"SIGN LANGUAGE RECOGNITION USING CNN"

A Major Project Report

Submitted in partial fulfilment of requirement for the award of Degree of

BACHELOR OF INFORMATION TECHNOLOGY ENGINEERING

Submitted to

POKHARA UNIVERSITY

Submitted By: -

ALISH DHAMALA [160102]

AVINASH GHIMIRE [160108]

KESHAV POUDEL [160146]

PAWAN KUMAR THAPA [160124]

PRAJWOL LAL MULEPATI [160125]

Under the Guidance of

Er. ABHISHESH DAHAL

DEPARTMENT OF INFORMATION TECHNOLOGY AND COMPUTER ENGINEERING

since 2001

COSMOS COLLEGE

OF MANAGEMENT AND TECHNOLOGY

Affiliated to Pokhara University

Tutepani, Lalitpur, Nepal

01 August 2021

since 2001

COSMOS COLLEGE

OF MANAGEMENT AND TECHNOLOGY

Affiliated to Pokhara University

DEPARTMENT OF INFORMATION TECHNOLOGY AND COMPUTER ENGINEERING

CERTIFICATE

This is to certify that the major project entitled **"SIGN LANGUAGE RECOGNITION USING CNN"**, submitted by **Alish Dhamala [160102], Avinash Ghimire [160108], Keshav Poudel [160146], Pawan Kumar Thapa [160124] and Prajwol Lal Mulepati [160125]** of B.E Final Year, IT Engineering have completed their major project under the guidance and supervision.

I approve the project for submission for the partial fulfilment of requirement for the award of Degree Bachelor of Engineering in Information Technology under Pokhara University.

Guided & Approved By:

Er. Abhishesh Dahal

(Project Supervisor)

Department of IT & Computer Engineering

Cosmos College of Management & Technology, PU

Er. Bibek Rupakheti

(Head of Department)

Department of IT & Computer Engineering

Cosmos College of Management & Technology, PU

Er. _____

(External)

Date of Approval: 01 August 2021

COPYRIGHT

Department of Information Technology and Computer Engineering
Cosmos College of Management and Technology, Pokhara University
Tutepani, Lalitpur, Nepal

ACKNOWLEDGEMENT

Co-ordination and teamwork are the basics of achieving success which an individual is incapable of getting single handedly. We feel great pleasure to acknowledge the assistance and contribution of various individuals who have co-operated and helped us to come up with this project. It is a matter of great pleasure for us to have the opportunity of carrying out this major project report on **"SIGN LANGUAGE RECOGNITION USING CNN"**.

First and foremost, we would like to express our sincere gratitude to our supervisor, **Er. Abhishesh Dahal** for his excellent guidance and encouragement. Similarly, we would like to acknowledge our HOD, **Er. Bibek Rupakheti** for his significant support, guidance, and suggestion during a meticulous and tireless shepherding process. We indebted to them.

Our special thanks goes to Vice Principal, **Er. Yuba Raj Siwakoti**; and **Er. Ankit Bhattarai** for his valuable suggestion which is truly appreciated. This project has come out in this form due to the timely encouragement and high valued suggestions given to our team.

We welcome any constructive critics and will be grateful to any appraisal.

Sincerely,

ALISH DHAMALA [160102]

AVINASH GHIMIRE [160108]

KESHAV POUDEL [160146]

PAWAN KUMAR THAPA [160124]

PRAJWOL LAL MULEPATI [160125]

ABSTRACT

As the advancement of the technology all around the world is increasing, the use of Human Computer Interaction (HCI) has improved day by day. Computer vision plays an important role to provide information to design more simple and efficient approach to HCI. This work targets real-time recognition of both static hand-poses and dynamic hand-gestures using Convolution neural network (CNN).

Sign language is the preferred method of communication among the deaf and the hearing-impaired people all over the world. Recognition of sign language can have varying degree of success when used in a computer vision or any other methods.

Our project aims at extending a step forward in this field by collecting dataset and then use various feature extraction techniques to extract useful information which is then input into various supervised learning techniques.

Key Words: HCI, CNN, Dataset, Sign Language, Extraction.

TABLE OF CONTENTS

LIST OF FIGURES 8
LIST OF ABBREVIATION 9
1. INTRODUCTION 10
1.1 Background 10
1.2 Problem Statement 11
1.3 Objectives 11
1.4 Scope 12
1.5 Feasibility Analysis 12
1.5.1 Economic Feasibility 12
1.5.2 Technical Feasibility 12
1.5.3 Operational Feasibility 12
1.6 Significance of Study 13
2. LITERATURE REVIEW 14
2.1 Related Works 14
2.2 Image Acquisition 16
3. METHODOLOGY 17
3.1.1 Image Segmentation 17
3.1.2 The Convolutional Neural Network Model 18
3.2.3 Prediction 20
3.3 Software Development Model 21
3.3.1 Incremental Model 21
3.4 Work Schedule 22
3.5 Working Mechanism 23
3.6 Requirements 24
3.7 Tools Used: 24
3.8 Technology Used: 25
4. SYSTEM DESIGN 26
4.1 System Block diagram 26
4.2 Flowchart 27
4.3 Use Case Diagram 28
4.4 Sequence Diagram 29
5. RESULT AND DISCUSSION 30
5.1 Limitations 30

5.2 Problems Faced 30
6. SCREENSHOOTS 31
6.1 Image Capturing 31
6.2 Collected Data 31
6.3 Trained Dataset 32
6.4 Accuracy of the Model 33
6.5 Sample Outputs 34
7. CONCLUSION 36
8. REFRENCES 37

LIST OF FIGURES

Figure 1 : ASL Dataset 16
Figure 2: Image segmentation [8] 17
Figure 3: Systematic diagram of basic CNN [9] 18
Figure 4: flow of our CNN Model 20
Figure 5: Incremental Model [10] 21
Figure 6: Gantt Chart 22
Figure 7: Data Flow Diagram 23
Figure 8: System Block diagram 26
Figure 9 : Flow Chart 27
Figure 10: Use Case Diagram 28
Figure 11: Sequence Diagram 29
Figure 12: Image capturing Sample 31
Figure 13: Collected Data sample 31
Figure 14 : Training Dataset 32
Figure 15: Training and Testing sample 32
Figure 16: Accuracy score 33
Figure 17: Sample Output-I 34
Figure 18: Sample Output-II 34
Figure 19: Sample Output-III 35

LIST OF ABBREVIATION

HCI - Human Computer Interaction

ASL - American Sign Language

ISL – Indian Sign Language

BSL- British Sign Language

NSL – Nepali Sign Language

CNN - Convolution Neural Network

ANN – Artificial Neural Network

RNN - Recurrent Neural Network

FNN - Feedforward Neural Network

MLP - Multi-Layer Perceptron

UI – User Interface

OpenCV – Open-Source Computer Vision Library

1. INTRODUCTION

1.1 Background

Sign language is a basic mode of communication between people who have difficulty in speech and hearing. Sign language are a set of languages that use predefined actions and movement to convey a message to others. In this disabled people use a simultaneous and precise movement of hands, hand shapes etc. However, normally people do not have much idea about sign language and find it hard to communicate with sign language user, which can cause confusion and demands for translator. In Nepal [1], prevalence of the deafness is about 2.8% which accounts to about 0.7 million population.

Around the world several efforts have been made to translate sign language to English and other languages. There are many apps available for American Sign Language (ASL), British Sign Language (BSL), and Indian Sign Language (ISL) etc. However, there hasn't been significant progress in case of Nepali Sign Language (NSL).

During the past years, efforts have been made to achieve more natural interactions between human and computer communication. So, this project is an initiation to translate the sign used by the people of Nepal in words understood by others and help to solve the problem of communication gap that people are facing now.

The problem we are investigating is sign language recognition through unsupervised feature learning. Being able to recognize sign language is an interesting computer vision problem while simultaneously being extremely useful for deaf people to interact with people who don’t know how to understand American Sign Language (ASL).

1.2 Problem Statement

Worldwide [2], at least 466 million people are likely to be deaf and mute (or simply deaf). Most of the people do not have access to public services. In the case of hearing impairments, mutism has created the need of using sign language. However, most of the non-disabled people do not know this type of language. Therefore, for deaf people performing daily activities turns out to be difficult, especially in public areas.

Additionally, it is difficult and expensive to make non-disabled people learn sign language. For these reasons, automatic systems that translate sign into text are required.

The problem statement revolves around the idea of a camera-based sign language recognition system that would be in use for the deaf for converting sign language gestures to text and then text to speech. Communication for the majority of people is not difficult. It should be the same way for the deaf. Understanding the exact context of symbolic expressions of deaf people is the challenging job in real life until and unless it is properly specified.

1.3 Objectives

Main Objective:

1. The main objective of this project is to recognize the gestures and displaying the correspondent word in real time.

Specific Objective:

1. The purpose of our project is to eliminate the barrier between the deaf, mute and rest of the people.
2. The purposed system will be able to recognize static and dynamic hand gestures.
3. This system can learn to classify the specific gestures pattern of any person.
4. Create user friendly UI to keep it simple.

1.4 Scope

The main area where Sign Language Recognition System can be used is in public places like parks to interact with other people, ticket issuing counters, hospitals etc., basically in almost every public place where interaction is needed. This system can be even used to train the normal people, where they can practice the symbols/gestures.

Recognizing hand gestures will widen the training set for the machine.

1.5 Feasibility Analysis

1.5.1 Economic Feasibility

Software requires no development costs whatsoever, as during operation we would only require a server setup and use of inbuilt camera. On a total analysis of cost, the project was economically feasible.

1.5.2 Technical Feasibility

The system will require less maintenance as the system is self-sustainable and sometimes may require some changes overtime. The system uses the latest emerging technology that solves communication gap in the real world. The tools necessary for the project are open source and have proper documentation that will guide us through the project.

1.5.3 Operational Feasibility

The system will include a user-friendly operational environment in terms of UI, so that even the users with basic knowledge of androids can have convenience while using it. Each feature will be easily accessible in the software, thus making it operationally feasible.

