution makes use of a collection of images depicting American alphabet gestures. Utilizing predictive measures of precision, recall, and f1-score, the system's performance is assessed. Their suggested system had an accuracy rate of 87%.[4]Apaper by Apoorva,Harshitha, Chaitra, Akshitha and Rajath discusses about a system that enables regular people and the deaf to communicate with one another. The study's objective is to provide a seamless discourse for those who do not know sign language. In this study, the skin's RGB value and the YCbCr colour space are used as a segmentation method. [5]The Support Vector Machine (SVM) Classifier, which is frequently used for classification and regression analysis, as well as Canny's edge detection and gradient histograms for feature extraction, are employed in the system's architecture. The main feature in this project is local contour sequence (LCS). In order to improve LCS and increase recognition accuracy, it can be utilised with other characteristics.[6]Three frames for every second of the video feed are captured. The frame with the static pose denoted by the hand was then identified by looking at three consecutive frames. A sign motion is recognised in this still position. Its meaning is subsequently determined by comparison to the database of gestures that have been previously stored. It uses Principal Component Analysis.[7]Using skin colour segmentation, it is able to extract indications from video sequences with dynamic and slightly crowded backgrounds. The suitable feature vector is extracted after making the distinction between static and dynamic gestures. Support vector machines are used to categorise these.

III. IMPLEMENTATION

The intent of this research is to develop a framework for sign language gestures that is automated.

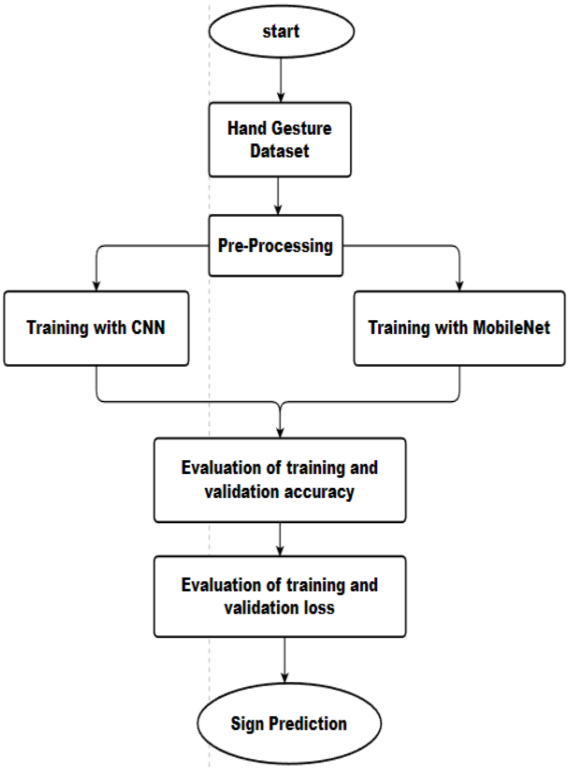


Figure 2: Methodology

The procedure is described in figure 2in which dataset from Kaggle undergoes preprocessing and trained using two models. One is CNN(Convolutional Neural Network) and the other is its pre-trained model MobileNet. Finally training as well as testing accuracy and loss is evaluated and it is ready for prediction.

a) CNN (Convolutional Neural Network)

A CNN (Convolutional Neural Network) is a Deep Learning training algorithm which can take an image as input, given various elements and objects like biases and weights in the image importance and distinguish between them.The layers used in our model are shown in figure 3. CNNs are made specifically to handle input data that has a grid-like layout, like pictures. They are made up of several layers that separate the features from the incoming data before categorising it into one or more groups. Backpropagation is a technique used to train CNNs, where the algorithm modifies the elements of the network like weights to reduce the discrepancy between the expected and actual output. To make sure the network learns accurate representations of the input data, the training procedure can take a while and necessitates a lot of labelled data. Convolutional, pooling and fully linked layers make up a CNN's fundamental building elements.In comparison to other classification methods, this method requires significantly less pre-processing.

