"A NOVEL APPROACH TO CONVERT SIGN LANGUAGE TO TEXT IN A CLIENT- SERVER ENVIRONMENT"

MAJOR PROJECT REPORT

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE AWARD OF THE DEGREE OF

BACHELOR OF TECHNOLOGY

(Computer Science and Engineering)

Submitted By:
Guneet Kohli(1805172)
Harneet Kaur(1805182)
Jashanpreet Kaur(1805188)

Submitted To:
Dr. Sumeet Kaur Sehra
Assistant Professor

Department of Computer Science and Engineering
Guru Nanak Dev Engineering College
Ludhiana,141006

ABSTRACT

The only medium of communication between the people with disabilities in hearing and speaking is sign language. Even if they need to present their thoughts or ideas to any individual, they will do that using actions. It may not be understood accurately and efficiently by the individual, which may result in misunderstanding leading to greater problems.

In this project, we have aimed at converting their actions, the sign language, to text. The actions will be understood by the system, depending upon the knowledge of the actions, and then translated accordingly to the required text. The text can be easily read and thus favoring communication.

The software also aims at saving time by encoding some signs to a text that would normally be large in size. We just need to use the desired sign and it will be converted automatically to the large text. This can also be used for security purposes, for developing one's own Sign language.

ACKNOWLEDGEMENT

I/We are highly grateful to Dr. Sehijpal Singh, Principal ,Guru Nanak Dev Engineering College (GNDEC) Ludhiana ,for providing this opportunity to carry out the minor project work on **A Novel Approach to convert Sign Language to Text in a Client- Server Environment .**

The constant guidance and encouragement received from Dr. Parminder Singh H.O.D. CSE Department ,GNDEC Ludhiana has been of great help in carrying out the project work and acknowledgement with reverential thanks.

I/We would like to express a deep sense of gratitude and thanks profusely to Dr. Sumeet Kaur Sehra , without her wise counsel and able guidance ,it would have been impossible to complete the project in this manner .

I/We express gratitude to other faculty members of the computer science and engineering department of GNDEC for their intellectual support throughout the course of this work.

Finally, I/We are indebted to all whosoever have contributed in this report work

Guneet Kohli

Harneet Kaur

Jashanpreet kaur

List of Figures

Fig no.	Figure Description	Page No.
1.0	American Sign Language	11
1.1	Many-to-one Problem	12
1.2	Proposed system	14
2.0	Basic steps involved in GNS3 setup	18
2.1	Configuration & Logical divisions of routers	19
3.0	Zero Level DFD	20
3.1	First Level DFD	20
3.2	Databases & XML Files	21
3.3	Approach used	22
3.4	Hand Detection	22
3.5	Histogram of the hand	23
3.6	Convolution of a matrix	23
3.7	Activation function: Rectifier function	24
3.8	Max Pooling	25
3.9	Flattening	25

Fig No.	Figure Description	Page No.
3.10	Model consisting of 5 layers	26
4.0	Steps involved in simulator setup	29
4.1	Pepcoding rules	30
4.2	Project Scheduling	31
4.3	Splitting of data	31
5.0	Final representation	32
5.1	User Interface	33
5.2	User Interface	33
5.3	Dragging and dropping routers and related components (as per topology)	36
5.4	Assigning IP’s to the routers accordingly as to create different areas	36
5.5	Assigning loopback to each router	37
5.6	Interconnecting the areas using OSPF	37
5.7	Pinging an IP from different area (to check connectivity)	38
5.8	Configuring newly created adapter	39
5.9	Attaching the host(created in topology) to our local PC	40
5.10	Creating a virtual Machine named Server 2008	40

Fig No.	Figure Description	Page No.
5.11	Browsing the windows server for booting the OS.	41
5.12	Internet Information Services (IIS)	42
5.13	Accepting requests from remote client computers	42
5.14	sharing and delivering information across networks	43
5.15	sharing and delivering information across networks	43
5.16	Browsing the website on Virtual server.	44
5.17	Pinging the website using IP address (as assigned) on local PC.	45
5.18	Final Representation of User Interface	46
5.19	Screenshot of the result obtained for letter A	47
5.20	Screenshot of the result obtained for letter E	47
5.21	Screenshot of the result obtained for letter F	48
5.22	Sign Detection	48
5.23	Sign Detection	49

LIST OF TABLES

Table no.	Table Description	Page No.
2.1	Software Requirements	16
2.2	Hardware Requirements	16

INDEX

Chapter 1. Introduction
1.1 Introduction to Project
1.2 Project Category(Internet based, Application or System Development, Research based ,Industry Automation, Network or System Administration)
1.3 Objectives
1.4 Problem Formulation
1.5 Identification/Reorganization of Need
1.6 Existing System
1.7 Proposed System
1.8 Unique Features of the System
Chapter 2. Requirement Analysis and System Specification
2.1 Feasibility study (Technical, Economical, Operational)
2.2 Software Requirement Specification Document which must include the following: (Data Requirement, Functional Requirement, Performance Requirement ,Dependability Requirement, Maintainability requirement, Security requirement, Look and feel requirement)
2.3 Validation
2.4 Expected hurdles
2.5 SDLC model to be used
Chapter 3. System Design
3.1 Design Approach (Function oriented or Object oriented)
3.2 Detail Design
3.3 System Design using various Structured analysis and design tools such as : DFD's,Data Dictionary,Structured charts,Flowcharts or UML
3.4 User Interface Design
3.5 Database Design
  3.5.1 ER Diagrams
  3.5.2 Normalization
  3.5.3 Database Manipulation
  3.5.4 Database Connection Controls and Strings
3.6 Methodology
Chapter 4. Implementation, Testing, and Maintenance
4.1 Introduction to Languages, IDE's, Tools and Technologies used for Implementation
4.2 Coding standards of Language used
4.3 Project Scheduling using various tools such as PERT, GANTT charts, Open PROJ etc.
4.4 Testing Techniques and Test Plans
Chapter 5. Results and Discussions
5.1 User Interface Representation (of Respective Project)

5.1.1 Brief Description of Various Modules of the system
5.2 Snapshots of system with brief detail of each
5.3 Back Ends Representation (Database to be used)
5.3.1 Snapshots of Database Tables with brief description

Chapter 6. Conclusion and Future Scope
References/Bibliography

CHAPTER 1:INTRODUCTION

1.1 Introduction to project

This project is an effort to develop a robust, reliable, and scalable system that can be used by people who are deaf or hard of hearing to communicate with their friends, family, and colleagues. The project was started by two deaf students at the University of Washington in Seattle.

SLT converts sign language gestures into text messages. It is both intuitive and practical for use by individuals who are deaf or hard of hearing because it does not require any special training to use it.

Usually, Sign Language is considered to be the most natural and expressive way for hearing the impaired people. Deaf people use sign languages for communication.To convey meaning in sign languages, humans use hand gestures, body, facial expressions and movements.These sign languages are easily understood by Humans, but automatic sign language recognition for machines is a challenging task. Thus, to overcome this shortcoming, automatic sign language converters are needed today.

Deaf people find it very difficult to communicate with the hearing person and also there are not many options available to help them. All of these alternatives have some major flaws.Usually Interpreters are not available and are often expensive.Using Pen and paper is also not a feasible idea as it is uncomfortable, messy and even time consuming, both for the deaf and the hearing person. Everything around is getting automated, with the evolution of IoT.

Sign language is a complex method of communication that requires the use of hands, arms and body to form different shapes. ASL-text or American Sign Language to text is an application that uses motion detection and recognition software in order to translate sign language into text format. It helps deaf people communicate with those who are hearing without having to learn sign language by using their PC.

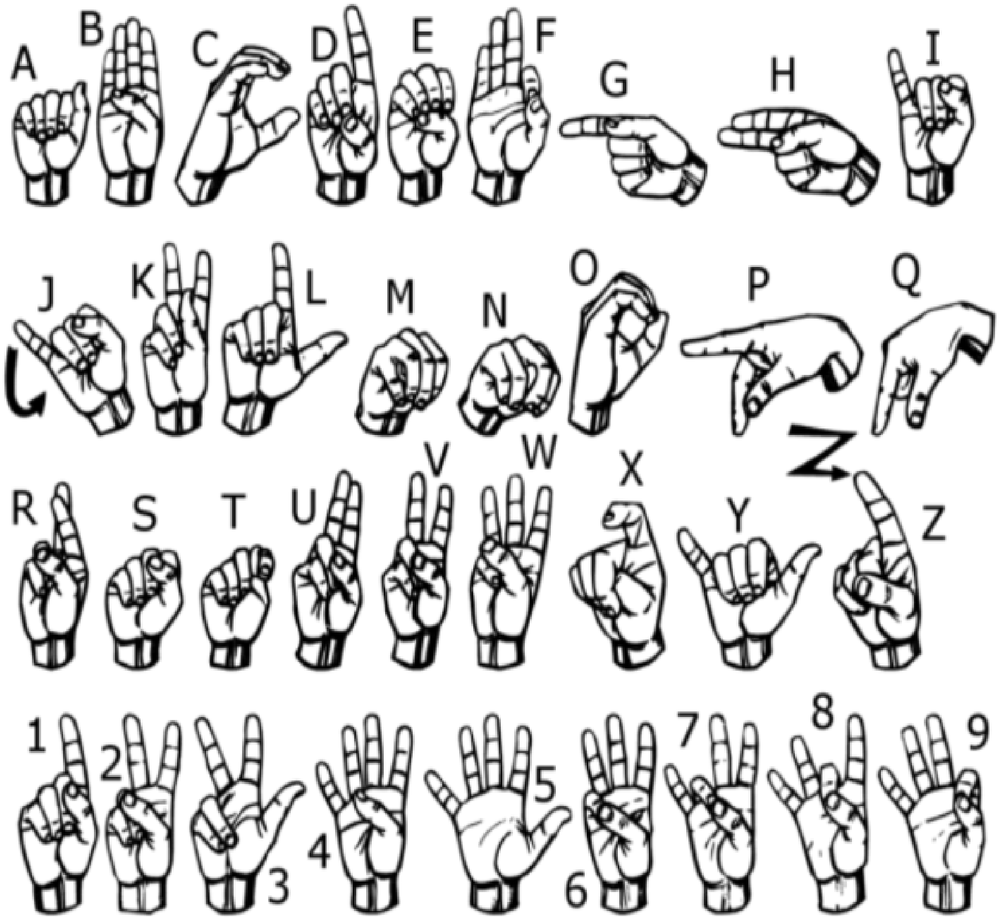


Figure 1.0: American Sign Language

1.2 Project Category

This would be a simple CNN project that will help its users to convert the sign language directly into text using different types of gestures which can be converted into text on the basis of the standard American Sign Language. The project will be trained on Keras. Word identification and working mathematical operations will be used. The major technologies used are IMAGE PROCESSING and CNN (Convolutional Neural Network).

