EasyChair Preprint

№ 2957

American Sign Language recognition using Convolution Neural Network for Raspberry Pi

Aashish Thapa Magar and Pramod Parajuli

March 14, 2020

American Sign Language recognition using Convolution Neural Network

Aashish Thapa Magar[1], Pramod Parajuli[2] (Supervisor)

ABSTRACT

Hand gestures represent a vast amount of information that can be used for basic communication by people with disabilities as well as augment communication for others. As the information of hand gestures rely on movement sequences, identifying hand gestures with accuracy in real-time is challenging. In the domain of human-computer interaction, hand gesture recognition models have been developed for mouse pointer movement, playing games with specific actions etc. Various techniques such as HOG transforms, SIFT, BRIEF, ORB have been used to identify region of interest and for classifying the region of interest techniques such as Support Vector Machines (SVMs), Hidden Markov Models (HMMs) etc. are being used. These methods demand heavy computational resources. This paper presents a novel method for recognizing American Sign Language (ASL) using image pre-processing methods and a Convolution Neural Network (CNN) for classification that is implemented in Raspberry Pi3. Images captured from the Raspberry Pi camera module are pre-processed for better clarity and region of interest isolation so that better set of features are extracted. These features are then fed into CNN for classification. Executing the model on Raspberry Pi3 has resulted into a satisfactory output as the classification result and time taken by the system has been acceptable to end-users.

1. INTRODUCTION

Hearing-impaired people are facing problem to interact with others causing communication gap. The only communication mechanism to deliver their thoughts and their feeling are by using the hand sign which is not understandable by many others due to which communication with those who do not understand hand signs is not possible. Therefore, as a solution to the problem, hand gesture recognition system has been the highly anticipated topic. Many researches have been performed to solve the issues but most of them requires input devices like colored globes or the sensors which can add extra resource requirements.

Moreover, to reduce the communication gap between the hearing-impaired people and normal people, there are two possible ways either every people should learn hand gesture-based communication or create a system that can translate the hand gestures to normal text readable to most humans.

There are mainly two types of gestures; first, static gesture and second, dynamic gesture. Static gesture is the gesture where the stationary hand or the images are used for attaining the information whereas the dynamic gesture is a gesture with moving hands or the sequences of the images are used for revealing information. The computing resource required for processing

1 Official4aashish@gmail.com

2 pramodparajuli@gmail.com

the static gesture is relatively lower than for the dynamic gesture as the static gesture uses a single posture whereas the dynamic gesture requires the sequence of postures for processing. The hand gesture recognition system can be used for multiple purposes. It can also be used as the translator to translate the gesture to the texts and can be used for human-computer interaction to ease the interaction between the human and the machines.

This paper is focused on developing static gesture recognition system which can translate the hand gesture to text-based output.

1.1. Objectives

- To develop the model for hand gesture recognition system using an open-source library (Open CV) and Convolutional Neural Network.
- To develop the prototype of a hand gesture recognition system that can be used to recognize the gesture and generate text-based output.

1.2. Detail Objectives

- To implement an Open CV pre-built function for image pre-processing.
- To implement Convolutional Neural Network model for training and classification.
- To develop prototype of the proposed system on Raspberry Pi3 with a camera module.
- To evaluate the working system for end-user acceptance.

1.3. Problem Statement

Natural interaction has been most desirable for interaction with the computer but the computer is unable to decode it. In the case of the hand gesture recognition system, the computer vision should be able to distinguish the hand from other objects. The next task is image classification. Many researches perform image classification and object detection by using high-cost equipment like Kinect sensor, depth camera, colored globes, etc. Besides the use of the costly hardware other researcher has developed the image pre-processing methods which are highly dependent on the learning environment. Another crucial problem for the development of the hand gesture recognition system is object localization. Unstable objects also make it difficult for object localization. Therefore, the gesture recognition systems would be unable to respond during the change of one gesture posture to another gesture posture.

1.4. Scope

The proposed hand gesture recognition system is intended to recognize the simple static American Sign Language and translate into text-based output.