1.6 Significance of Study

The study investigates the need of a system, i.e., camera-based sign language recognition which translates the hand gestures into American alphabets. This helps us to utilize the current technology in a very significant way. Studying ASL promotes better awareness of and sensitivity to the deaf and hard of hearing community.

By the study we found out that there are not many existing sign language recognition systems in the context of Nepal. So, we thought it would a very useful system for the differently able people to communicate with other people very commonly as there will be less communication gap. It uses very less amount of resources so we thought it would be very significant to develop such system.

The history of sign language has an interesting past, being the first form of communication in early man. Sign language then went on to help end the discrimination of deaf people and helped the deaf to become educated like their hearing peers.

2. LITERATURE REVIEW

2.1 Related Works

There are a lot of research done in the field of sign language recognition through hand gestures. These specific gestures indicate specific English alphabets.

1. Gesture Recognition Using Artificial Neural Network

Khushboo Arora, Shrutika Suri, Divya Arora and Vaishali Pandey [3] published their research paper on Gesture Recognition Using ANN. Their research was mainly based on two types of gesture recognition namely Hand & arm gesture recognition: and Hand & face gesture recognition. The goal of face gesture recognition was to make machine effectively understand human emotion, regardless of the physical differences between individuals and body gesture recognition was to identify different body gesture to identify human activity.

2. Sign Language Recognition using Image-based Hand Gestures Recognition technique

Nikam and Ambekar [4] proposed image-based segmentation methods for robust recognition of static and dynamic hand gestures in real-time. These methods are used for an intuitive interaction with an assistance-system in which the skin-tones are used to segment the hands. Various sign language systems have been developed around the world, but they are neither flexible nor cost effective, so this paper introduced software which presents system which automatically recognizes sign language which help deaf and dumb people to interact with each other and normal people.

3. Nepali Sign Language Translation Using Convolutional Neural Network

Drish Mali, Rubash Mali, Sushila Sipai and Sanjeeb Prasad Pandey [5] published their paper on NSL TRANSLATION USING CNN which used 3 steps for sign recognition: image segmentation, developing a CNN model and prediction. Initially, a captured image is modeled using the HSV (hue, saturation, value) color model to obtain red mask. Then CNN uses multiple layers designed to require minimal preprocessing. The architecture of model consists to two convolutional layers, pooling layer which extracts features and Artificial Neural Network (ANN) which classify the features into classes. Finally, the class with the highest probability is predicted as the class of the input image by the model.

4. Sign Language Recognition Using Keras and Tensorflow

EvilPort2 [8] which implements CNN algorithm to recognize American Sign Language using hand gestures using Python, OpenCV, Keras and Tensorflow. There is 46 gesture sample (26 English alphabet+ 10 numbers) using OpenCV. He then converted images to grayscale which were then flipped along vertical axis making total number of images of each gesture to 2400.

5. David and Shah [7] proposed a model-based approach by using a finite state machine to model four qualitatively distinct phases of a generic gesture. Hand shapes were described by a list of vectors and then matched with the stored vector models.

There are two common deep neural network architectures: the Convolutional Neural Network (CNN) and the Recurrent Neural Network (RNN). CNN is a special form of the Feedforward Neural Network (FNN), also known as the multi-layer perceptron (MLP), trained with backpropagation. CNN are used to recognize visual patterns directly from pixel images with variability. There has been a large amount of recent efforts devoted to the understanding of CNNs. Examples include scattering networks, tensor analysis, generative modeling, relevance propagation and Taylor decomposition etc.

This paper presents the idea of recognizing the ASL hand gestures. Most of the authors have used red glove for easier segmentation and 2D CNN for recognition.

2.2 Image Acquisition

Data is acquired through a 1 Mega pixel camera. The image data set consists of ASL gestures referenced from American Sign Language Dictionary and American Sign Language Alphabet provided by National Deaf Federation - America (NDF-A). A total of 26 Basic signs were selected. The data set consists of 590 grey scale images (472 training and 118 test set) for each sign. These images were captured by 2 people under different lighting conditions. Each image in the training set were re-sized into 96*96-pixel image and was further augmented using the operation shear, re-scale, zoom, horizontal flip.

The Sign were captured using ASL given in figure below

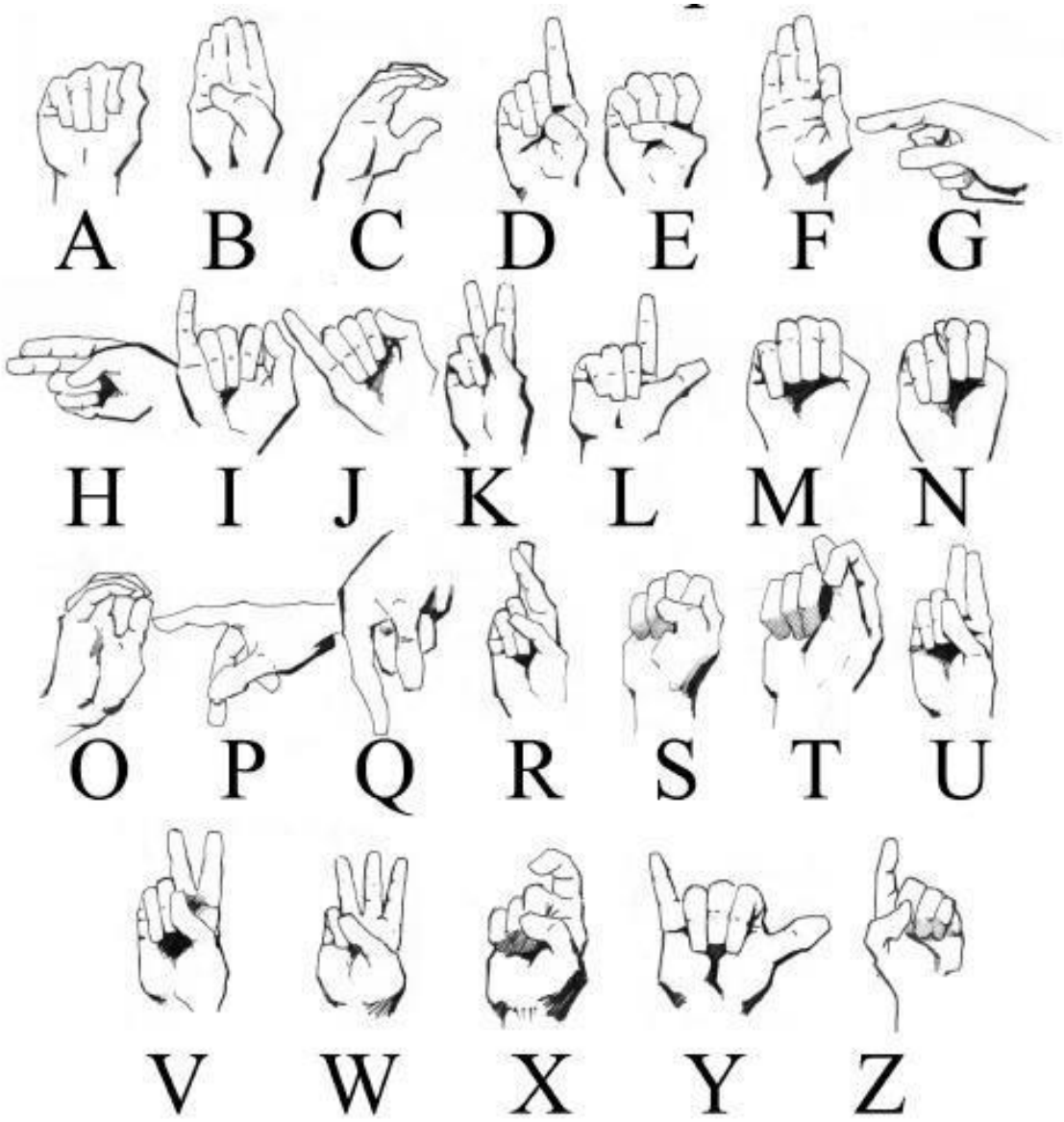

Figure 1 : ASL Dataset

3. METHODOLOGY

The methods used for the accomplishment of the recognition of signs of NSL can be divided into 3 steps: Image Segmentation, developing a CNN model and Prediction.

3.1.1 Image Segmentation

Initially, an image was captured using the camera of a laptop. Image was modeled using the HSV (hue, saturation, value) color model to obtain greyscale mask of the image as shown in figure below.

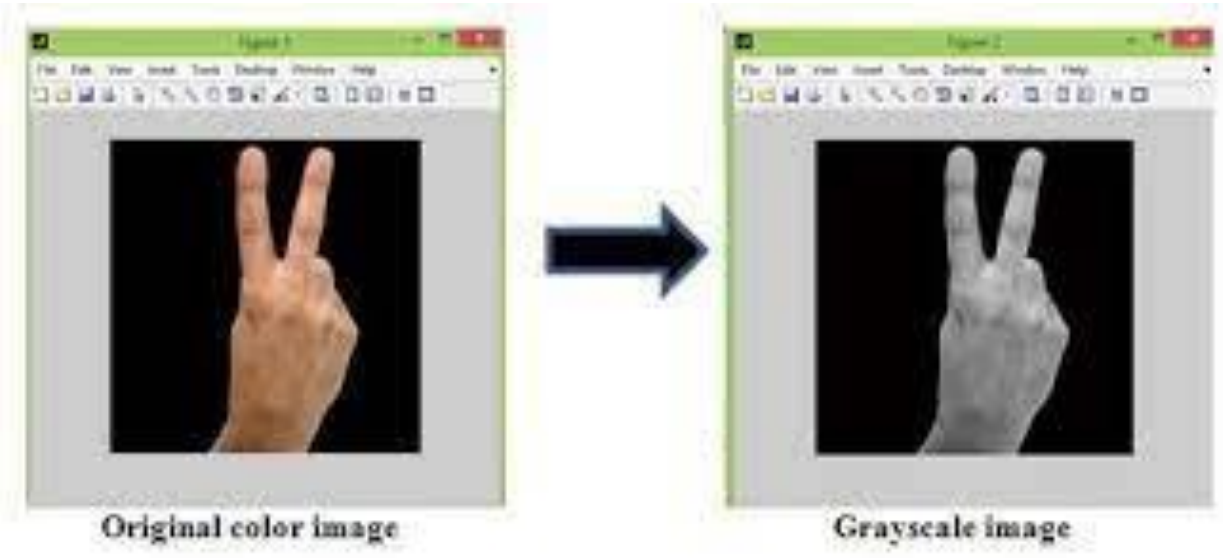

Figure 2: Image segmentation [8]

3.1.2 The Convolutional Neural Network Model

Convolutional Neural Network (CNN) is a class of deep, feed-forward artificial neural networks, most commonly applied to analyze visual imagery. CNNs use a variation of multilayer perceptron designed to require minimal preprocessing. CNN has proved to be a significant milestone in the area of detection and classification of images.