1.3 OBJECTIVES

- To create gesture samples using OpenCV.
- To classify the gestures and label them with one of the various categories already defined while training the dataset

- To be able to identify sign language characters individually and as words when concatenated

1.4 PROBLEM FORMULATION

We are given a query video Q. We will be finding its corresponding sign S.
Our final goal is to learn a function A: Q →S which maps any query video Q belonging to a domain X to its corresponding sign S. Query video Q is a sequence of frames = ($q1$, $q2$, … , qt). This is a many-to-one problem because we are mapping multiple inputs in a sequence to a single output.

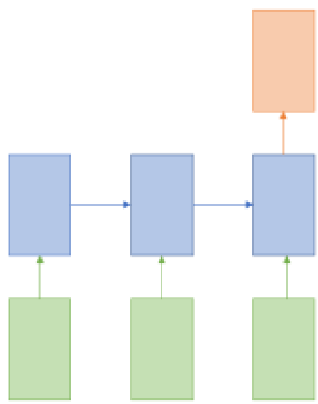


Figure 1.1: Many-to-One Problem

So, we can rewrite learning function A: Q → S to A: (q_1, q_2, … , q_t) →S where we simply substitute Q to (q_1, q_2, … , q_t).
We want a first model, that we call Model-1, to learn a function E: Q → F that maps input sequence (q_1, q_2, … , q_t) to feature sequence (f_1, f_2, … , f_t) by minimizing the Mean Squared Error loss function: $M.\,S.\,E\;Loss = (x - y)^2$
where x is the image pixels that is fed as input to the Model-1 and y is the pixels predicted by the Model-1.

1.5 IDENTIFICATION OF NEED

With the advancements in technology, the demand for Machine Learning and its applications is increasing at a fast pace. In the context of Machine Learning, in order to make any algorithm or model useful, its accuracy and efficiency must be very high.Hence,the knowledge of Machine Learning thus becomes an important factor. Since everything is getting automated, using an Interpreter to translate Sign language to text, in the era of Machine Learning, is just a waste of resources. Hence,classification of objects, object detection and image processing, all these plays a very vital role. By providing an automatic translation system, the main aim is to bridge the gap between normal and deaf, mute individuals

Almost 2 million people in this world are classified as Deaf and Dumb. As the only means of communication is sign language,they have a great difficulty in communicating with each other and also with the other individuals. Another drawback is that they need to learn this sign language. Suppose there is a person who is unaware of this sign language, so it would be extremely difficult for him to understand and decode their actions as it would be nearly impossible to identify anything without the prior knowledge of the language

Even if we consider the case of computers, they also need to have information in their memory to identify and provide data related to any object. IMAGE PROCESSING and CNN (Convolutional Neural Networks) are the major technologies used. This software is designed to bridge the communication gap between normal and deaf-mute individuals by providing an automatic translation system.

1.6 EXISTING SYSTEM

In the field of Computer Vision, Sign Language Recognition is not a new problem. Even in the 90's, the Hidden Markov Approach was used by the researchers for Sign Language Recognition because of their capability to capture temporal information.

Extra hardware sensors are used as feature extractors by the Researchers and then classifiers are used on these extracted features. But this work dealt with static data and was not real-time. Many researchers started using deep-learning techniques for sign language recognition due to the capability of these deep learning techniques to provide high performance and better performance but still, they were dealing with static data. Recently, with complex convolutional neural networks, the world has achieved real-time sign recognition performance and they have worked on dynamic data. Even if these Deep Neural Network methods need a lot of training data but they achieve high accuracy

We will be addressing the following few shortcomings of the previous literature with our work:

- Introducing a new deep-learning architecture which will achieve comparable results with limited training data
- Automate sign-language recognition for isolated signs

1.7 PROPOSED SYSTEM

By using a webcam,the signs are captured. Further processing of these signs is done for feature extraction using some colour model. Then by using a pattern matching algorithm, extracted features are compared. For the purpose of calculating the sign recognition, the features are compared with a testing database. Finally, the recognized gesture is converted into text. Hence, for a deaf-dumb people, this system will be serving as a great opportunity to communicate with non-signing people without the need of an interpreter.

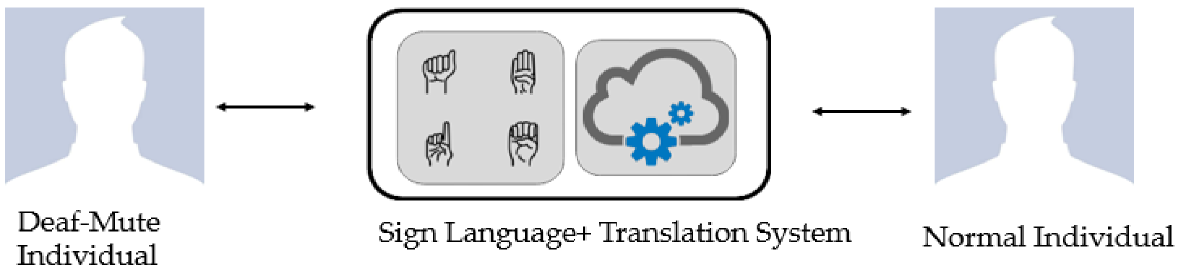


Figure 1.2: Proposed System

To make the dataset, various images of some gestures are taken and then processed. On these captured and modified images, the CNN model is then trained using Keras.The signs which are to be translated are then fed to the software which matches it with the existing images and classifies it.

1.8 UNIQUE FEATURES OF THE SYSTEM

- To create gesture samples using OpenCV.
- To classify the gestures and label them with one of the various categories already defined while training the dataset
- To be able to identify sign language characters individually and as words when concatenated

CHAPTER 2:SYSTEM REQUIREMENTS

2.1 FEASIBILITY STUDY

The aim of this document is to provide a detailed description of the translator of Sign language to text. The applications and features of the system will be covered, also the interfaces of the system, what the system is expected to do, the constraints that the project will work under and how it will behave in response to external stimuli. This is intended for both the developers and the users of this system.

This system is primarily intended for making an Interpreter. Whoever wants to employ deaf and mute employees can use it to convey employee messages to the end consumer. So, this can become a great application in Business. The deaf and mute people will use it to communicate. Further applications can be extended to security purposes, by developing a sign language of your own. And can even be used for observing and analysing any suspicious actions.

Glossary:

• **Feature:** Features are individual measurable properties or characteristics of a phenomenon being observed. These require classification.

• **Label:** Labels are the final output. We can also consider the output classes to be the labels.

• **Model:** A machine learning model is a mathematical portrayal of a real-life problem. There are various algorithms that perform different tasks with different levels of accuracy.

• **Regression:** Regression is a statistical method that is used to predict real and continuous valued functions.

• **Classification:** In classification, we will need to categorize data into a finite number of predefined classes.

• **CNN :** It is a Machine Learning unit algorithm, for supervised learning, which is used in classification of large amount of data

• **Image Processing :** The various modifications done on a raw image to make it suitable for the training model.

• **Training-set :** This is the data set over which CNN model is trained. The predictions are completely dependent on the training-data set.

The main objective of the software is to classify the gestures and label them with one of the various categories already defined while training the dataset. With the help of deep learning, we have then tested the same for some data . There are some previously captured gestures on which system has been trained and these gestures have been labeled with the text associated with them. To extract the features efficiently, multiple copies of each gesture image have been created, then the CNN model is trained. The required gesture has to be signaled in front of the camera connected to the system, which will be recorded and matched and then using CNN algorithm, it will be classified.

2.2 SOFTWARE REQUIREMENTS

Table 2.1 : Software Requirements

Software	Minimum Requirements
Python	Version 3.0 or Higher
Operating System	Single Core 1.0 GHzor Higher
Tensorflow	Version 2.2 or Higher
Keras	Version 2.3.0 or Higher
OpenCV	Version 3.4 or Higher
Graphics Card	64Mb or Higher
Disk Space	650 MB or Higher
RAM	800MB or Higher
GNS3	Version 2.2.0 or Higher

HARDWARE

Table 2.2 : Hardware Requirements

Hardware	Minimum Requirements
Any Computing Device	2 GHz minimum, multi-core processor
Disk Space	At least 1 GB
Memory (RAM)	At least 2GB, preferably higher
Camera	2 MegaPixel or Higher

FUNCTIONAL REQUIREMENTS

The main objective of the software is to translate the sign language into text. The software is trained and developed to translate even words as initially, only one character was translated at a time. By concatenating various characters , words can be formed as well and on the output window, these formed words will be displayed. Pre-processing of the captured images is done. The system then modifies the images captured and trains the model to classify the signals in one of these defined labels.

NON-FUNCTIONAL REQUIREMENTS

To facilitate communication for the disabled is the sole purpose of this software. The previous available devices were slow and inefficient. Thus, to translate the signs accurately and at a relatively faster rate, this software is built. It is designed efficiently such that it can be modified easily making it easy to maintain.

2.4 EXPECTED HURDLES

Due to its multi-modal nature, the recognition of continuous sign language is a challenging task. Both manual and non-manual features must be used for recognition of sign language.

Manual features include hand shape, position and movement. Non-manual features are related to facial expressions, head pose and lip shape.To interpret a sign, these well-defined manual and non manual features are used parallely, Sentencing of the sign language to its subunits must also be done

CHAPTER 3
SYSTEM DESIGN
3.1 DESIGN APPROACH

The network software used is GNS3.

GNS3 (Graphical Network Simulator) is an open source software that simulates complex networks.All of this is done without having dedicated network hardware such as routers and switches. GNS3 is able to run real Cisco IOS images using Dynamips in GNS3 infrastructure. Cisco IOS images could be used more specifically with the back end hypervisor software through this network simulator. The following emulators are used to run the very same operating systems as in real networks:

§ Dynamips, the well-known Cisco IOS emulator.- It enables network devices such as Cisco Routers to work in a virtual environment.