2. LITERATURE REVIEW

The hand gesture recognition system is a common field of studies for human-computer interaction purposes. As mentioned in the problem statement, one of the problems for the gesture recognition system is hand tracking. To solve it, input materials like colored globes for tracking hand and locating the fingers are being used. Tracking hand can be a difficult job due to the complex background and shadows and the system is unable to segment the color (Mutha et al., 2015). Skin coloring model is another model to address the problem of hand tracking and eliminate the input-based gesture system (Hasija et al., 2014).

Image pre-processing is the most important step while working with computer vision. Image pre-processing is the operation performed on images for the lowest level abstraction which the

main purpose is to improve or enhance some feature of image for further processing and analysis task.

Image classification requires multiple steps; first, image capture/acquisition step to capture an image through multiple sources such as camera, file stream etc. Then second step to perform filtering, smoothing, color conversion, binary conversion, etc. the captured image. Then in third step, feature is extracted. Then finally image is classified using Artificial Neural Networks (ANNs).

In addition to hand-tracking, image segmentation has been another problem to be solved during the development of the hand gesture recognition system. Efficient hand tracking and segmentation are the must feature sfor the development of a hand gesture recognition system to work properly. The extraction of raw postures and gesture data for recognition in the globe-based gesture system requires the globe to be attached to the computer and users are must wear the globe (Pradipa et al, 2014). Histogram-based technique has been quite popular approach for image processing. In this approach, the vector is used on the basis of the orientation histogram (Patel et al., 2018). The histogram orientation methods follow the pattern recognition and use black and white-colored images for the digitization of the image. The detection and compression of the gesture are eased by the digitization of the image and the pattern recognition.

Image classification is introduced to reduce the gap between human vision and computer vision by training the model with the data. Differentiating the image into the prescribed category based on the content of the vision helps to achieve the image classification (Neelima et al., 2018). Deep learning method can be best for image classification as the machine learning consists of features extraction module to extract the important features such as edges, texture, etc. and classification module can classify on the basis of the features extracted.

ASL is one of the hard sign languages where multiple gestures look similar to the other. Though the human eye can easily distinguish the gesture it is hard for a computer model to easily distinguish the gesture.

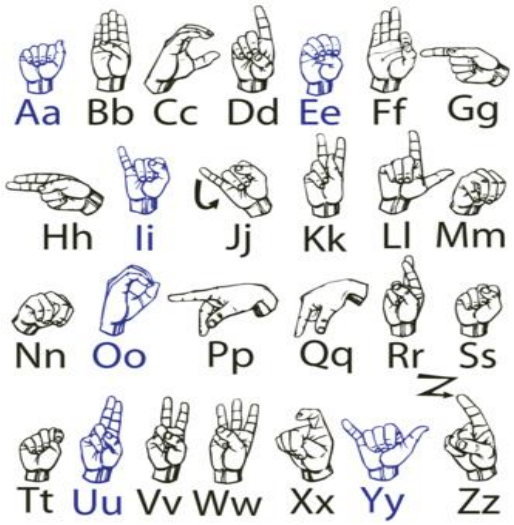

Figure 2.1: American Sign Language Gesture and Characters

2.1. Research Gap

Though there is much research that is related to the hand gesture recognition system, most of them require special input devices to be attached to the computer like globe-based system. The proposed hand gesture recognition system will be addressing the problem where the special input devices will not be required for the recognition of the hand gesture and will be able to recognize the static gesture and provide text-based output.

3. RESEARCH METHODOLOGY

Initially, the user has to provide static gesture input. The gesture will be captured by the camera module and will be transferred to the internal process. During the internal process, the received image will be processed studying the image color depth, the histogram. Then the image is shaped out to a vector to generate features. The generated features will be used to train and classify the image. Corresponding label of a class will be the text-based output of the system. The architecture of the proposed system is depicted in figure 3.1 and the process is outlined in figure 3.2.

3.1. Proposed model

The proposed system is intended to run on Raspberry PI and will be using the Raspberry pi camera module for capturing the image. The system captures the image using the camera module and the image will be processed for the feature extraction and is stored in the feature set. For the generation of the feature set, the system will be using the CNN model. The extracted feature will be classified and the gesture is recognized and the text is output.