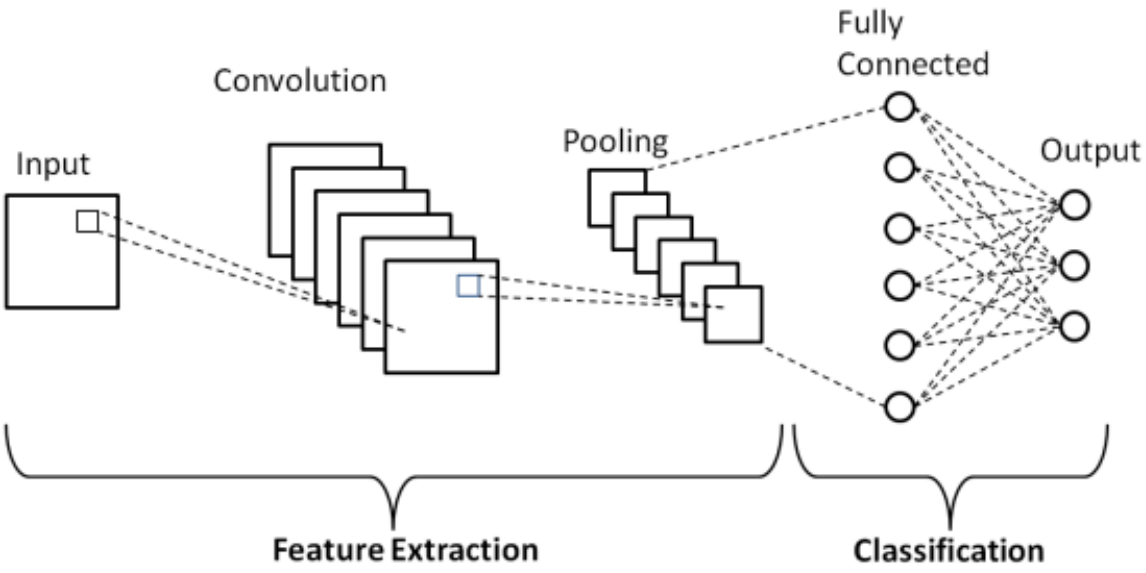

Figure 3: Systematic diagram of basic CNN [9]

The architecture of model consists of two convolutional layer, pooling layer and one Artificial Neural Network (ANN). Convolutional layer, pooling layer are used to extract features and ANN is used to classify the features into classes.

In convolution layers, 32 feature map is sampled of the input to generate feature map using convolution operation which is further sampled by pooling matrix using max pooling technique. The convolution and max pooling operation are further repeated once. The output is then attended into 1D vector and supplied to the ANN as an input. Then, the ANN uses the input and the weight which are generated from training and the 128 neurons in the fully connected (hidden) layer for the image classification into one of the classes.

3.1.2.1 Layers used to build ConvNets

A simple ConvNet is a sequence of layers, and every layer of a ConvNet transforms one volume of activations to another through a differentiable function. We use three main types of layers to build ConvNet architectures: Convolutional Layer, Pooling Layer, and Fully Connected Layer. We will stack these layers to form a full ConvNet architecture.

In our case a simple ConvNet have the architecture [INPUT - CONV - RELU - POOL - FC].

In more detail:

a. INPUT [96x96x3] will hold the raw pixel values of the image, in this case an image of width 96, height 96, and with three color channels R, G, B.
b. CONV layer will compute the output of neurons that are connected to local regions in the input, each computing a dot product between their weights and a small region they are connected to in the input volume. This may result in volume such as [96x96x5] we decided to use 5 filters.
c. RELU layer will apply an elementwise activation function, such as the max (0,x) thresholding at zero. This leaves the size of the volume unchanged ([96x96x12]).
d. POOL layer will perform a down sampling operation along the spatial dimensions (width, height), resulting in volume such as [16x16x5] and every time changing size so that training can be done with variable sampling.
e. FC (i.e., fully connected) layer will compute the class scores, resulting in volume of size [1x1x1024], where each of the 1024 numbers correspond to a class score, such as among the 1024 categories. As with ordinary Neural Networks and as the name implies, each neuron in this layer will be connected to all the numbers in the previous volume.

In this way, ConvNet transform the original image layer by layer from the original pixel values to the final class scores. Note that some layers contain parameters and other don't. In particular, the CONV/FC layers perform transformations that are a function of not only the activations in the input volume, but also of the parameters (the weights and biases of the neurons). On the other hand, the RELU/POOL layers will implement a fixed function. The parameters in the CONV/FC layers will be trained with gradient descent so that the class scores that the ConvNet computes are consistent with the labels in the training set for each image.

Simple flow of our CNN is shown in figure below

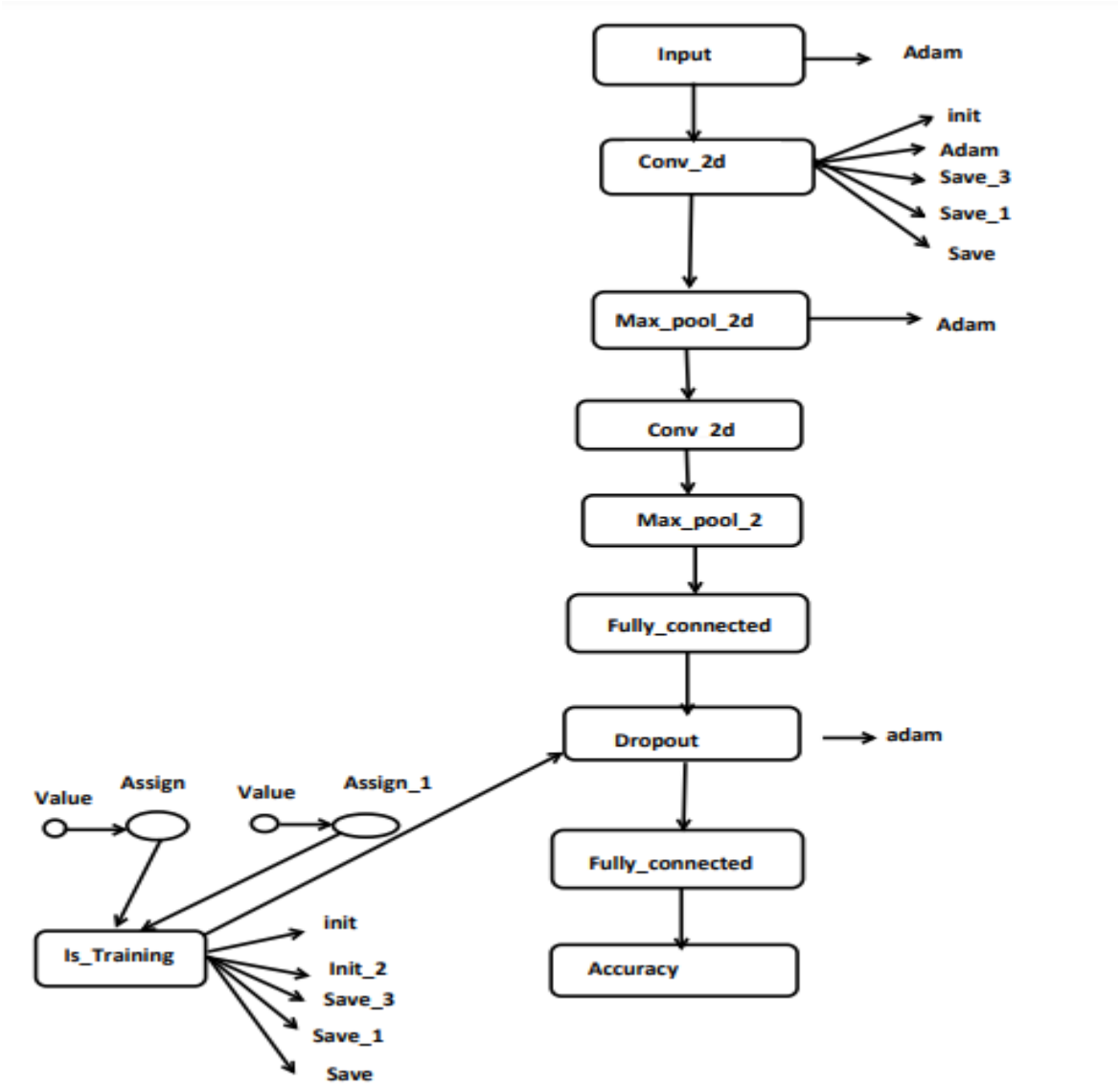

Figure 4: flow of our CNN Model

3.2.3 Prediction

The class with the highest probability is predicted as the class of the input image by the model.

3.3 Software Development Model

The software development model are the various processes or methodologies that are being selected for the development of the project depending on the project's aims and goals. The software development process must generate working software quickly with lower initial cost and easy risk management. The software must be easier to test and debug during small iterations. On analyzing and researching all these features of Software Development Process, Incremental Model was assured to be the best fit for this project.