§ VirtualBox, runs desktop and server operating systems

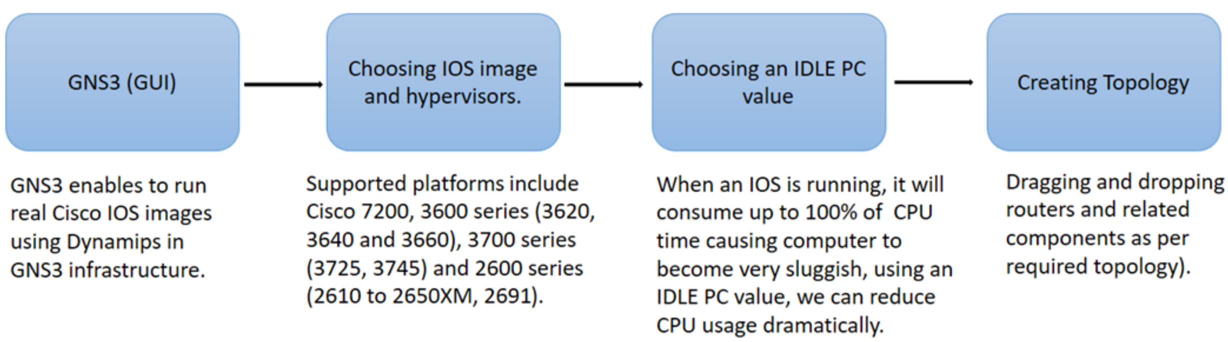


Figure 2.0: Basic steps involved in GNS3 setup

3.2 DETAIL DESIGN

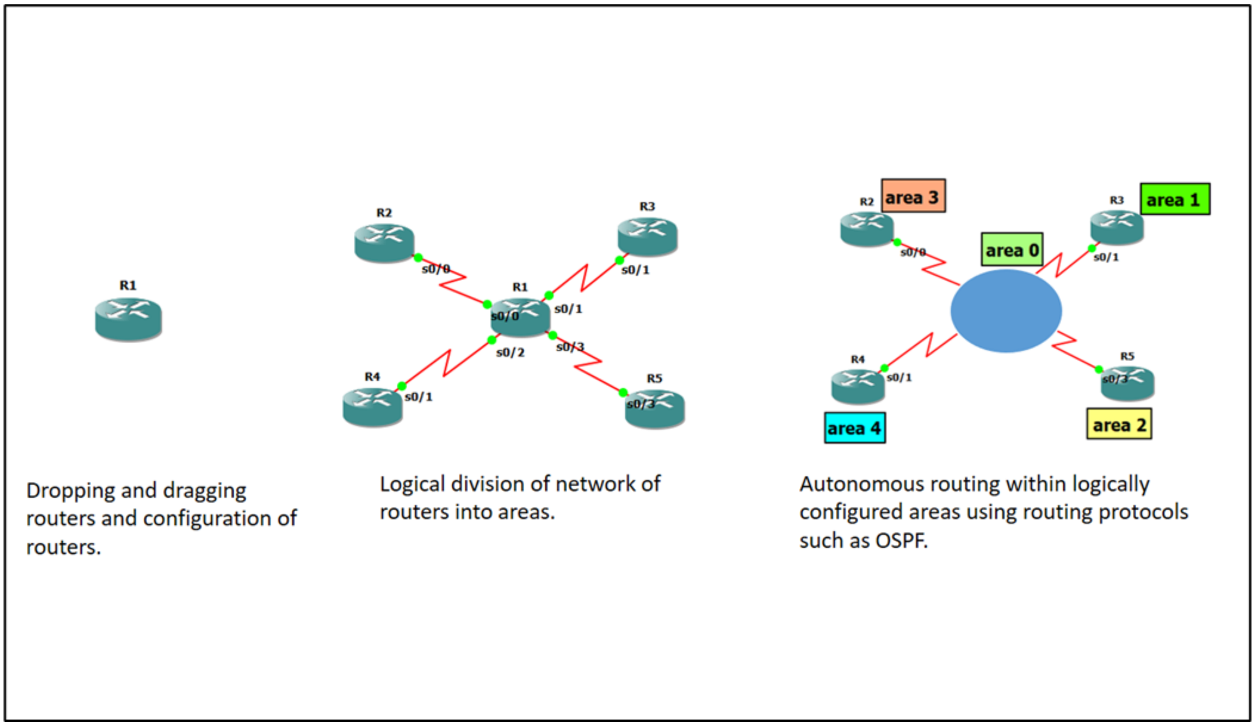


Figure 2.1: Configuration and logical division of routers

The implementation of Sign to Text Converting Model is done as follows:

· **Creating a user interface-** A user friendly User Interface is created providing buttons for various operations including adding more gestures, Training the model.

· **Creating a topology in GNS3**- It includes dragging and dropping routers and performing initial configuration of them and Distributing IP routing using OSPF (Open Shortest Path First).

· **Connecting GNS3 topology to host computer**- The topology thus formed could be connected to the local host (other devices connected in LAN could also be included).

· **Virtual server creation**- Virtual server is created on PC to host the website (user interface).

· **Web Hosting**-Website is hosted on virtual server using IIS Manager.

· **Website accessing** – after the network gets fully interconnected, the website could be accessed from any component of the network.