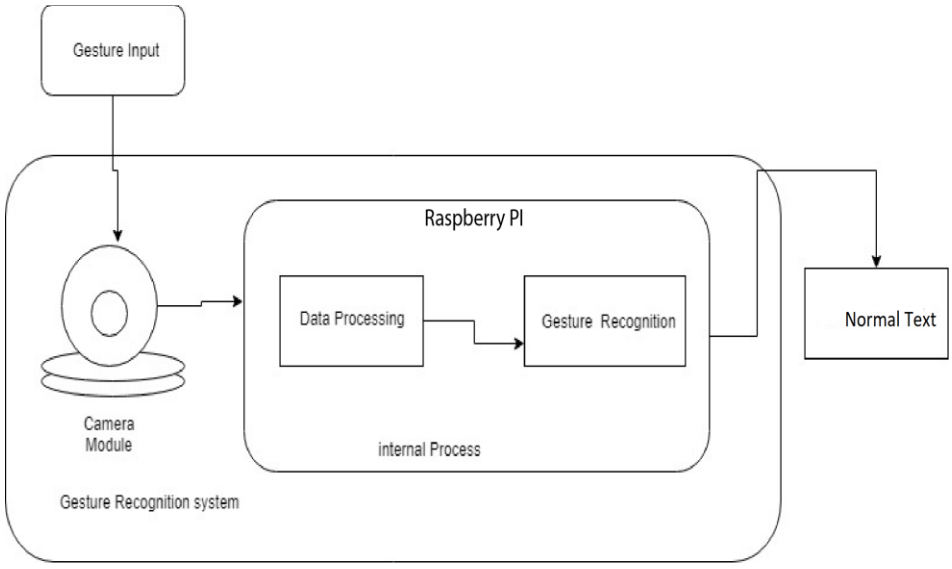

Figure 3.1: Schematic Diagram of Hand Gesture Recognition System

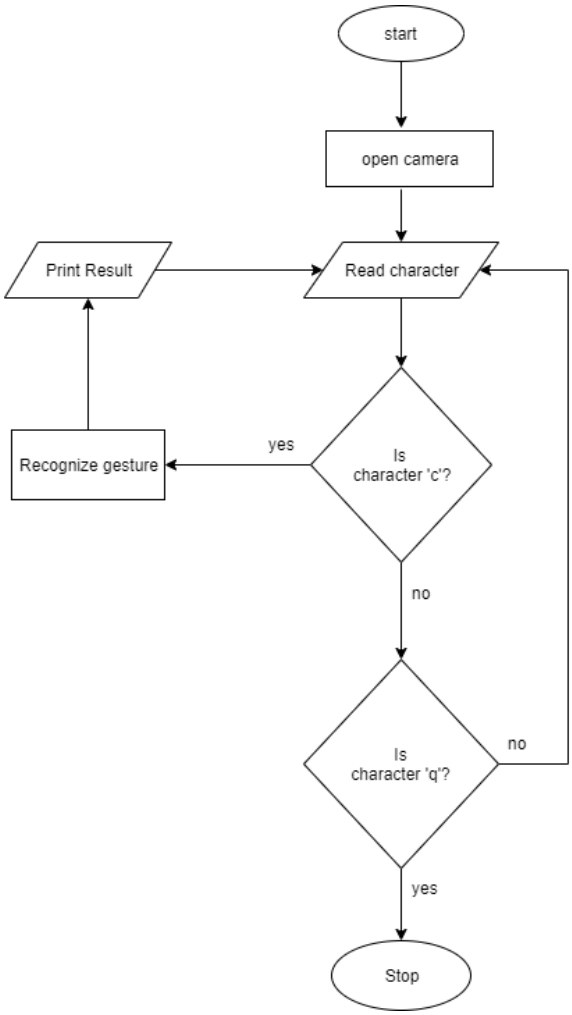

Figure 3.2: Flowchart of recognition system

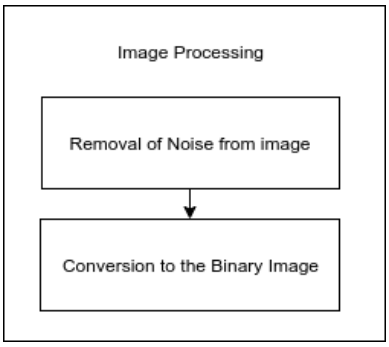

Figure3.3: Image Processing Section

During Image Processing step two main tasks take place they are: -

1. Removal of noise from the image and
2. Conversion to the binary image.

Removal of noise is done by using Gaussian filter as shown in eq(i).

$$G(x,y) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{\frac{-x^2+y^2}{2\sigma^2}} \dots\dots\dots\dots (i)$$

where x is the distance from the origin in the horizontal axis, y is the distance from the origin in the vertical axis, and σ is the standard deviation of the Gaussian distribution.