3.3.1 Incremental Model

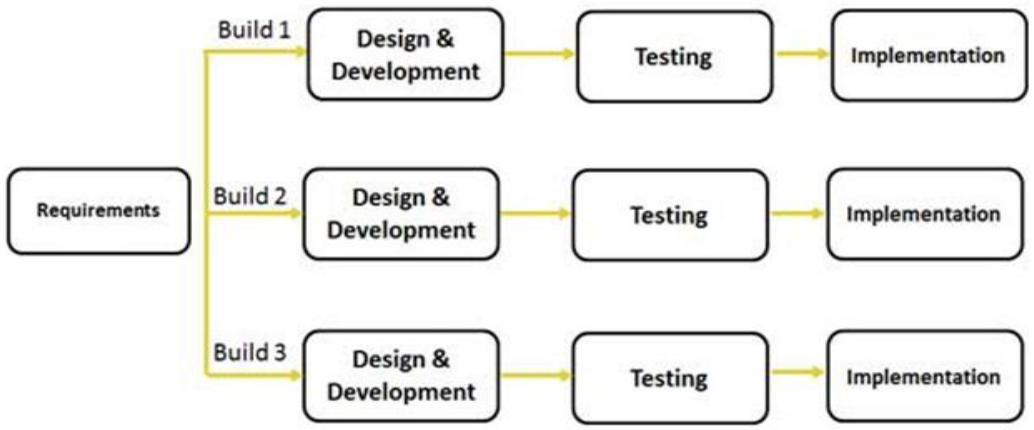

Figure 5: Incremental Model [10]

3.4 Work Schedule

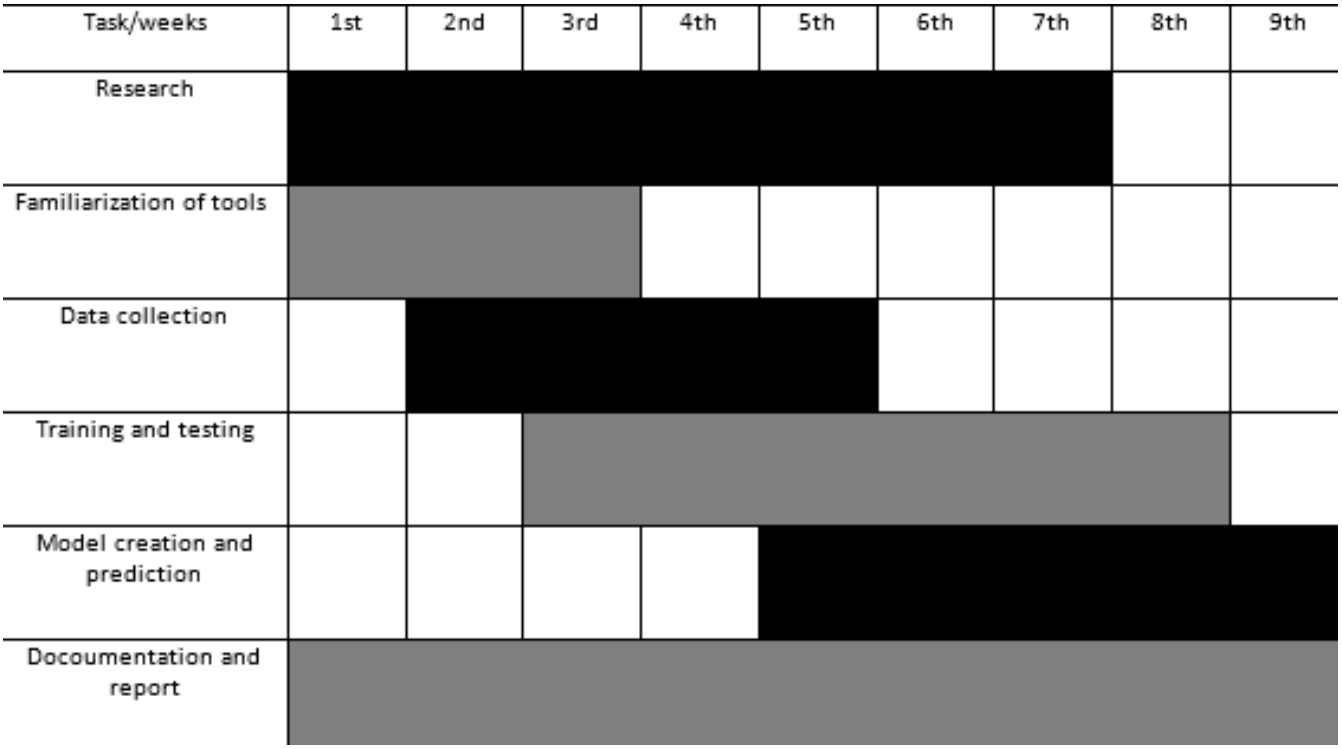

Figure 6: Gantt Chart

3.5 Working Mechanism

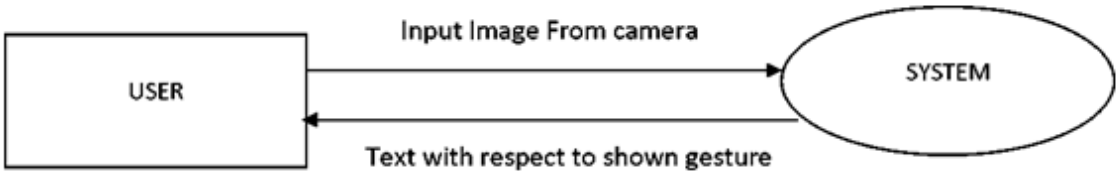

Fig: level 0 DFD of the system

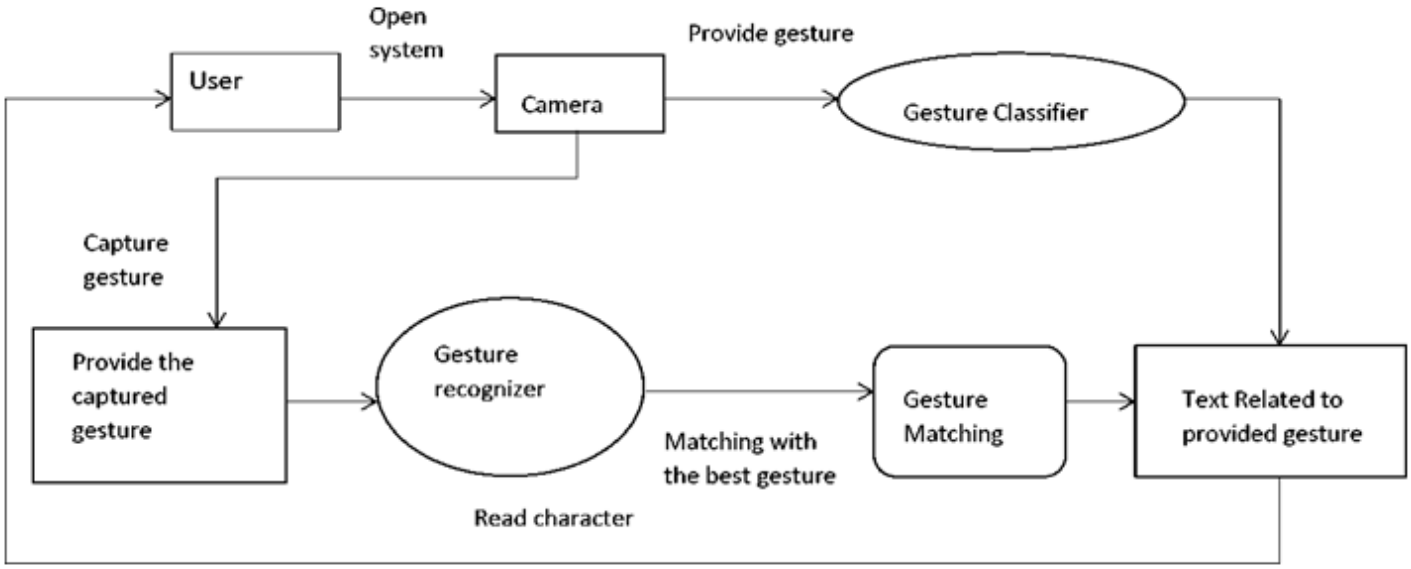

Fig: Level 1 DFD of the system

Figure 7: Data Flow Diagram

3.6 Requirements

Functional Requirements:

1. Authentication: The system should be able to verify each participant involved in the network.

2. Permission Based: The system should verify roles of each participant based on the permissions set for them.

3. Addition to the block: The system should add information in the CNN once it has successfully passed all the necessary tests.

Non-Functional Requirements:

1. Security: The system should provide a level of trust/guarantee to the users that participating in the network.

2. Transparency: The system should deliver a transparent view of the product/service from its origin to the consumers.

3.7 Tools Used:

Visual Studio Code: Visual Studio is a free source code editor refined and optimized for building and debugging modern web and cloud application.

Sublime Text: Sublime Text is a shareware cross-platform source code editor with a Python application programming interface. It natively supports many programming languages and markup languages, and functions can be added by users with plugins, typically community-built and maintained under free-software licenses.

3.8 Technology Used:

Python: Python is interpreted, high-level, and general-purpose programming language. It is the most popular language for machine learning and artificial intelligence. It is a general-purpose programming language, so it can be used for many things. Python is used for web development, AI, machine learning, operating systems, mobile application development, and video games.

NumPy: NumPy is a library for python programming language. It supports large multi-dimensional arrays and matrices. It brings the computational power of languages like C and FORTRAN to Python, a language much easier to learn and use.

OpenCV: OpenCV is a library of programming functions mainly aimed at real-time computer vision. It is the huge open-source library for the computer vision, machine learning, and image processing and now it plays a major role in real-time operation which is very important in today's systems.

TensorFlow: TensorFlow is an end-to-end open-source platform for machine learning. It has a comprehensive, flexible ecosystem of tools, libraries and community resources that lets researchers push the state-of-the-art in ML and developers easily build and deploy ML powered applications.

4. SYSTEM DESIGN

In order to extract features and recognize gesture following system design is used

1. The webcam which allows the user to capture the scene. This phase is called image acquisition.

2. After capturing the image, next step is to detect the hand gesture and separate the hand gesture from the scene. Because our interest is only the hand gesture which is needed for accurate classification. This is done by defining the region of interest (ROI).