3.3 SYSTEM DESIGN USING DFDs

3.3.1 Zero level DFD

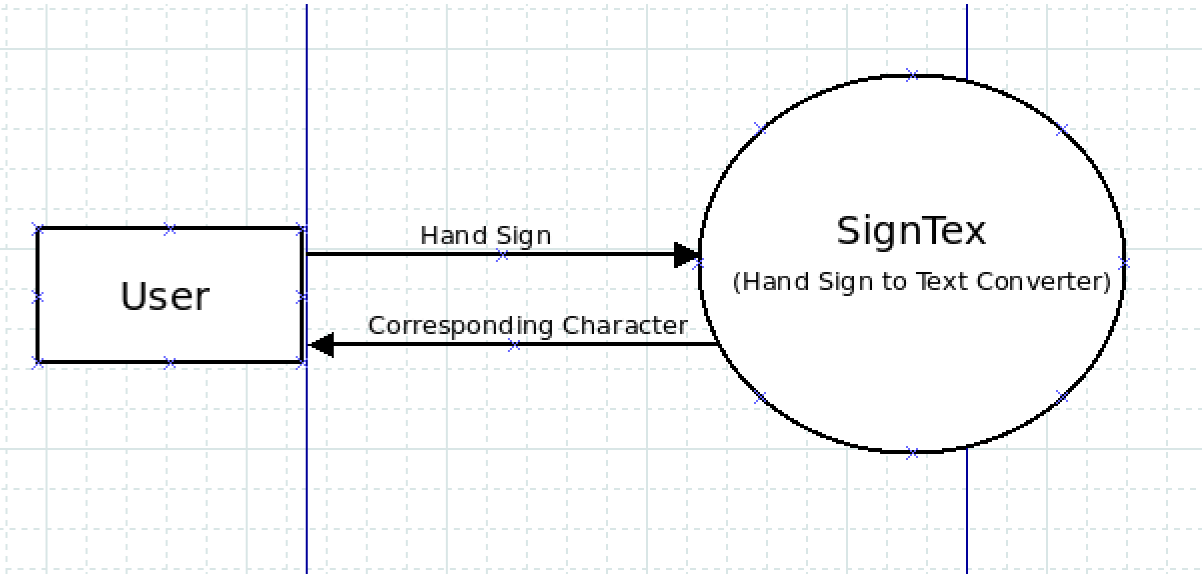


Figure 3.0: Zero Level DFD

3.3.2 First level DFD

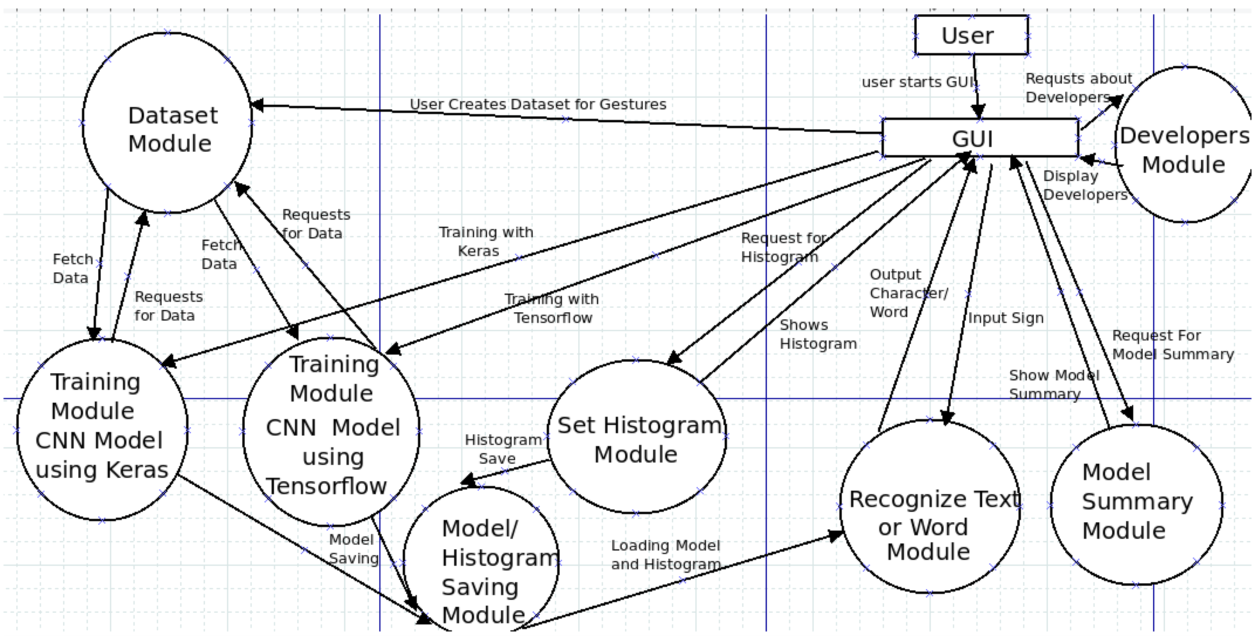


Figure 3.1: First Level DFD

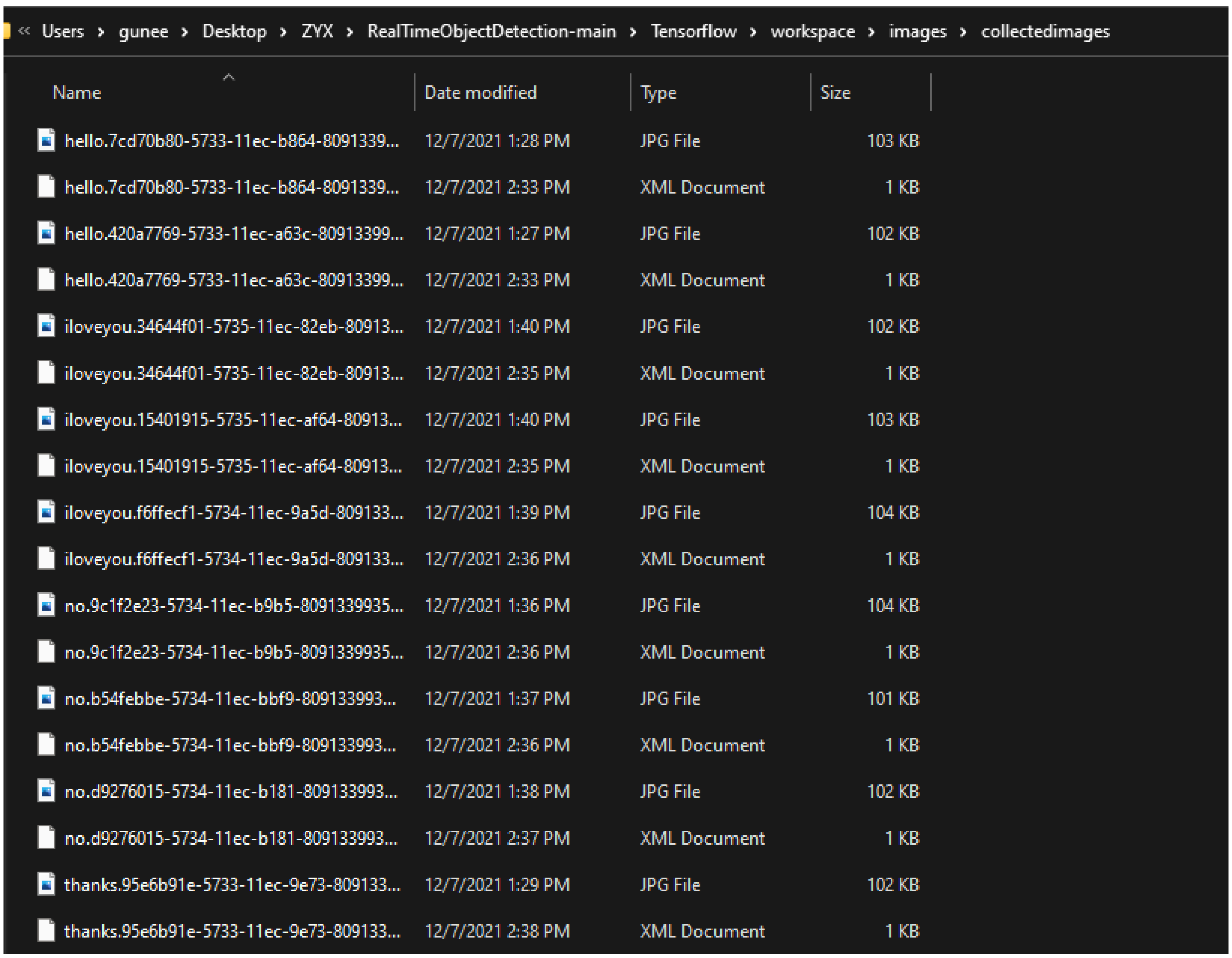
3.4 USER INTERFACE DESIGN

An User Interface is created which allows the User to add new gestures with some specific meaning, and the option to train the model with the added features. The created UI also directs the user to the ”hand recognition” window, which will further be used to do gestures.

3.5 DATABASE DESIGN

On the topic of database design for American Sign Language to text. The project is about designing a database for the American Sign Language alphabet, numbers and words.

The project will require the student to do research on the American Sign Language alphabet, numbers and words. A database is designed that stores all these data elements. The signs are stored in .xml file from which feature extraction is made easy. Jpg images are converted to XML files using labelmg.


« Users › gunee › Desktop › ZYX › RealTimeObjectDetection-main › Tensorflow › workspace › images › collectedimages

Name	Date modified	Type	Size
hello.7cd70b80-5733-11ec-b864-8091339...	12/7/2021 1:28 PM	JPG File	103 KB
hello.7cd70b80-5733-11ec-b864-8091339...	12/7/2021 2:33 PM	XML Document	1 KB
hello.420a7769-5733-11ec-a63c-80913399...	12/7/2021 1:27 PM	JPG File	102 KB
hello.420a7769-5733-11ec-a63c-80913399...	12/7/2021 2:33 PM	XML Document	1 KB
iloveyou.34644f01-5735-11ec-82eb-80913...	12/7/2021 1:40 PM	JPG File	102 KB
iloveyou.34644f01-5735-11ec-82eb-80913...	12/7/2021 2:35 PM	XML Document	1 KB
iloveyou.15401915-5735-11ec-af64-80913...	12/7/2021 1:40 PM	JPG File	103 KB
iloveyou.15401915-5735-11ec-af64-80913...	12/7/2021 2:35 PM	XML Document	1 KB
iloveyou.f6ffecf1-5734-11ec-9a5d-809133...	12/7/2021 1:39 PM	JPG File	104 KB
iloveyou.f6ffecf1-5734-11ec-9a5d-809133...	12/7/2021 2:36 PM	XML Document	1 KB
no.9c1f2e23-5734-11ec-b9b5-8091339935...	12/7/2021 1:36 PM	JPG File	104 KB
no.9c1f2e23-5734-11ec-b9b5-8091339935...	12/7/2021 2:36 PM	XML Document	1 KB
no.b54febbe-5734-11ec-bbf9-809133993...	12/7/2021 1:37 PM	JPG File	101 KB
no.b54febbe-5734-11ec-bbf9-809133993...	12/7/2021 2:36 PM	XML Document	1 KB
no.d9276015-5734-11ec-b181-809133993...	12/7/2021 1:38 PM	JPG File	102 KB
no.d9276015-5734-11ec-b181-809133993...	12/7/2021 2:37 PM	XML Document	1 KB
thanks.95e6b91e-5733-11ec-9e73-809133...	12/7/2021 1:29 PM	JPG File	102 KB
thanks.95e6b91e-5733-11ec-9e73-809133...	12/7/2021 2:38 PM	XML Document	1 KB

Figure 3.2: Databases and XML files

3.6 METHODOLOGY

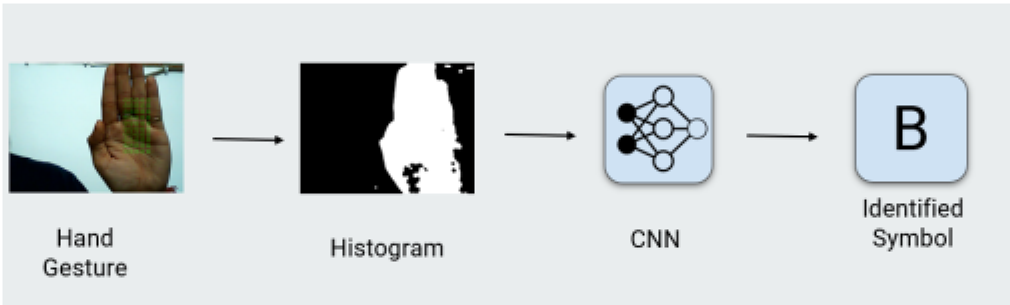


Figure 3.3: Approach Used

The model is trained for a total of 44 gestures, including 26 English alphabets, 10 numeric digits and 8 other commonly used symbols. Each gesture's 1200 images have been captured and then each image is flipped, making a total of 2400 images for each gesture. The images have then been resized to 50*50 pixels and converted to grayscale.

To identify the skin of the hand of the speaker and to separate it from the background, a histogram is created. On each gesture, the Neural Network is trained using Keras. The person uses the sign language to be recorded in the camera of the device with the software,whenever the translator is to be used and then fetches it into the Neural Network and the sign is classified.

3.6.1 Pre-processing the Data

3.6.1.1 HISTOGRAM CREATION

To generate a histogram, the OpenCV library has been used that will separate the hand gestures from the background. 50 squares in the form of 5*10 are displayed for this purpose and the hand must cover all the squares. Then the image is captured and a histogram is plotted of the area covering the squares.

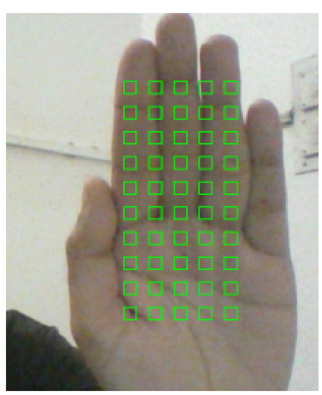


Figure 3.4: Hand detection

3.6.1.2 IMAGE PROCESSING

Figure 3.5: Histogram of the hand

Using the intensities obtained from the histogram, the images which are captured of the gestures are then processed. The images are resized to 50*50 pixels and then these are converted to gray scale. Along the vertical axis, each image is then flipped

3.6.1.3 CONVOLUTION

This is the most important part in feature detection.With the feature detector, the image obtained is then convoluted to form the feature map. The image is convoluted with a number of features and hence a number of feature maps are present.Larger the number of features, better it is to classify the image.

$$(f * g)(t) = \int_{-\infty}^{\infty} f(\tau)g(t-\tau)d\tau$$

The main aim of convolution is to detect features.

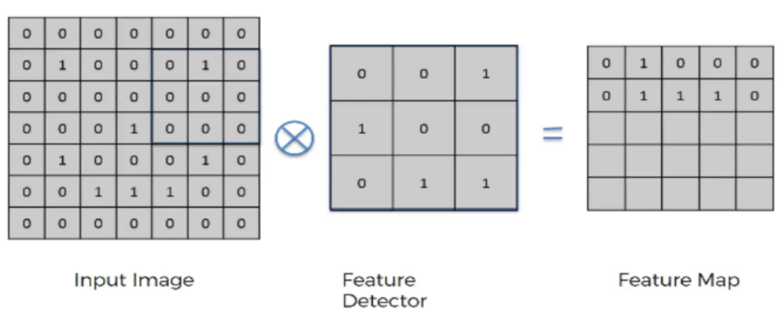


Figure 3.6: Convolution of a matrix

3.6.1.4 ReLU Layer

Then comes the Activation function. The Convoluted image is passed through this Activation function. The RECTIFIER FUNCTION is used as the activation function.This increases the non-linearity in the image.

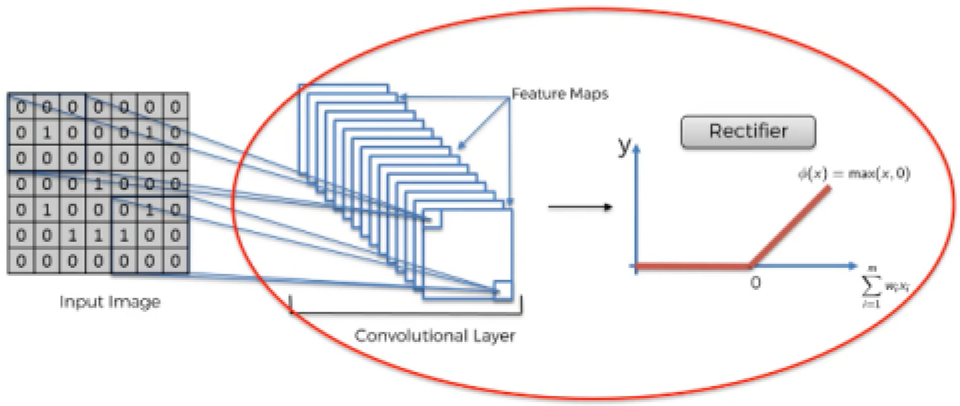


Figure 3.7: Activation function: Rectifier function

3.6.1.5 MAX POOLING

The convoluted result is contained in the feature map. A large number of such maps are there, hence enormous data. Also, one feature may differ in size, orientation in different images. Using Max Pooling, both these issues are resolved. A small grid is selected and then the maximum value is preserved, which reduces the size of data as well.