The conversion of the image to binary is done by *thresholding process as shown in* eq(ii).

$$T = T[x, y, p(x,y), f(x,y)] \dots\dots\dots\dots\dots (ii)$$

where, T is the threshold value, x and y are the coordinates of threshold value point, p(x, y) and f(x, y) is the gray level image pixels.

The CNN model consists of two basic parts of feature extraction and image classification. Feature extraction consists of multiple layers followed by the *max-pooling* and *ReLU activation* function. In convolution layer, the input image is filtered which is extracted by scanning a certain portion of the image which is 3 pixels by 3pixels in dimension. The output of element-wise multiplication forms the feature map. This step takes place until the whole image is scanned.

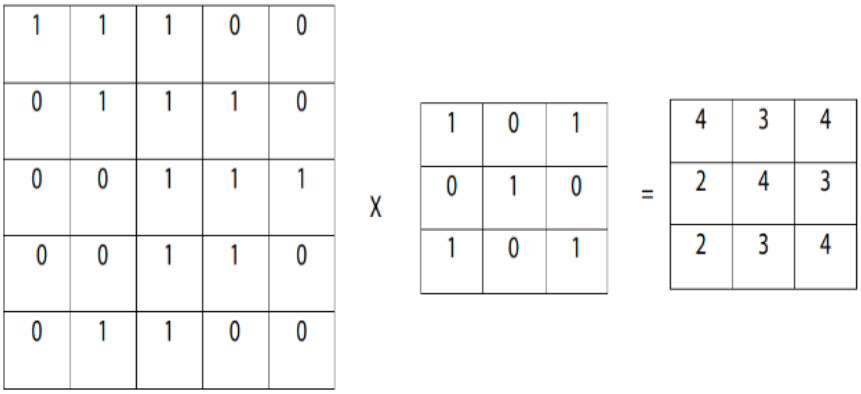

Figure 3.4: Convolutional Layer

At the end of the convolution operation, the output is subjected to the activation function for non-linearity which is also known as the ReLU activation function. At this phase, all the negative values are replaced by zero. After the steps of the ReLU the max-pooling steps take place which the main task is to reduce the dimension of the input image. The final layer of CNN is the fully connected ANN. The main goal of the ANN is to analyze the features of the input and combine them into different attributes that performs image classification.

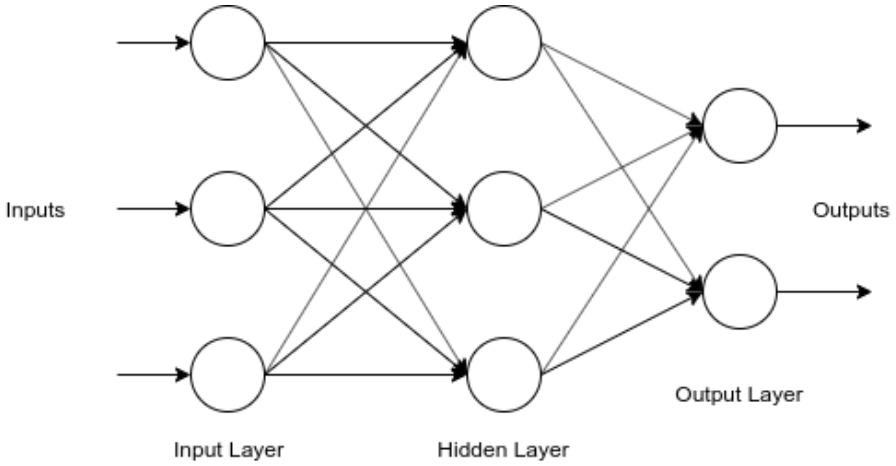

Figure 3.5: Fully / Densely Connected Layer

Simple CNN model network image classification has the architecture of [INPUT - CONV LAYER - RELU - POOL - FC]. The most common CNN follows the pattern:

INPUT -> [[CONV -> RELU] *N -> POOL?] *M -> [FC -> RELU] *K -> FC

where

** represents the repetition,*

POOL? Represents the optional pooling layer and

K, M, N are the constant value that is >= 0(K, N values is usually <= 3).