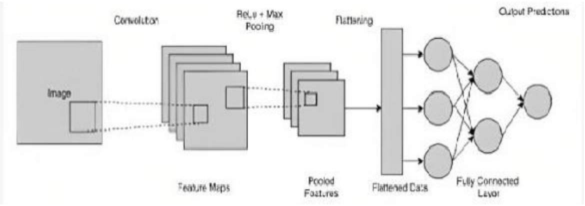
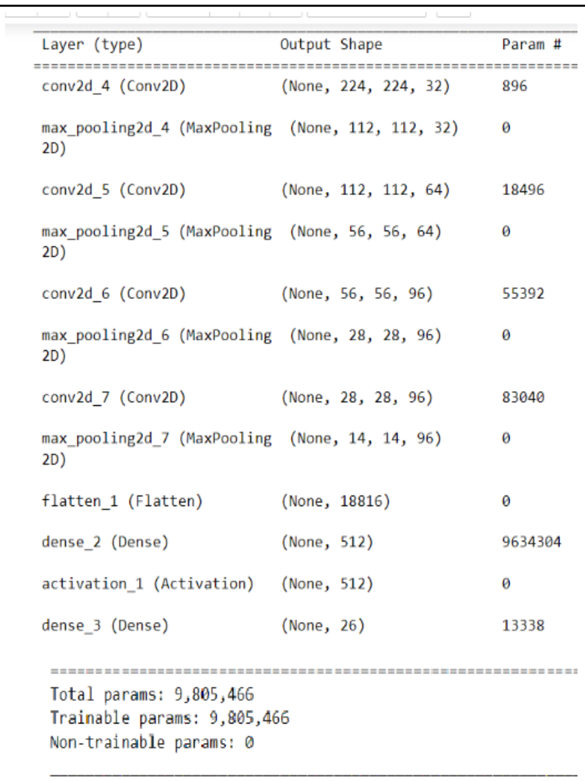


Figure 3: CNN Design

The process goes as follows in our model

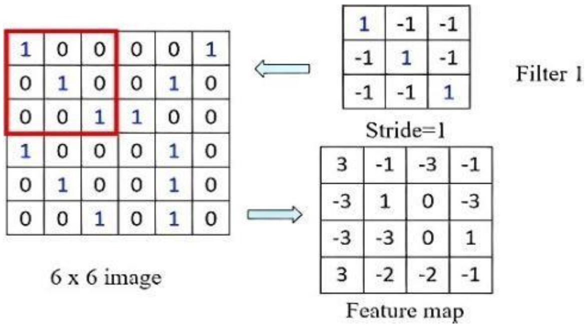
1. Convolution with 32 3x3 filters and ReLU activation function
2. Max Pooling
3. Convolution with 64 3x3 filters and ReLU activation function
4. Max Pooling
5. Convolution with 96 3x3 filters and ReLU activation function
6. Max Pooling
7. Convolution with 96 3x3 filters and ReLU activation function
8. Max Pooling
9. Flattening
10. Fully connected
11. ReLU
12. Fully connected + Softmax



```
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d_4 (Conv2D)           (None, 224, 224, 32)      896

 max_pooling2d_4 (MaxPooling  (None, 112, 112, 32)     0
 2D)

 conv2d_5 (Conv2D)           (None, 112, 112, 64)      18496

 max_pooling2d_5 (MaxPooling  (None, 56, 56, 64)       0
 2D)

 conv2d_6 (Conv2D)           (None, 56, 56, 96)        55392

 max_pooling2d_6 (MaxPooling  (None, 28, 28, 96)       0
 2D)

 conv2d_7 (Conv2D)           (None, 28, 28, 96)        83040

 max_pooling2d_7 (MaxPooling  (None, 14, 14, 96)       0
 2D)

 flatten_1 (Flatten)         (None, 18816)             0

 dense_2 (Dense)             (None, 512)               9634304

 activation_1 (Activation)   (None, 512)               0

 dense_3 (Dense)             (None, 26)                13338

=================================================================
Total params: 9,805,466
Trainable params: 9,805,466
Non-trainable params: 0
_________________________________________________________________
```

Figure 4: Model Summary using CNN

CONVOLUTION: Convolution is used to extract features from the input with the help of a convolutional layer. Filters are employed in feature extraction. Filters can alternatively be referred to as "masks," "feature detectors," and "kernels." For feature extraction in this convolution procedure, it passes a similar mask through the image. The kernel values in the picture that the mask covers that correspond to the values in the mask are multiplied. The total of the multiplied values, which is referred to as the response, yields the output pixel value that corresponds to the solution. The example demo is given below in figure 5.