Figure 3.8: Max Pooling

3.6.1.6 FLATTENING

To use the pooled image as input in the next Artificial Neural Network, the features of the pooled image need to be flattened. These form the input layer of the Artificial Neural Network. The max pooling output is transformed into a column.

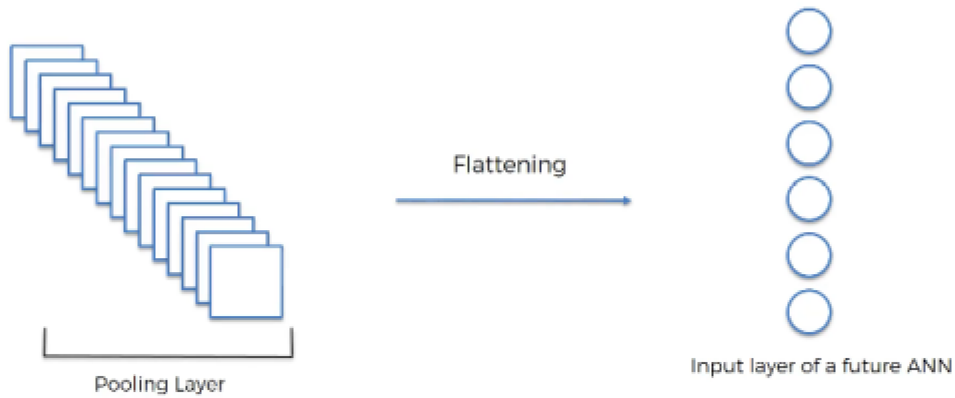


Figure 3.9: Flattening

3.6.2 Building the Model

CNN is an Artificial Neural Network that is used in Image recognition and processing. It detects specific features in the images and then classifies them accordingly based on the presence of those features.Our model consists of 5 layers in total, including 3 Convolutional layers and 2 fully connected Dense layers.

The first CNN layer consists of 16 neurons, the second layer consists of 32 neurons and the third layer consists of 64 neurons. The first dense layer uses the Rectifier function as the Activation function and consists of 128 neurons. Softmax function is used by the second dense layer as the Activation function.

3.6.2.1 ACTIVATION FUNCTIONS

Activation function is where calculations happen. It is the function that is applied to the sum of weighted inputs to the neuron. We have used two different activation functions in our neural network,

- **Relu Activation Function:** ReLU stands for Rectified Linear Unit. As there is no complicated math, it's cheap to compute. The model therefore takes less time to train or run. This is used to increase the non-linearity of the images. When dealing with small values, it can serve of a great use, as in our case.The formula for ReLu function is given by:

$$y=max(0,x)$$

- **Softmax Activation function**:Softmax is an activation function. It is frequently used in classifications. Softmax's output is large if the score is large. Its output is small if the score is small. The proportion is not uniform. Softmax is exponential and enlarges differences. The formula for Softmax function is given by:

$$Softmax(x_i) = \frac{e^{x_i}}{\sum_j e^{x_j}}$$

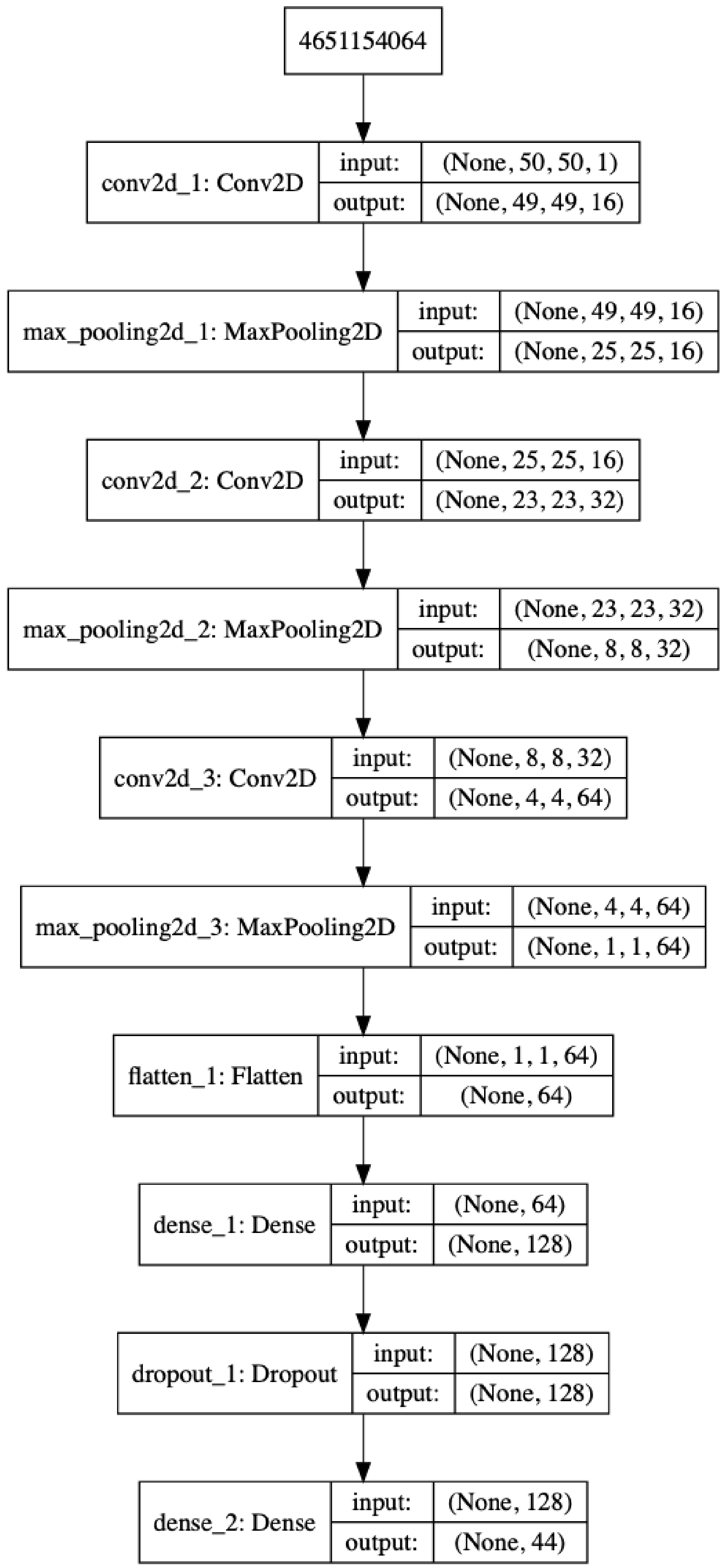


Figure 3.10: Model consisting of 5 layers

CHAPTER 4
IMPLEMENTATION, TESTING, AND MAINTENANCE

4.1 INTRODUCTION TO LANGUAGES, IDE’S, TOOLS AND TECHNOLOGIES USED FOR IMPLEMENTATION

4.1.1 LANGUAGES USED FOR IMPLEMENTATION

Python: It is a programming language used for integrating systems efficiently and lets users work quickly. It is used for various purposes.Python is a programming language that allows programmers to work in an interactive environment.

Python is a dynamic, interpreted programming language that runs on many platforms including Windows, Linux/Unix, Mac OS X, and has been ported to Java and .NET virtual machines. It is an object-oriented language with high level data types. The language features are designed around major programming concepts of simplicity, readability, and conciseness. Python's syntax is very easy to learn.

Python has the triad of being easy to read for beginners, but also equally powerful for professional programmers who want to use it in their development stack.

Libraries used: OpenCV, Keras, Tensorflow

OpenCV: Python is a programming language that is widely used for data science and machine learning. It has libraries that have been developed over the years to solve a wide variety of problems. OpenCV is a library for computer vision applications.The OpenCV library helps in the fields of image processing, video transmission, and recognition. Some of its functions include distance transformations, object oriented transforms, and 2D image filters.OpenCV was initially developed by Intel scientists but it has been open sourced since 1999 so everyone can use it freely.

Keras: Keras allows programmers to define deep neural network architectures by describing the data flow between layers. This significantly reduces the amount of coding needed to build an artificial neural network compared to other frameworks such as

TensorFlow, CNTK, Theano. This way, programmers can focus on training their models instead of spending time on code optimization and backpropagation algorithms.

Moreover, Keras provides a variety of high-level building blocks for deep neural networks: convolutional networks (CNN), recurrent networks (RNN), and generative models such as generative adversarial networks (GAN).

Tensorflow: Python is a programming language that is widely used. It's popular because it's versatile and easy to learn. With Python, you can create scripts, apps, games and more.The TensorFlow is the most popular framework for deep learning applications in Python. It has been used by Google to power many of its machine-learning products and services, including speech recognition on Google Home devices, automatic image labeling on Google Photos, the ability to search YouTube videos using spoken queries, and RankBrain for search results on Google Search.

HTML: HTML is a markup language and code for websites and other webpages.HTML is not just a programming language, it's also a markup language. HTML tags tell the browser how to present the content on the page.HTML is an acronym for HyperText Markup Language. It is the language that web pages are written in to provide formatting and content. HTML is not a programming language, but it can be used to embed one. It provides tags that indicate where text should be displayed on the page, images, links to other pages, and videos. The HTML tags also indicate how information should be formatted when it's viewed by a web browser.

Networking: GNS3 is a graphical network simulator for Cisco or Juniper Networks routers. This software enables users to configure, explore, and troubleshoot the large-scale networks. GNS3 can be used to create lab setups with various network topologies. It offers an easy-to-use interface that contains several handy features like drag-and-drop, copy/paste, cut/paste, etc.

4.1.2 IDE's USED

Spyder: Spyder is a free and open source scientific environment written in Python, for Python, and designed by and for scientists, engineers and data analysts. It features a unique combination of the advanced editing, analysis, debugging, and profiling functionality of a comprehensive development tool with the data exploration, interactive execution, deep inspection, and beautiful visualization capabilities of a scientific package.

Anaconda: Anaconda is a distribution of the Python and R programming languages for scientific computing (data science, machine learning applications, large-scale data processing, predictive analytics, etc.), that aims to simplify package management and deployment. The distribution includes data-science packages suitable for Windows, Linux, and macOS.

Notepad++: Notepad++ is a free (as in "free speech" and also as in "free beer") source code editor and Notepad replacement that supports several languages. Running in the MS Windows environment, its use is governed by the GNU General Public License.