3.2. Data Collection

The proposed hand gesture recognition system is focused to recognize the static gesture, to train the system and to classify gesture the dataset is required. For the purpose of training ASL data-set from National Center for Sign Language and Gesture Resources (NCSLGR) is used that is available at *http://www.bu.edu/asllrp/ncslgr.html.*

4. IMPLEMENTATION AND EXPERIMENTS

4.1. Hardware and Software

The proposed system uses Raspberry Pi, camera module, and workstation computer. The system is programmed in python language with version 3 and multiple libraries like Open CV, NumPy, Matplotlib, tensorflow and keras are used for data processing, model building and testing. Raspberry Pi is configured with Raspbian OS that supports python 3+.

4.2. Training and generating Convolutional Model

CNN is used as an artificial neural network to train the system with the dataset of ASL. The dataset consists of a variation of the same gesture which will train the system to get more accuracy with variations of the same gesture. This helps to recognize gestures in various conditions. The features of each image are extracted and stored for future use in image classification process.

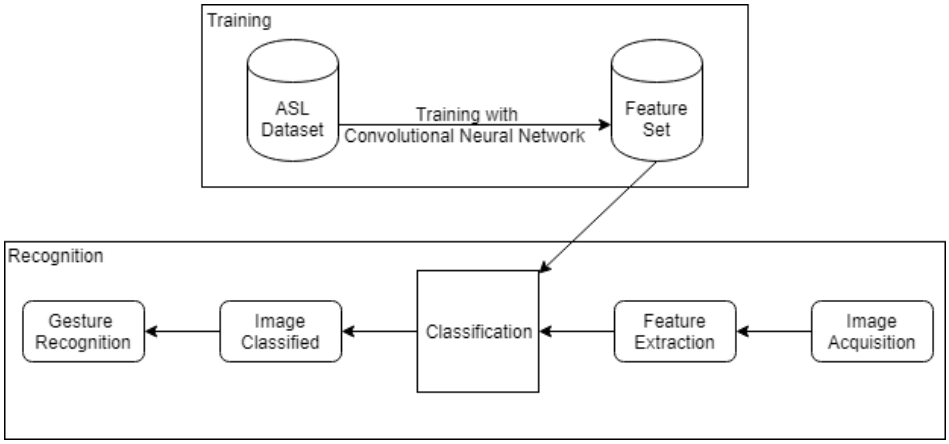

Figure4.1: Training and Image Classification steps

The model was trained with features extracted for 25 iterations/epoch with ReLU and SoftMax activation functions. ReLU activation function is used for non-linear activation function and is the most used activation function. The main task of the ReLU activation function is to eliminate the negative input and make it 0 and does not change the positive input value.

$$f(x) = max(0, x) \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots (iii)$$

The SoftMax activation function is used to turn the input numbers to the probability to be in a certain class which ranges from 0 to 1.

$$\sigma(z)_j = \frac{e^{z_j}}{\sum_{k=1}^{K} e^{z_k}} \dots\dots\dots\dots\dots\dots\dots\dots\dots (iv)$$

where, z is the vector of input to the output layer and j indexes the output units, so j = 1, 2, …, K.

4.3. Features of the proposed system

The features of the proposed **Hand Gesture Recognition System** are: -

1. The proposed system is built using the python3 programming language which makes the system operating system independent i.e. the system can run on any operating system that is able to run the python 3 and the required libraries Keras, Tensorflow, NumPy, Pandas, Matplotlib and OpenCV.
2. The proposed system is able to recognize the simple static gesture of the ASL alphabets.
3. The proposed system is modular and can be easily updated or upgraded.

4.4. Testing and Evaluation

The model is trained and tested for accuracy and the loss. The model was tested with the characters dataset of the ASL and was evaluated during the process of training. Figure 4.2 shows the results.