3. Preprocessing is done through multiple steps

   a) Convert RGB image to gray scale image
   b) Gray filtering using value
   c) Noise removal and smoothing
   d) Remove small objects other than hand

4. Feature extraction

5. Pattern matching and recognition by CNN using TensorFlow

4.1 System Block diagram

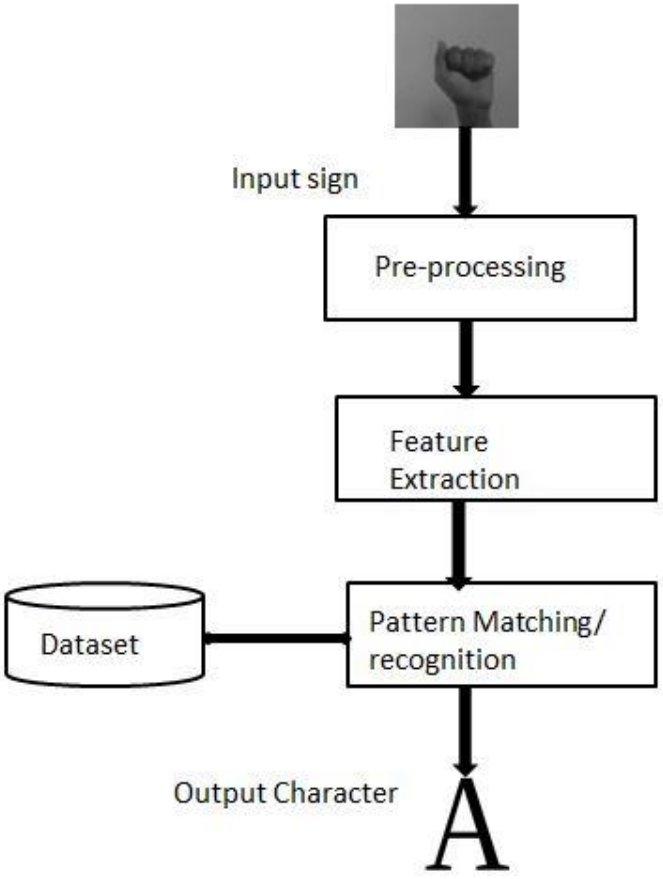

Figure 8: System Block diagram

4.2 Flowchart

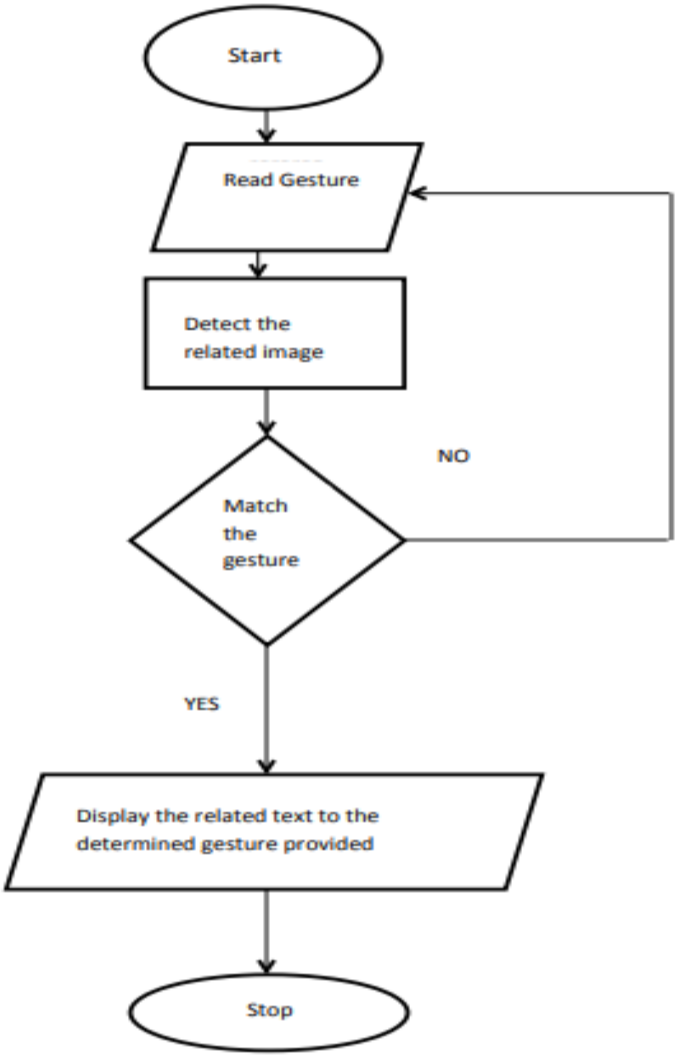

Figure 9 : Flow Chart

4.3 Use Case Diagram

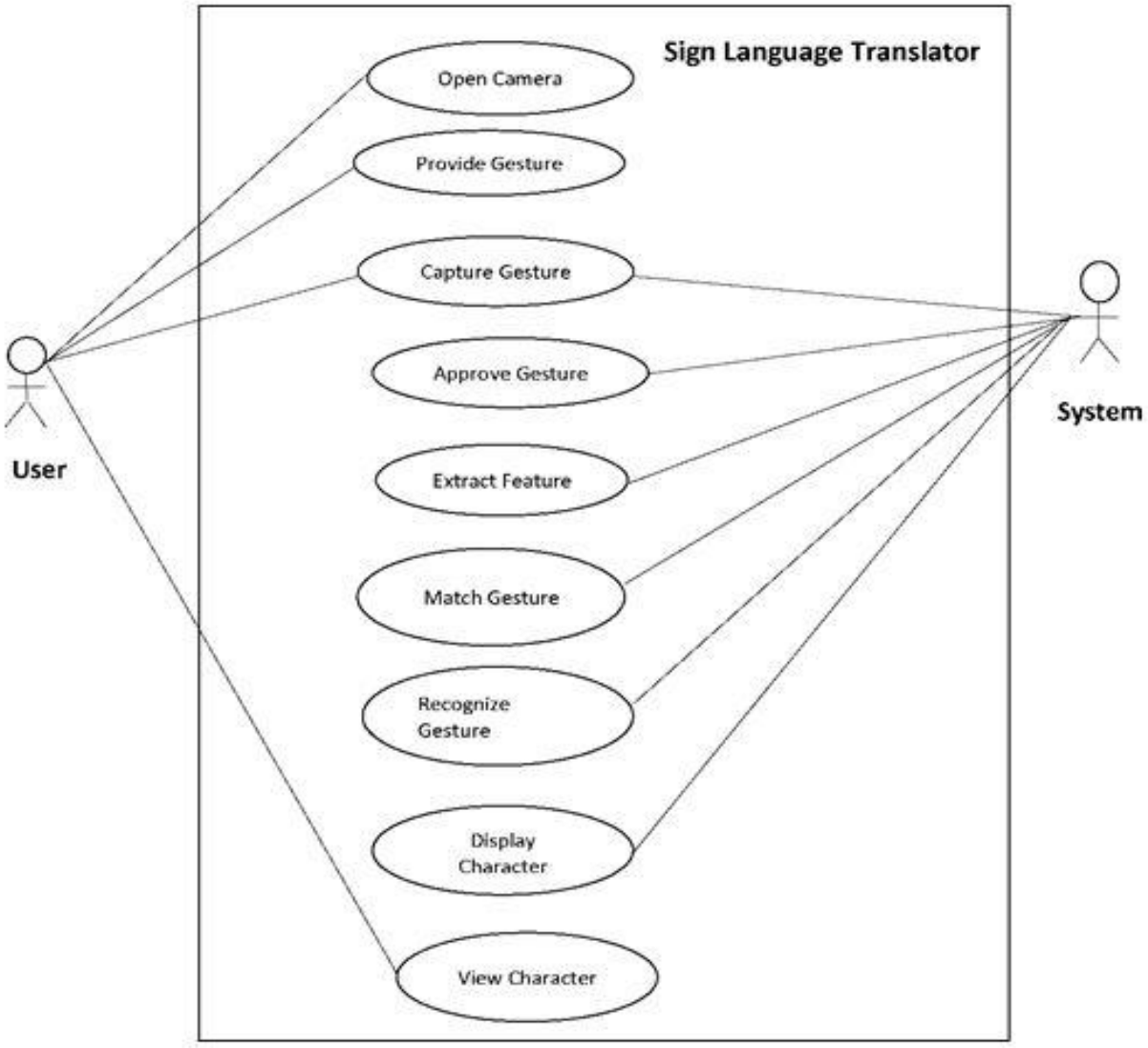

Figure 10: Use Case Diagram

4.4 Sequence Diagram

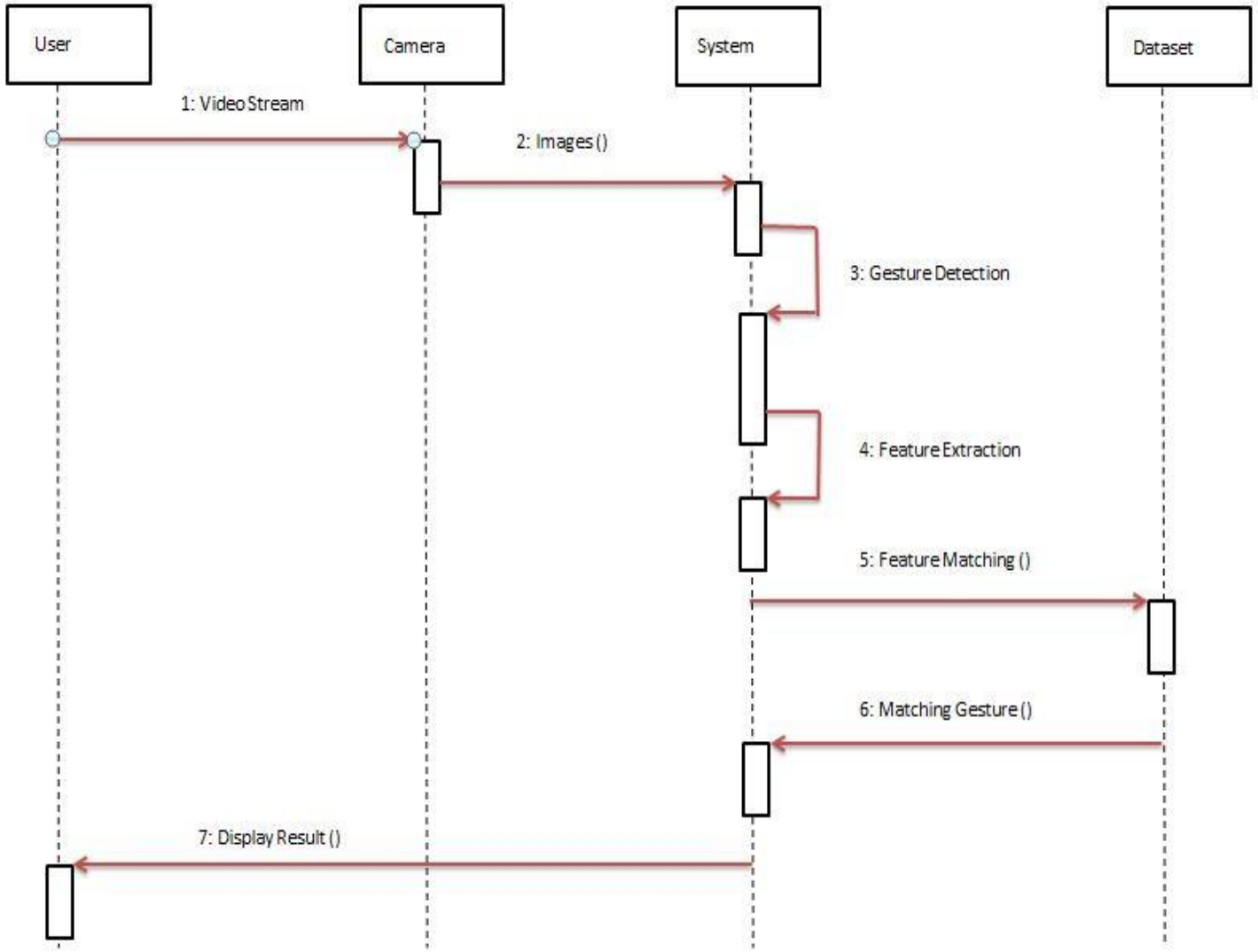

Figure 11: Sequence Diagram

5. RESULT AND DISCUSSION

In this project, a Dell laptop with the specifications of 4GB RAM, 2.3 GHz i3 processor and another Dell laptop of 8 GB RAM and 2.7 GHz i7 processor were used to train and test the model.

The number of epochs used for training the model was 7 and 20 respectively. The CNN model was compiled using Adam Algorithm for gradient descent process and categorical cross entropy as loss function.

It took 14 minutes for the Dell laptop to train the model and 30 minutes for another Dell laptop to train the model. The accuracy of this model was 81.64% (on the data set of 590 images for training and 118 of each class for testing).

Again, the CNN model was trained for predicting for 26 signs classes each of 590 images with total 15419 images of class after augmentation.

Training and testing were divided in the ratio of 80:20(15419 for training and 3083 for validating).

5.1 Limitations

- In real time gesture recognition, the background and position of hand should be in the same position as given the dataset, otherwise the output won't be shown properly.
- The system can only recognize the letters and does not show words combing the letters.

5.2 Problems Faced

- A lot of training was needed, as accuracy was showing way low than feasible limit (70%).
- Due to lack of knowledge of TensorFlow and OpenCV, it was difficult at first in getting familiar with its uses.