4.1.3 TOOLS AND TECHNOLOGIES USED FOR IMPLEMENTATION

GNS3: GNS3 is a graphical network simulator for Cisco or Juniper Networks routers. This software enables users to configure, explore, and troubleshoot the large-scale networks. GNS3 can be used to create lab setups with various network topologies. It offers an easy-to-use interface that contains several handy features like drag-and-drop, copy/paste, cut/paste, etc.

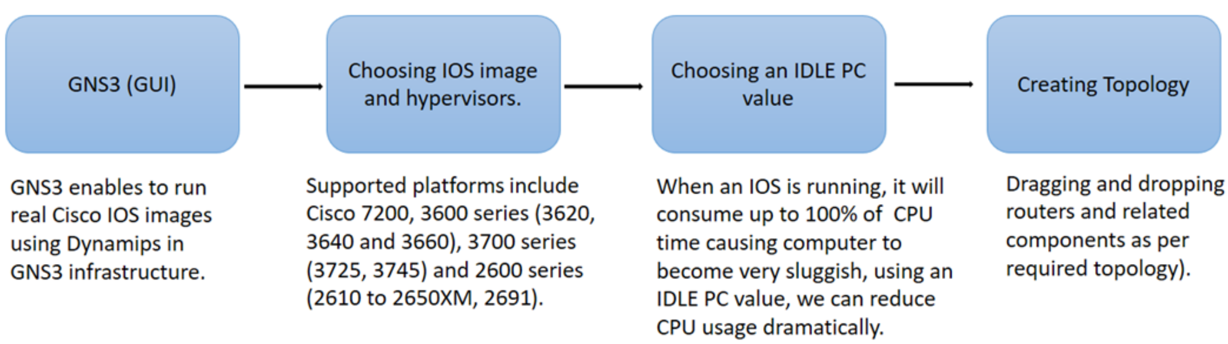


Figure 4.0: Steps involved in simulator setup

4.2 CODING STANDARDS OF LANGUAGE USED

PEP 8 -- Style Guide for Python Code

PEP 8 is a coding style guide that sets out to clarify Python code readability. The fundamental idea of PEP 8 is that code doesn't have to be perfect, but it does have to be readable.

Naming conventions for variables and constants, Indentation, Lines and spacing, Coding style are the various rules in the style guide.

PEP:	8
Title:	Style Guide for Python Code
Author:	Guido van Rossum <guido at python.org>, Barry Warsaw <barry at python.org>, Nick Coghlan <ncoghlan at gmail.com>
Status:	Active
Type:	Process
Created:	05-Jul-2001
Post-History:	05-Jul-2001, 01-Aug-2013

Figure 4.1: Pep Coding rules

4.3 Project Scheduling using various tools such as PERT, GANTT charts, OpenPROJ etc.

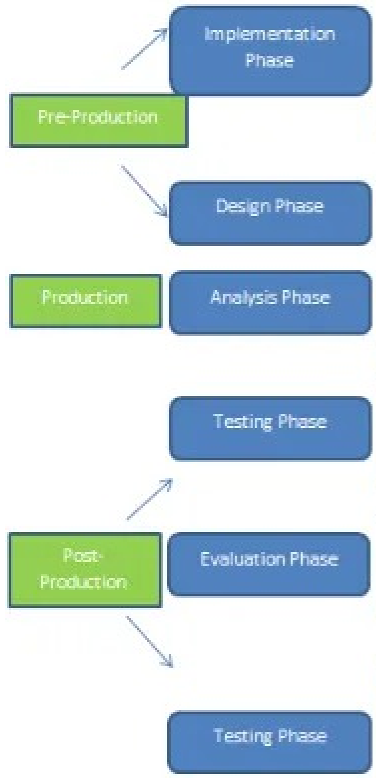


Figure 4.2: Project Scheduling

4.4 Testing Techniques and Test Plans

Data is split in train, test and validation sets, in 80:10:10.