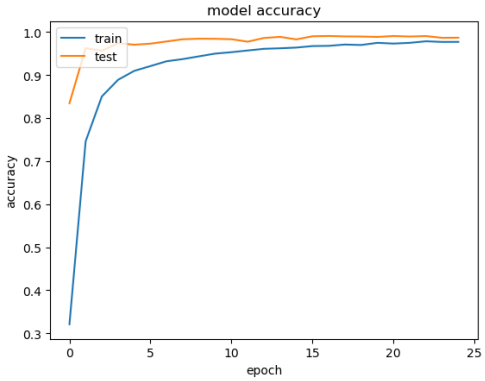

Figure 4.2(a): Model Accuracy

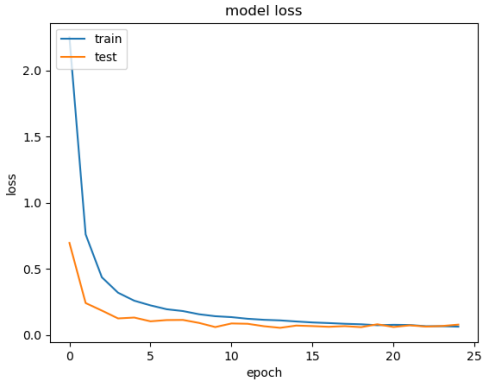

Figure 4.2(b): Model Loss

The model was also tested in real-life environment in indoor settings. Figure 4.3 shows the outcome with predicted text.

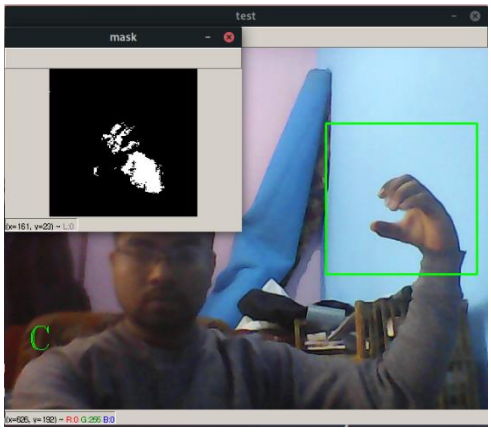

Figure 4.3(a): ASL character C detection

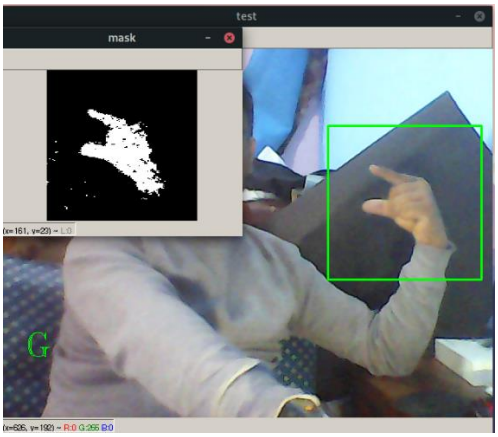

Figure 4.3(b): ASL character G detection

4.5 Results

The proposed Hand Gesture Recognition System uses ASL alphabets gesture datasets to train the model. The dataset is divided into two sets one is training and the other is testing sets. Each set contains a threshold image of each character from A to Z which is 26 characters. The training set contains the training image of 1750 and a testing set contains 250 images.

Table 4.1: Training Accuracy and Loss Result of multiple Algorithm

Model	**Average Accuracy**	**Average Loss**
CNN	98%	2%
SVM	92%	8%
HECS	97%	3%
CaffeNet	88%	12%

SAE-PCA	93%	7%

Comparing multiple neural networks, table 4.1 shows the training accuracy and the loss of the particular model. CNN model shows high level of training accuracy and low loss which states CNN to be the best model for image classification on the dataset.

Testing of the system with real-life gesture input in two different lighting condition one with bright lighting conditions and with the single-colored background and other with the normal lighting condition was also conducted. For both configurations, the number of inputs were 10 different gesture.

Table 4.2: Testing System in good lightning condition

S.N.	Sample	Result
1	M	Not Recognized
2	E	Not Recognized
3	I	Recognized
4	H	Recognized
5	G	Recognized
6	S	Not Recognized
7	Y	Recognized
8	L	Recognized
9	C	Recognized
10	V	Recognized
Accuracy		70%

Table 4.3: Testing System in average lightning condition

S.N.	Sample	Result
1	M	Not Recognized
2	E	Not Recognized
3	I	Not Recognized
4	H	Recognized
5	G	Not Recognized
6	S	Not Recognized
7	Y	Recognized
8	L	Recognized
9	C	Recognized
10	V	Recognized
Accuracy		50%

For real-life environment testing, the accuracy rate was higher for environment with good light. The system was able to classify 70% of gesture whereas while the object was not exposed to normal lightning condition the system was able to classify the gesture with an accuracy of 50% only.