6. SCREENSHOOTS

6.1 Image Capturing

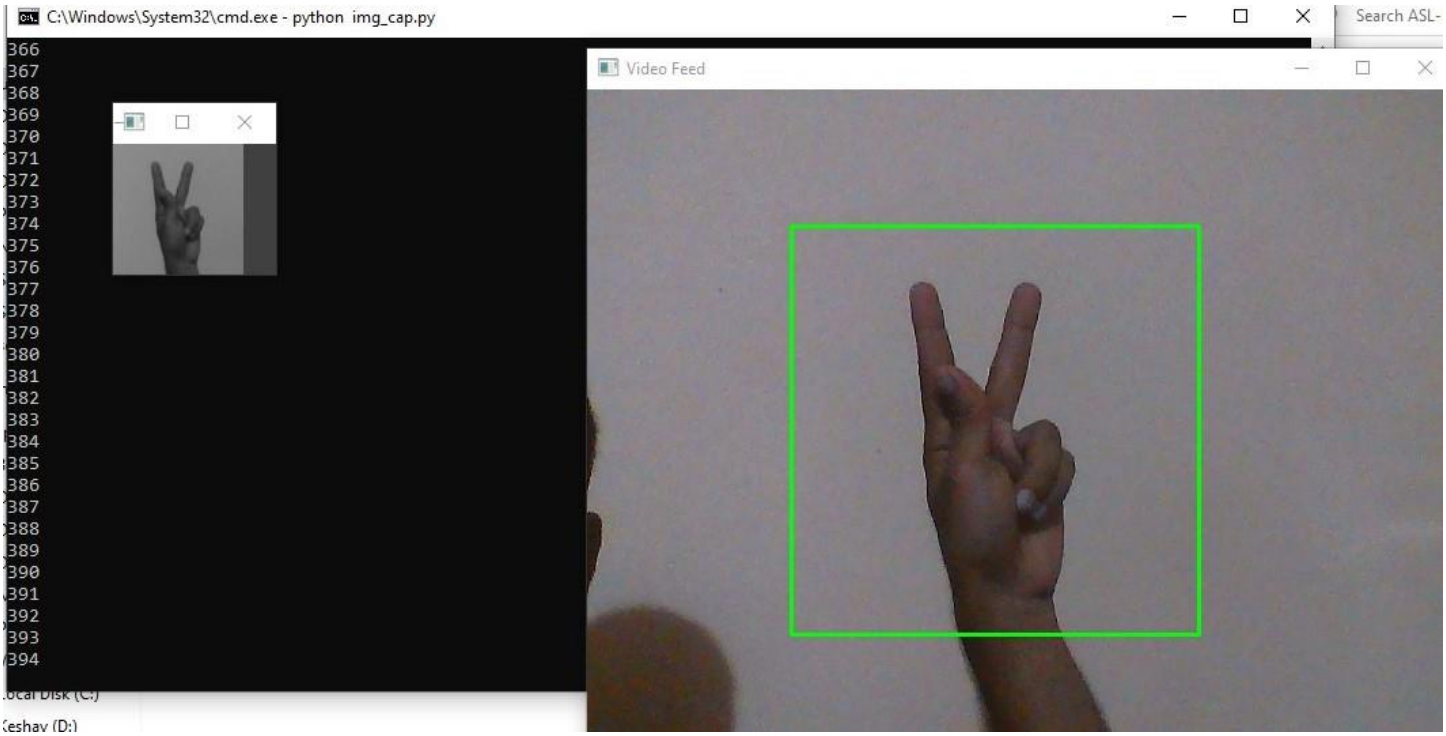

Figure 12: Image capturing Sample

6.2 Collected Data

Figure 13: Collected Data sample

6.3 Trained Dataset

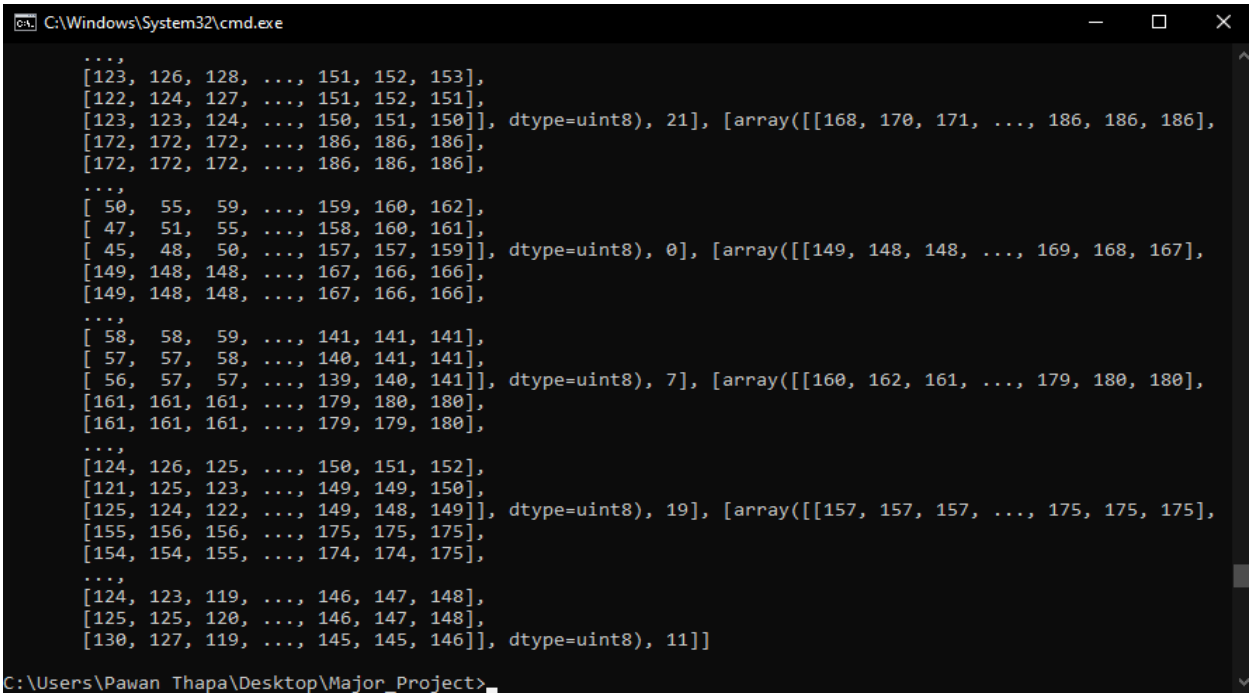

Figure 14 : Training Dataset

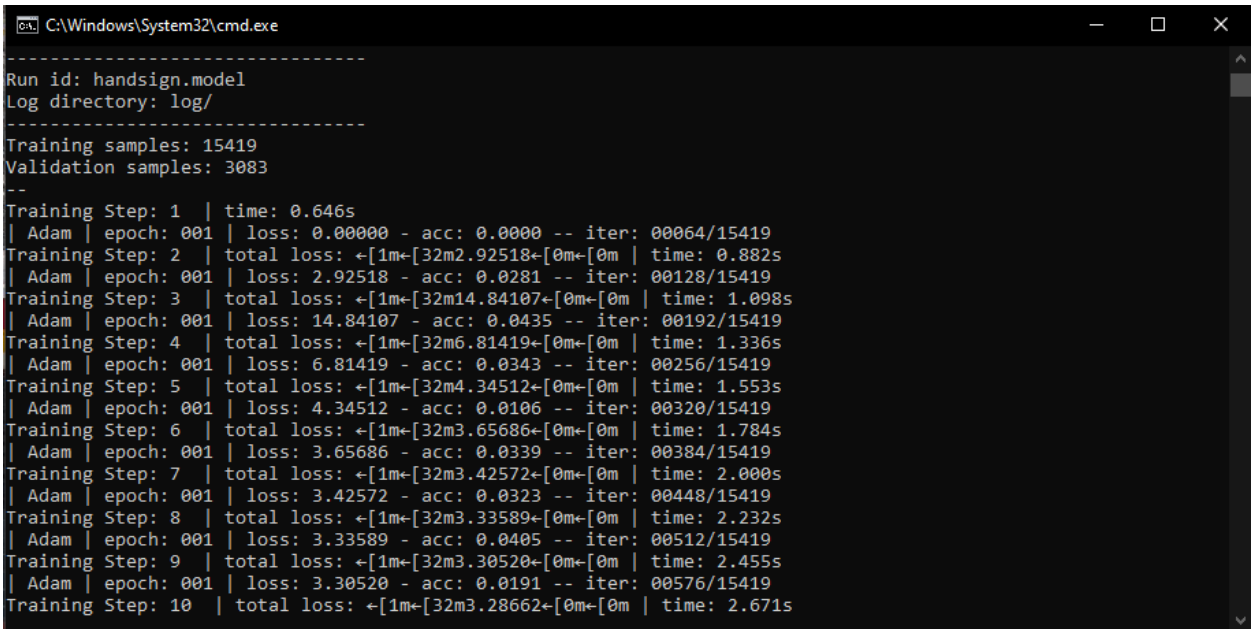

Figure 15: Training and Testing sample

6.4 Accuracy of the Model

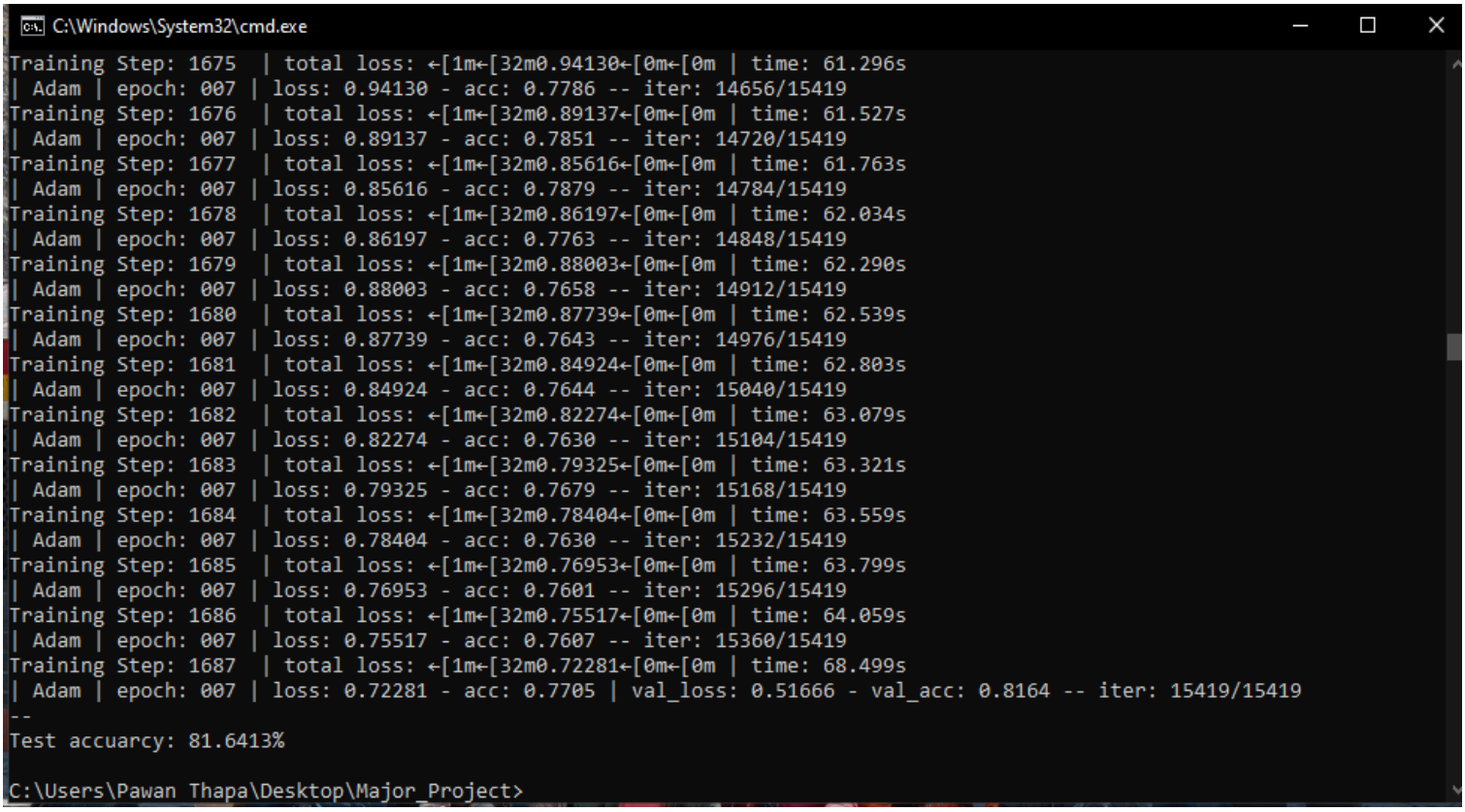

Figure 16: Accuracy score

6.5 Sample Outputs

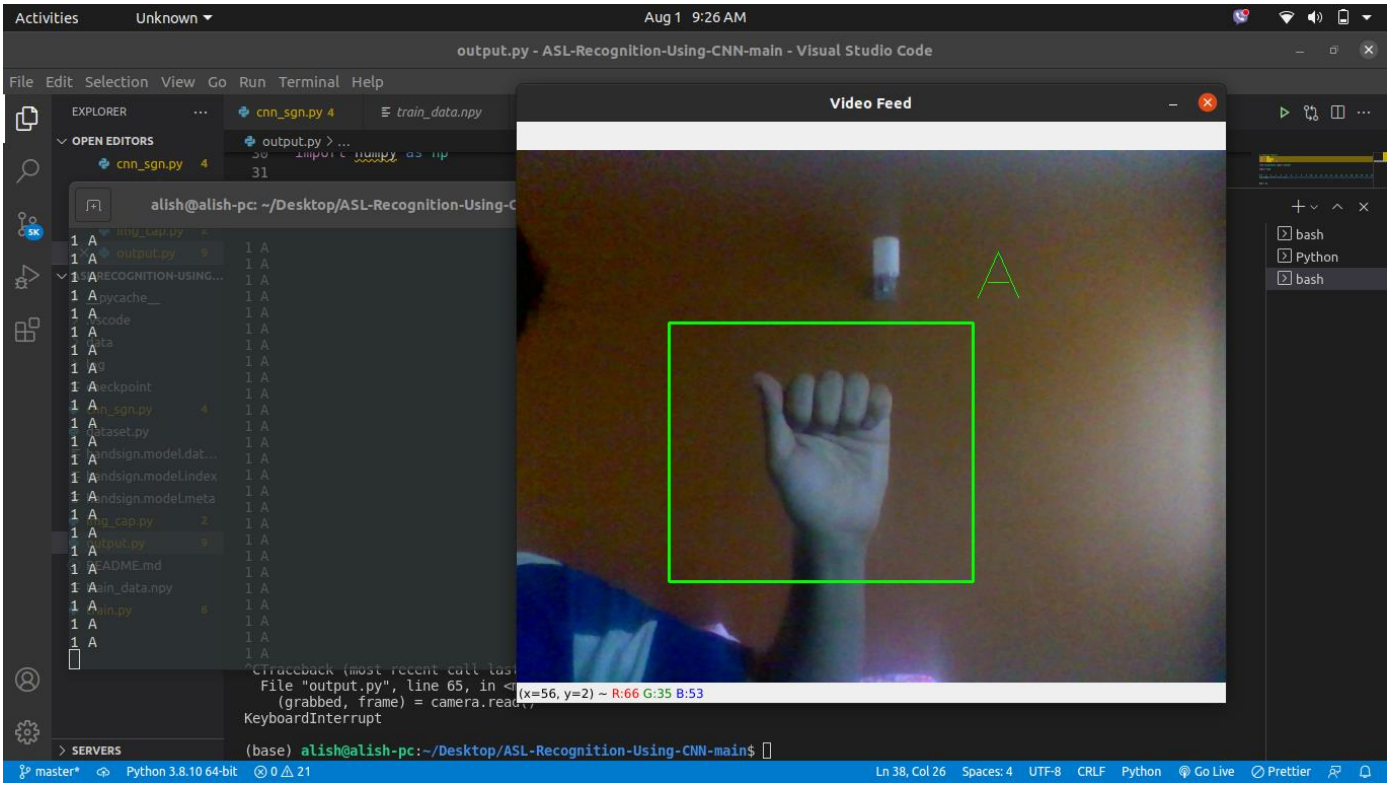

Figure 17: Sample Output-I

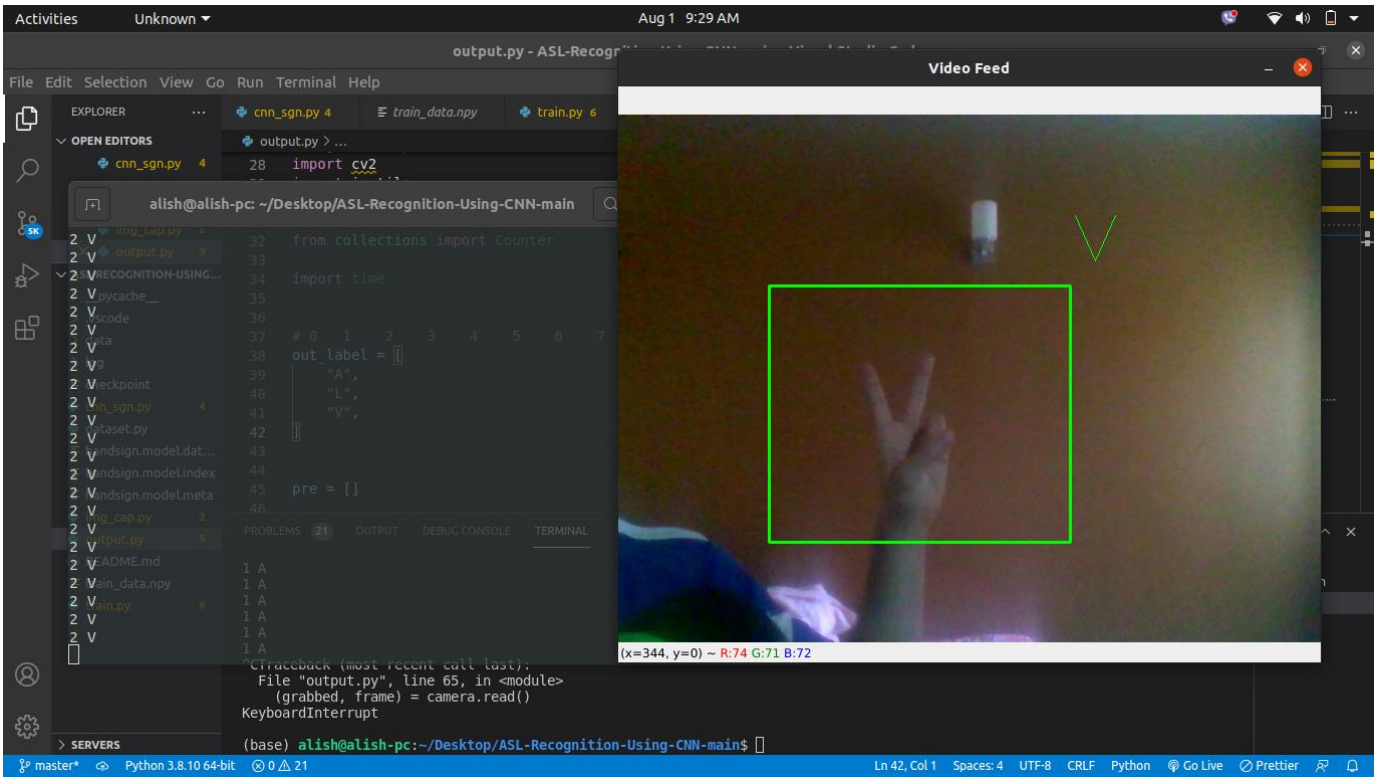

Figure 18: Sample Output-II

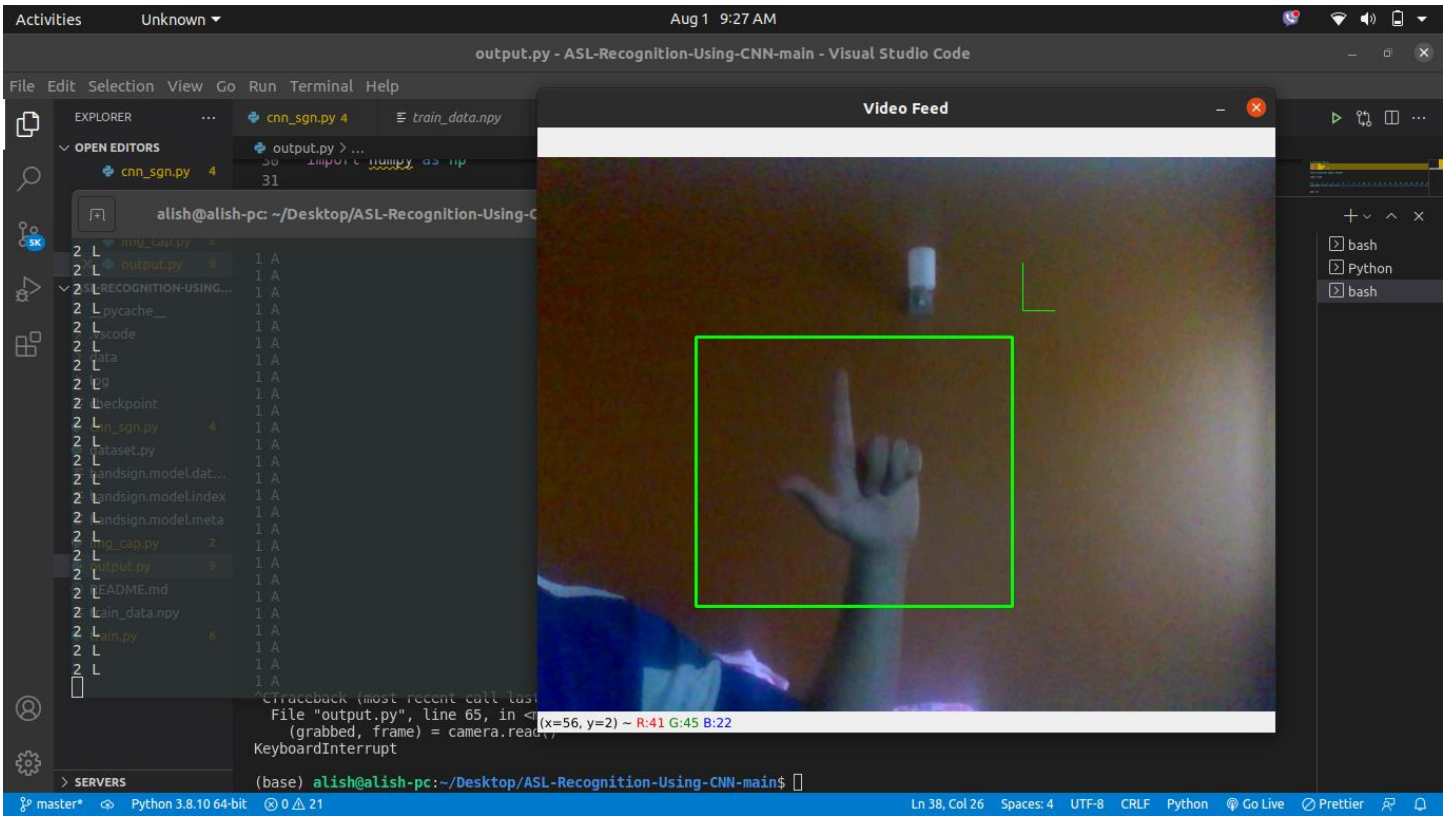

Figure 19: Sample Output-III

7. CONCLUSION

Thus, the system will be able to recognize human hand gesture and display ASL in words on the screen. This will enable people to understand ASL easily even when they don't know the sign language. As we are developing web-based sign language recognition application, the application will be compatible with all the modern browsers and as we