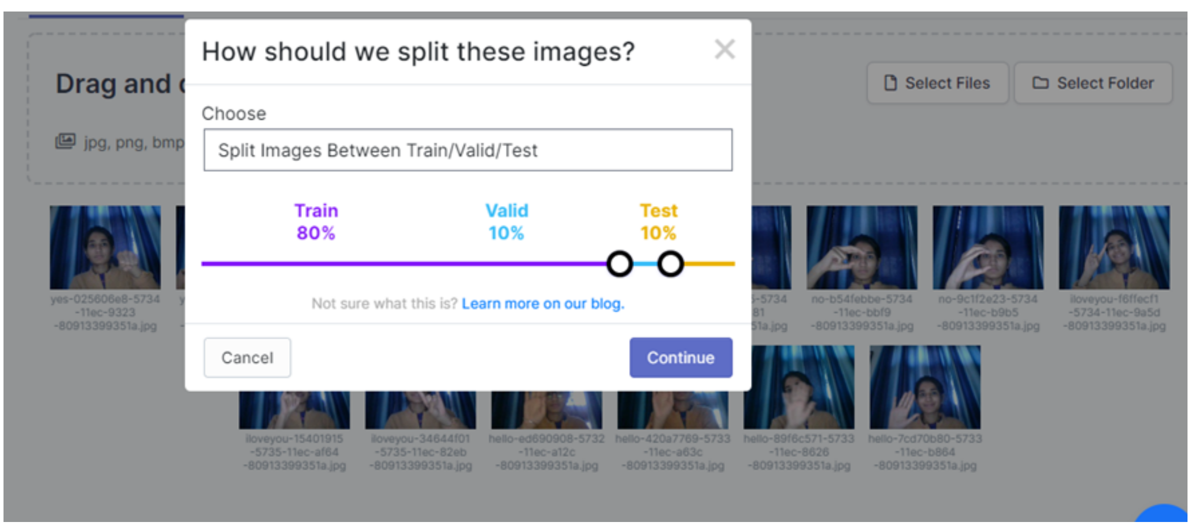


Figure 4.3:Splitting of Data

CHAPTER 5:RESULTS AND DISCUSSIONS

5.1 User Interface Representation

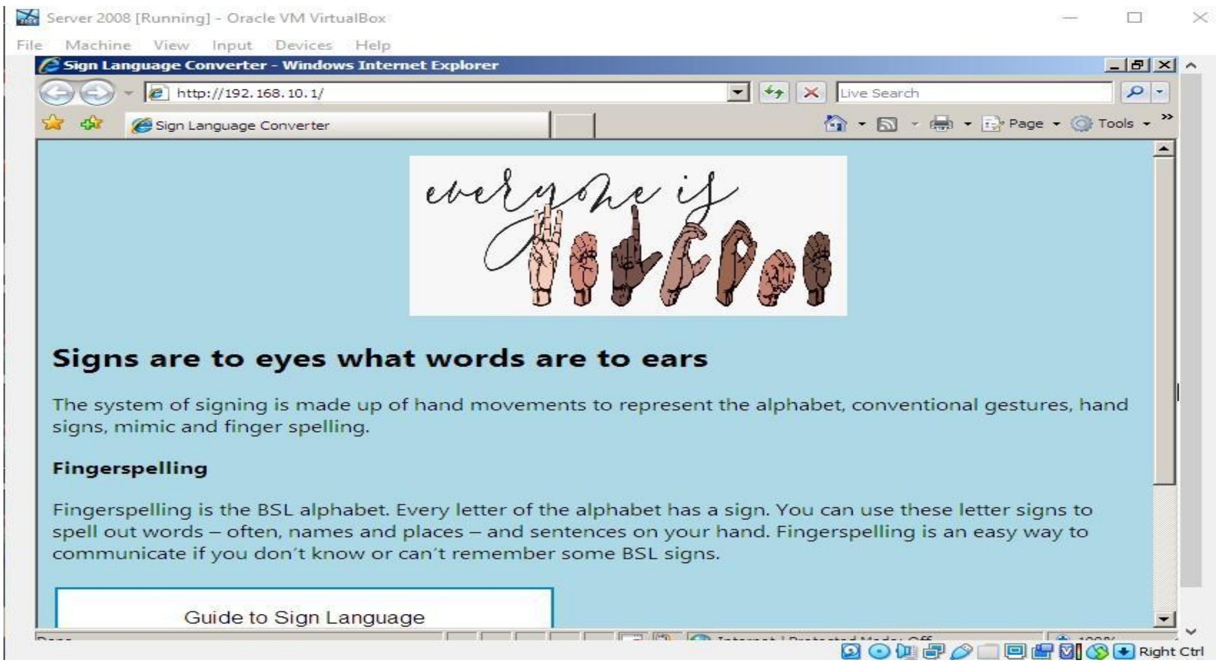


Figure 5.0: Final Representation

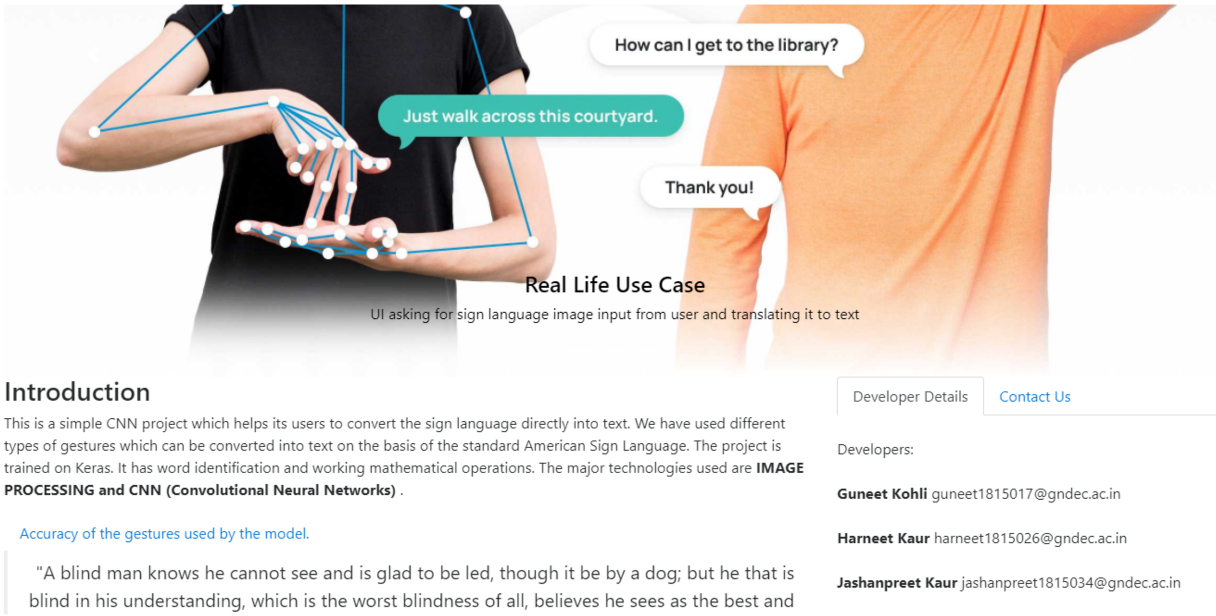


Figure 5.1: User Interface

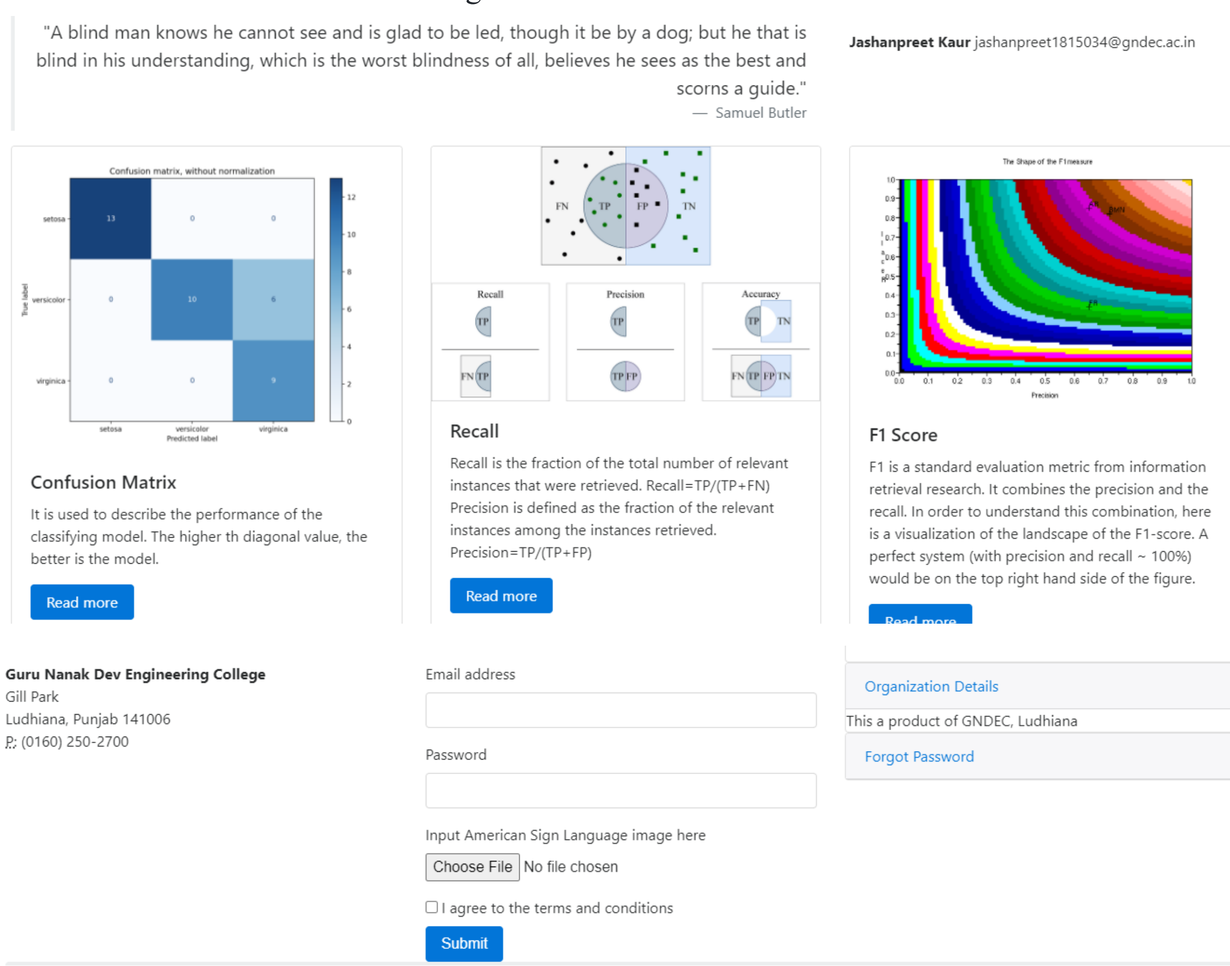


Figure 5.2: User Interface

5.1.1 Brief Description of Various Modules of the system

The proposed system contains four modules such as: pre-processing and hand segmentation, feature extraction, sign recognition and sign to text.

Pre-Processing:

Pre-processing consists of image acquisition, segmentation and morphological filtering methods.

Image acquisition:

This is the first step of pre-processing. This is the process of sensing an image. So in an image acquisition, the image is sensed by "illumination". It will also involve pre-processing such as scaling. In image acquisition the image will be taken from the database.

Segmentation:

Segmentation is the process in which an image is converted into small segments so that the more accurate image attribute can be extracted. If the segments are properly autonomous (two segments of an image should not have any identical information) then representation and description of image will be accurate and while taking rugged segmentation, the result will not be accurate. Here the Segmentation of hands is carried out to separate objects and the background. The segmented hand image is represented certain features

Morphological filtering:

The image components are extracted by Morphological Filtering tools which are useful for representation and description of shape. Definitely the output of this process is image attribute.

The features extracted from the segmentation operation used for gesture recognition. The smooth contour is obtained by removing the noise from the images with Morphological filtering techniques. The pre-processing operation is done on the stored database.

Feature Extraction:

The reduction of data dimensionality by encoding related information in a compressed representation and removing less discriminative data is called the Feature extraction Technique. Feature extraction is vital to gesture recognition performance. Therefore, the selection of which features to deal with and the extraction method are probably the most significant design decisions in hand motion and gesture recognition development. Here principal component is used as main features

Sign Recognition:

Sign recognition is a dimensionality reduction technique based on extracting the desired number of principal components of the multi-dimensional data.

Each gesture is represented as a column vector in the training phase. These gesture vectors are then normalized with respect to the average gesture. Next, the algorithm finds the eigenvectors of the covariance matrix of normalized gestures by using a speed up technique that reduces the number of multiplications to be performed. The corresponding gesture space projections were obtained by the eigenvector matrix then multiplied by each of the gesture vectors.

In the recognition phase, a subject gesture is normalized with respect to the average gesture and then projected onto gesture space using the eigenvector matrix. Lastly, Euclidean distance is computed between this projection and all known projections. The minimum value of these comparisons is selected for recognition during the training phase. Finally, the recognized sign is converted into appropriate text which is displayed on GUI.

Implementation in Client Server Environment

- Web hosting is the basic infrastructure that put a website on the internet. It is an online service that allows you to publish your website files onto the internet. Hosting is the practice of using a server to host a website.A web host, or web hosting service provider, is a business that provides the technologies and services needed for the website or webpage to be viewed in the Internet.
- Websites are hosted, or stored, on special computers called servers.
- When Internet users want to view your website, all they need to do is type your website address or domain into their browser.
- Their computer will then connect to your server and your webpages will be delivered to them through the browser.

Steps involved -

- **Creating a topology in GNS3**

    1. Adding routers and performing initial configuration of them.
    2. Assigning IP’s to the routers.
    3. Distributing IP routing using OSPF (Open Shortest Path First)

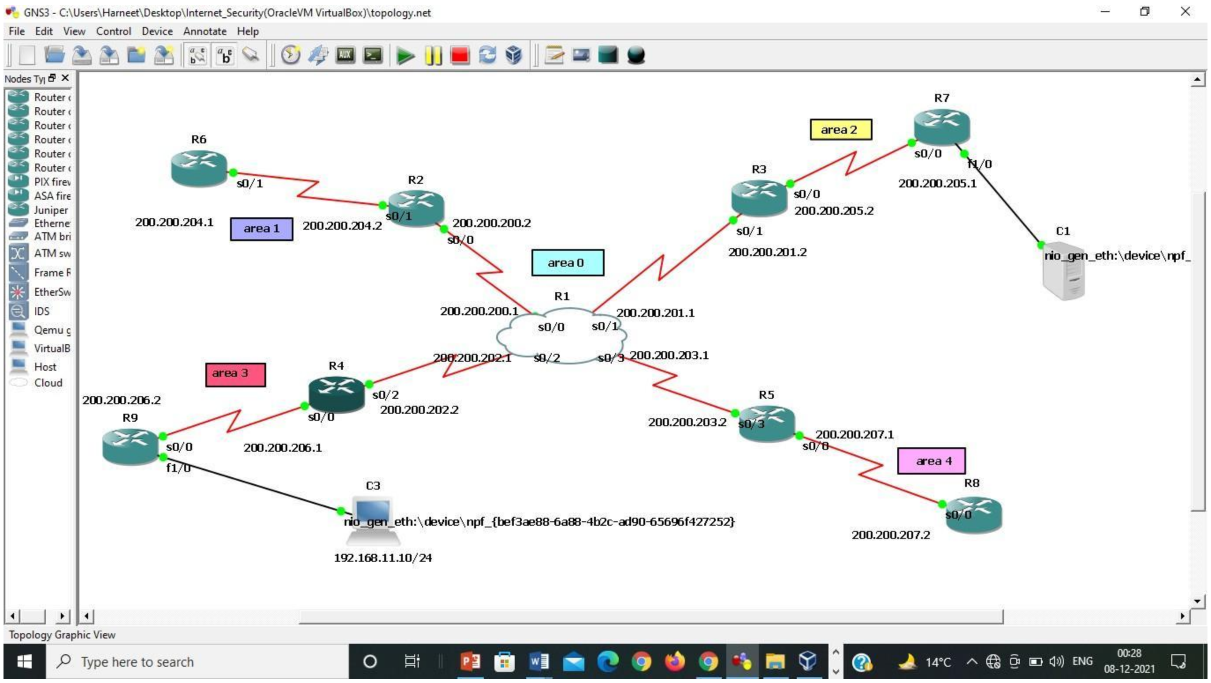


Figure 5.3: Dragging and dropping routers and related components (as per topology)