5. CONCLUSION AND RECOMMENDATIONS

5.1 Conclusion

An American Sign Language recognition system using Convolution Neural Network for Raspberry Pi was developed using implementing CNN model. It was able to recognize the simple static ASL gestures with accuracy above 96% and classify real-world environment image with accuracy above 70% in good lighting conditions.

5.2 Recommendations

The system can be enhanced to work more accurately and extra features like dynamic gesture recognition can be added in further future. As for more accuracy, the system requires good lightning conditions which can be enhanced by using calibration methods. The system can also be enhanced by adding up features to add more datasets.

As for the future, the system recently can recognize the ASL character only and can be enhanced to recognize the digits and the dynamic gesture too. The system particularly requires good lighting condition to recognize the gesture and can be enhanced by using a depth-sensing camera which helps and allows to recognize the 3D gesture and the motion gesture.

ACKNOWLEDGEMENTS

I would like to express my special thanks of gratitude to Dr. Pramod Parajuli, who supervised me for the completion of the project. Secondly, I would like to express my special thanks of gratitude BIT Program Coordinator, Mrs. Sarita Neupane, who helped me in doing a research.

REFERENCES

Ahmed, T. (2012). A Neural Network based Real-Time Hand Gesture Recognition System. International Journal of Computer Applications, 59(4), pp.17-22.

Andrew.gibiansky.com. Available at: http://andrew.gibiansky.com/blog/machine-learning/convolutional-neural-networks/ [Accessed 12 Aug. 2019].

AuthorCafe. (2019). Summer Research Fellowship Programme of India's Science Academies 2017. [online] Available at: https://edu.authorcafe.com/academies/6813/sign-language - recognition [Accessed 12 Aug. 2019].

Barczak, _ et al. (2011). A New 2D Static Hand Gesture Colour Image Dataset for ASL Gestures. Res Lett Inf Math Sci. 15.

Gibiansky, A. (2019). Convolutional Neural Networks - Andrew Gibiansky. [online]

Guru99.com. (2019). TensorFlow Image Classification: CNN (Convolutional Neural Network). [online] Available at: https://www.guru99.com/convnet-tensorflow-image-classification.html [Accessed 12 Aug. 2019].

Keras.io. (2019). Convolutional Layers - Keras Documentation. [online] Available at: https://keras.io/layers/convolutional/ [Accessed 26 July. 2019].

Khan, R.Z. (2012). Hand Gesture Recognition: A Literature Review. International Journal of Artificial Intelligence & Applications, 3(4), pp.161-174.

Labhane, et al. (2012). Multipoint Hand Gesture Recognition for Controlling Bot. International Journal of Scientific Research, 1(1), pp.46-48.

Li, L. & Zhang, L. (2012). Corner Detection of Hand Gesture. TELKOMNIKA Indonesian Journal of Electrical Engineering, 10(8).

Neelima, et al. (2018). Image classification using Deep learning. International Journal of Engineering & Technology, 7(2.7), p.614.

N, U. & I.R, D. (2016). Development of an Efficient Hand Gesture Recognition system for human-computer interaction. International Journal of Engineering and Computer SciModelence.

Oyedotun, et al. (2016). Deep learning in vision-based static hand gesture recognition. Neural Computing and Applications, 28(12), pp.3941-3951.

Roomi, _ (2010). Hand Gesture Recognition for Human-Computer Interaction. Journal of Computer Science, 6(9), pp.1002-1007.

Shutterstock.com. (2019). Asl Images, Stock Photos & Vectors | Shutterstock. [online] Available at: https://www.shutterstock.com/search/asl [Accessed 12 Sep. 2019].

Traore, et al. (2018). Deep convolution neural network for image recognition. Ecological Informatics, 48, pp.257-268.

Zhang, et al. (2013). Robust Hand Gesture Detection Based on Feature Classifier. Advanced Materials Research, 823, pp.626-630.