1	0	0	0	0	1
0	1	0	0	1	0
0	0	1	1	0	0
1	0	0	0	1	0
0	1	0	0	1	0
0	0	1	0	1	0

6 x 6 image

1	-1	-1
-1	1	-1
-1	-1	1

Filter 1

Stride=1

3	-1	-3	-1
-3	1	0	-3
-3	-3	0	1
3	-2	-2	-1

Feature map

Figure5: Convolution

(1x1)+(0x(-1))+(0x(-1))+(0x(-1))+(1x1)+(0x(-1))+(0x(-1))+(0x(-1))+(1x1) = 1+0+0+0+1+0+0+0+1 which gives 3 as the value of a first pixel in the output(Feature map) when the 3x3 filter (Filter 1) is centered on the highlighted part of the input image (6x6 image) in figure 5. The stride in the figure represents the number of pixels the filter has to move over the input image during the operation. In our instance, 3x3 filters were employed. Convolution operations have been performed using a 2D convolutional layer, also known as conv2D.

POOLING: The convolutional operation is followed by a pooling layer. The feature map's overall efficacy is improved by lowering the feature map's dimensions. It is of two types. One is Max Pooling and the other is Average Pooling. We have used Max Pooling, which involves scanning an image with a filter and returning the highest pixel value at each instance as a separate pixel in a new image. In our case, we have used a 2x2 filter for Max Pooling.

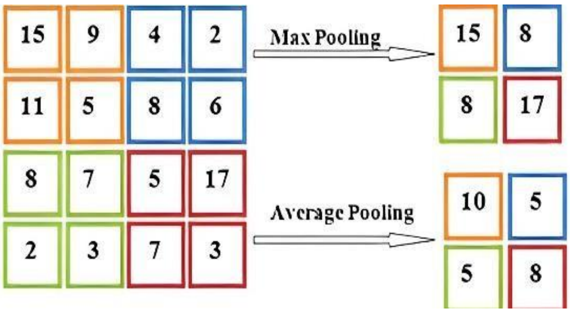


Figure6: Pooling

FLATTENING: After performing a polling operation, the resulting two dimensional arrays are flattened to create a single, extended, continuous linear vector.The fully linked layer receives this flattened value obtained by flattening as input to identify the picture (the dense layer).

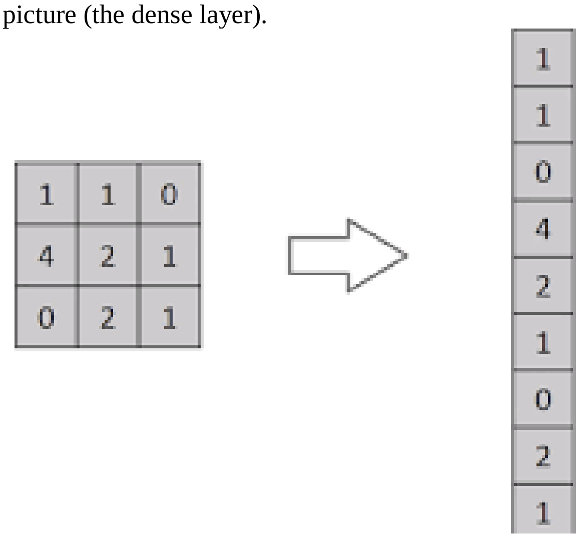


Figure7: Feature map is converted into linear vector

DENSE: In neural networks, fully-connected layers, also referred to as linear layers, are frequently employed. They link every input neuron to every output neuron. The input image is classified into a label using the fully connected layer. This layer links the data that was extracted in the earlier phases to the output layer, ultimately labelling the input according to the classification. So, we have given 26 as a parameter for dense layer in last as we have 26classifications(A-Z). The linking in fully linked layer is shown figure 8.

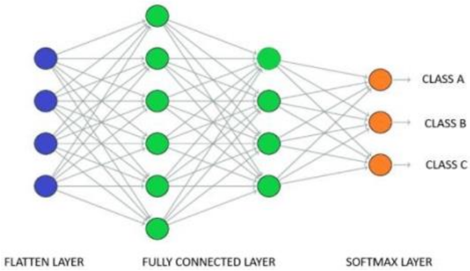


Figure8: Fully Linked Layer

ACTIVATION FUNCTION: The addition of this function to the network is intended to give neural networks nonlinear expression capabilities, allowing for better results and increased accuracy. Different activation mechanisms perform differently in various neural networks. In the above model, we have used two types of activation functions. One is ReLU, and the other is Softmax. Rectified Linear Unit, or ReLU, function may be used the most frequently for buried layers. We employed ReLU in hidden layers to solve the vanishing gradient issue and speed up computation performance. If the input is negative, the function gives zero, otherwise it returns x.