aspect it will be high performance and very less delay during processing. User friendly UI to keep it simple and less complex. The purposed system will be able to recognize static and dynamic hand gestures.

8. REFRENCES

[1] B. Sharma, M. P. Dahal, BasantaKhadka, 'Hearing Impairment in Nepal', Hearing Impairment, pp. 70-77, 2004.

[2]World health organization (2018) world report on disability. https://www.spotlightmetal.com/machine-learning--definition-and-application-examples-a-746226/?cmp=go-ta-art-trf-SLM_Allg_DSA-20201118&gclid=EAIaIQobChMI-Mn1q6PI7QIVDiUrCh2dggbDEAAYAiAAEgLZdvD_BwE

[3]Khushboo Arora, Shrutika Suri, Divya Arora and Vaishali Pandey; "Gesture Recognition Using Artificial Neural Network", International Journal of Computer Sciencesand Engineering (IJCSE), Volume-2, Issue-4, April 2014.

Microsoft Word - 38-IJCSE-00223 (ijcseonline.org)

[4] Ashish S. Nikam and Aarti G. Ambekar, 'Sign language recognition using image based hand gesture recognition techniques', IEEE online int. conf. GET, pp. 14-17, 2016.

Sign language recognition using image based hand gesture recognition techniques | IEEE Conference Publication | IEEE Xplore

[5] Drish Mali, Rubash Mali, SushilaSipai and Sanjeeb Prasad Pandey "NSL TRANSLATION USING CNN", KEC Conference, Voulme-1, September 2018.

10. Pdf (kec.edu.np)

[6] Chung-Lin Huang and Wen-Yi Huang, 'Sign language recognition using model-based tracking and a 3D Hopfield neural network', Machine Vision and Applications, vol. 10,pages 292-307,1998.

MVA155 (nthu.edu.tw)

[7] J. Davis and M. Shah, 'Visual gesture recognition', IEE Proceedings on Vision, Image and Signal Processing, 141(2): pages 101106, 1994.

[8] Segmentation method for ASL, pnrsolution.org, Datacenter, Vol-3, Issue-5.

[9] Schematic diagram of a basic convolutional neural network (CNN) architecture, https://www.researchgate.net/figure/Schematic-diagram-of-a-basic-convolutional-neural-network-CNN-architecture-26_fig1_336805909.

[10] SLC Iterative Model, https://www.tutorialspoint.com/sdlc/sdlc_iterative_model.htm