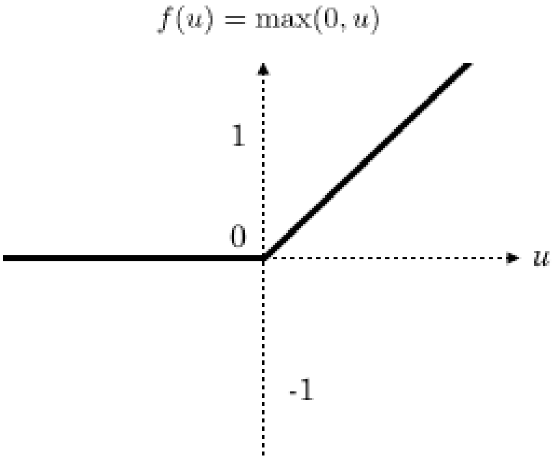


Figure 9: ReLU activation function

In Softmax, the neural network's unprocessed outputs are converted into a vector of probabilities by the softmax activation function, which performs probability distribution over the input. It is helpful in multiclass classification problems with N classes. In our case, it is helpful to assign probabilities to 26 classifications.

b) Mobile Net (A Pre-trained CNN Model)

Mobile Net is a CNN architecture that is efficient and adaptable for real-world uses. There are many Mobile Net variants that range in accuracy and computational complexity. The depth multiplier parameter of the original Mobile Net design, which controls the number of channels in each layer, allows for trade-offs between model size and accuracy. The 27 convolution layers of the Mobile Net model are composed of 13 depth wise convolutional layers, one average pooling layer, one fully connected layer, and one soft max layer.

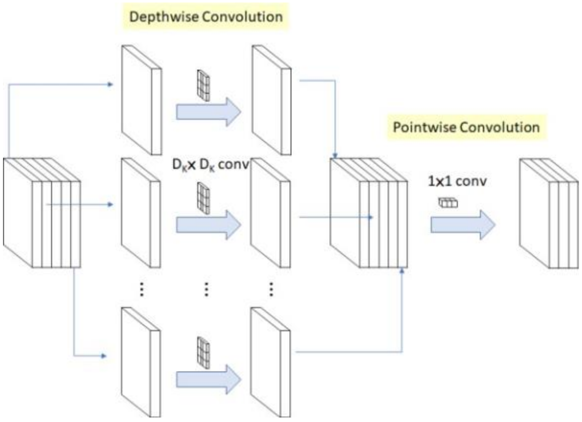
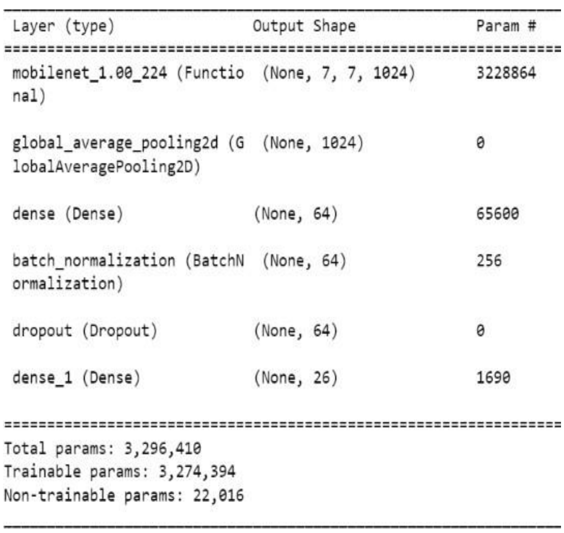


Figure 10: Depth-wise and Point wise convolution in Mobile Net

The depthwise separable convolutions that form the foundation of MobileNet's topology are divided into two primary operations: depthwise convolutions and pointwise convolutions. A single filter is applied to each channel of the input tensor in a depthwise convolution, whereas a set of 1x1 filters are applied to the output of a depthwise convolution in a pointwise convolution.This approach drastically lowers the amount of parameters and computational cost compared to traditional convolutions, while still preserving accuracy.The term depth-wise convolution refers to the channel-wise DK x DK spatial convolution.If there are five channels as in the figure10, we will have five DK x DK spatial convolutions. The 1 x 1 convolution in figure 10 used to adjust the dimension is called pointwise convolution.



```
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 mobilenet_1.00_224 (Functio  (None, 7, 7, 1024)       3228864
 nal)

 global_average_pooling2d (G  (None, 1024)             0
 lobalAveragePooling2D)

 dense (Dense)               (None, 64)                65600

 batch_normalization (BatchN  (None, 64)               256
 ormalization)

 dropout (Dropout)           (None, 64)                0

 dense_1 (Dense)             (None, 26)                1690

=================================================================
Total params: 3,296,410
Trainable params: 3,274,394
Non-trainable params: 22,016
_________________________________________________________________
```

Figure 11: Model Summary using MobileNet

A base model is created at first using Mobile Net with include_top=False for feature extraction. A 2-dimensional Global Average Pooling operation is next added, which entails averaging the feature maps from the previous convolutional layer to provide a single value for each feature map. This can then be fed into a fully connected layer. The output of a preceding layer is then normalised by adding Batch Normalization, which involves removing the batch mean and dividing by the batch standard deviation. After that, learnable parameters are used to scale and translate the normalised data.To avoid over fitting, the dropout layer is included. It happens when a model is too good at classifying training data, becoming overly specialised and unable to generalise to new, untried data.For picture categorization, a fully linked layer is inserted last.

IV. RESULTSANDDISCUSSIONS

While building the model, we used thirty percent of the samples for testing and seventy percent of the samples for training. In all, we tested 26 classes using 5445 photos, and we trained 12705 images. We used a batch size of 20 and 5 epochs for both models. Therefore, each epoch has 636 iterations. We used the categorical cross-entropy loss function . Adam(optimizer) is used to lower the losses by adjusting the neural network's weights and learning rate. The accuracy of both models during training and evaluation is shown below.

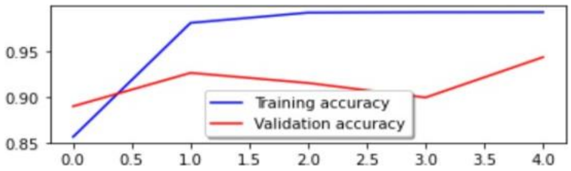


Figure 12: Accuracy for CNN

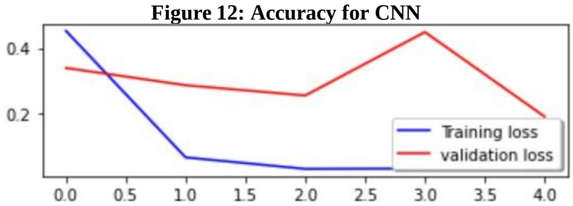


Figure 13: Loss for CNN

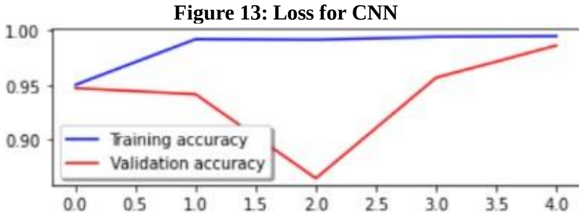


Figure 14: Accuracy for Mobile Net

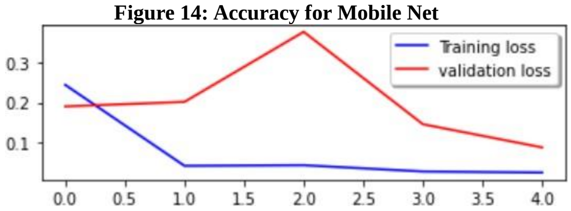


Figure 15: Loss for Mobile Net

As the saved model of Mobile Net's model provides more accuracy(98 percent) than cnn(94 percent), we used it for live prediction.. During image uploading, the recognised sign is converted into both text and audio. An object for Hand Detector is made for live prediction so that when the camera is opened, it will track the hand. The image was later cropped in order to find the ROI inside the specified limits. This hand that was identified is regarded as a test image. For live prediction on this test picture, we loaded our saved model and applied it.

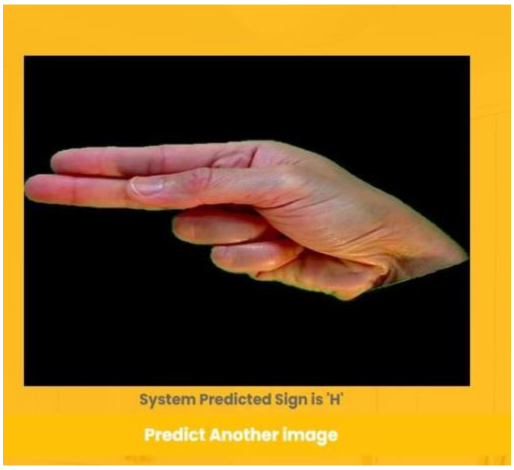


Figure 16: Prediction of ‘H’ during image uploading

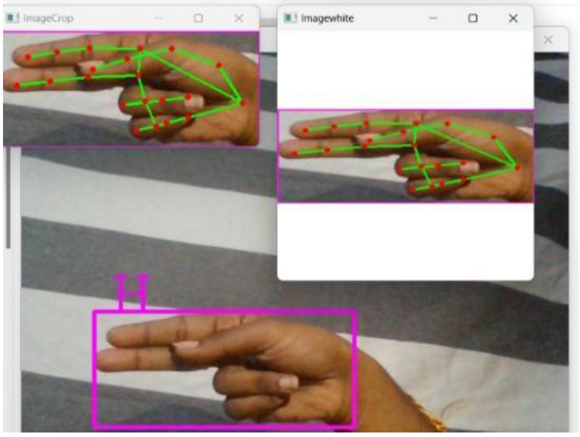


Figure 17: Live Prediction of ‘H’

V. CONCLUSION

Through the process of the whole project, we learned and found information about convolutional neural networks and image processing that we can use to classify images in the future. We used various built-in modules for our requirements. Based on the above results, we have observed that the pretrained model Mobile Net of CNN is working more effectively than the CNN that we have created by adding layers to the sequential model with the help of keras module. We can identify the accuracy and loss of training and validation data for each epoch during the training phase. Finally, the model predicts the gesture, which is then shown on the screen through the provided graphical user interface. We come to the conclusion that using excellent lighting and a better camera for good performance can improve the prediction.

REFERENCES

[1] D. Abdulla, S. Abdulla, R. Manaf and A. H. Jarndal, "Design and implementation of a sign-to-speech/text system for deaf and dumb people," 2016 5th International Conference on Electronic Devices, Systems and Applications (ICEDSA), Ras Al Khaimah, United Arab Emirates, 2016, pp. 1-4, doi: 10.1109/ICEDSA.2016.7818467.

[2] "Sign language recognition using sensor gloves," Proceedings of the 9th International Conference on Neural Information Processing, 2002, ICONIP '02., Singapore, 2002, pp. 2204-2206 vol.5, doi: 10.1109/ICONIP.2002.1201884.[3] T. Y. Ceiang and M. J. Hsiao, ―Carry-select adder using single ripplecarry adder,‖ Electron. Lett., vol. 34, no. 22, pp. 2101–2103, Oct. 1998.

[3] Kaslay, Siddharth, Tejal Kesarkar, and Kanchan Shinde. "ASL Gesture Recognition using Various Feature Extraction Techniques and SVM." International Research Journal of Engineering and Technology (IRJET), vol. 7, no. 6, June 2020, pp. 3956. ISSN: 2395-0056.

[4] Apoorva M A, Harshitha M S, Chaitra S, Akshitha V, and Rajath A N. "An Efficient and Robust System for Hand Gesture Recognition and Interpretation." International Journal of Engineering Research & Technology (IJERT), vol. 6, no. 7, 2017, pp. 398-402.[5] J. M. Rabaey, Digtal Integrated Circuits—A Design PerspectiveUpper Saddle River, NJ: Prentice-Hall, 2001.

[5] Nagashree, R. N., Stafford Michahial, Aishwarya G. N., Beebi Hajira Azeez, Jayalakshmi M. R., and R Krupa Rani. "Hand Gesture Recognition using Support Vector Machine." The International Journal Of Engineering And Science (IJES), vol. 4, no. 6, June 2015, pp. 42-46. ISSN (e): 2319 – 1813, ISSN (p): 2319 – 1805.

[6] A. Saxena, D.K. Jain, and A. Singhal. "Sign Language Recognition Using Principal Component Analysis." Proceedings of the 2014 International Conference on Advances in Electronics, Computers and Communications (ICAECC), pp. 1-4, 2014.

[7] "Sign language identification," 2016 3rd International Conference on Recent Advances in Information Technology (RAIT), Dhanbad, India, pp. 422-428, doi: 10.1109/RAIT.2016.7507939, by A. Kumar, K. Thankachan, and M. M. Dominic.

★ ★ ★