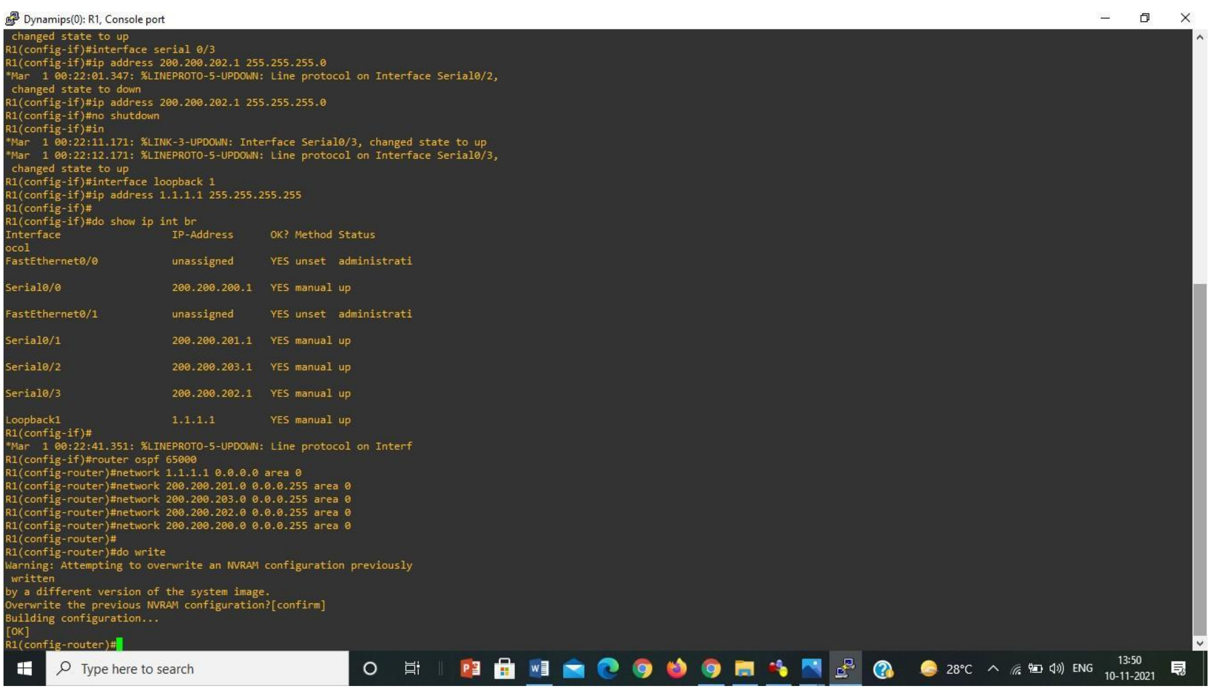


Figure 5.4: Assigning IP's to the routers accordingly as to create different areas.

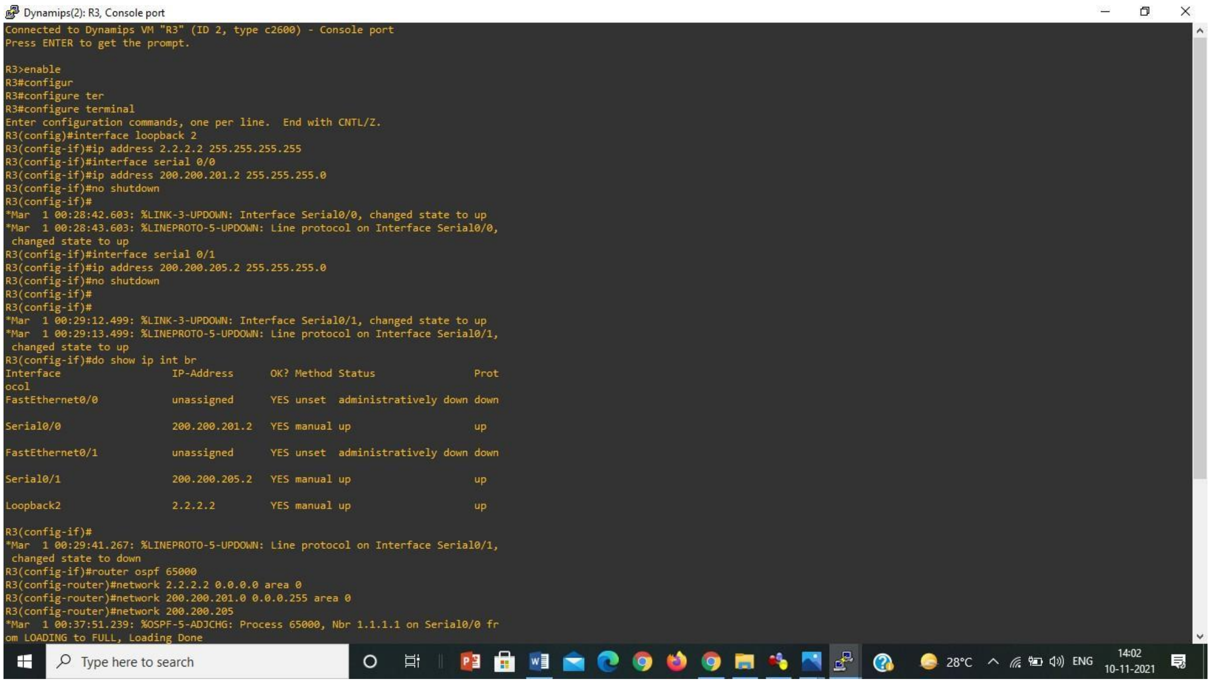


Figure 5.5: Assigning loopback to each router.

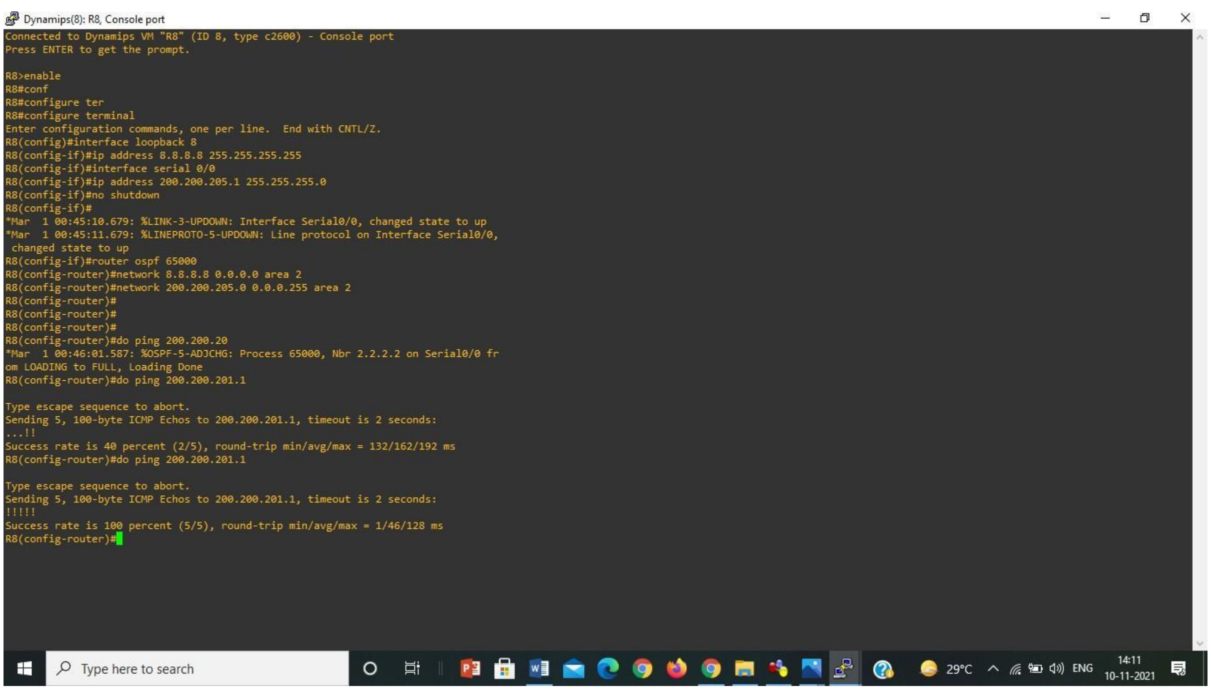


Figure 5.6: Interconnecting the areas using OSPF.

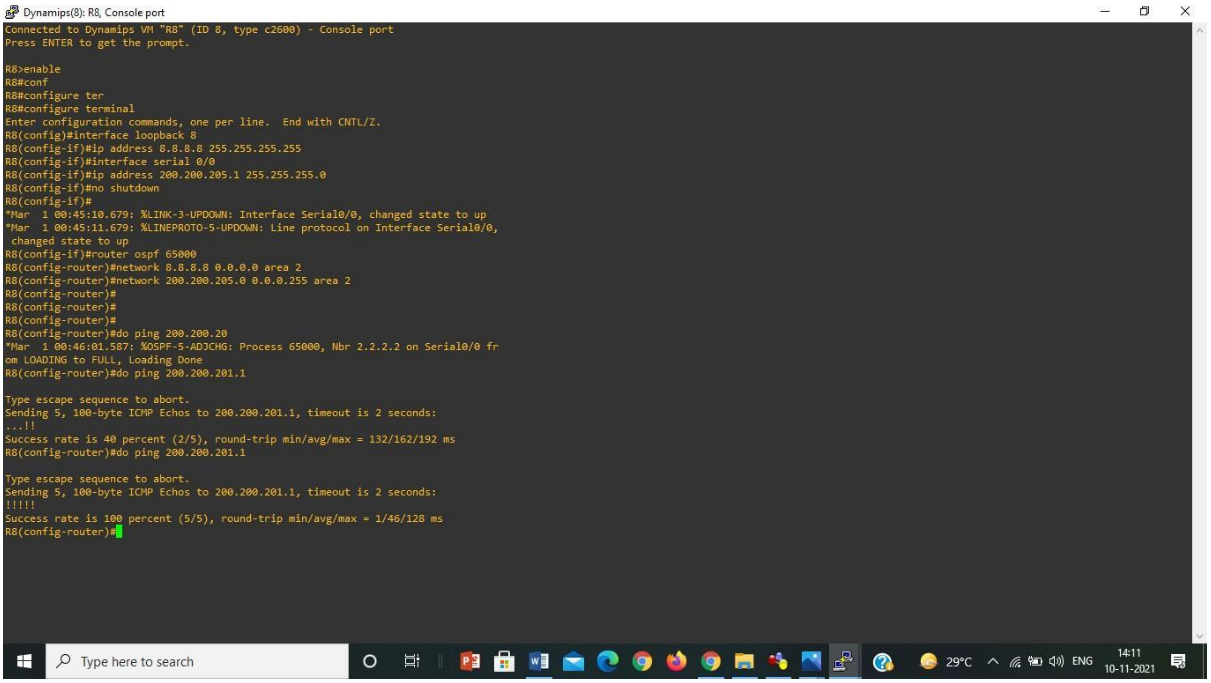


Figure 5.7: Pinging an IP from different area (to check connectivity)

Advantages of protocol being used – Open shortest Path First

1. The OSPF protocol is a link-state routing protocol, which means that the routers exchange topology information with their nearest neighbors. The topology information is flooded throughout the AS, so that every router within the AS has a complete picture of the topology.
2. The main advantage of a link state routing protocol like OSPF is that the complete knowledge of topology allows routers to calculate routes that satisfy particular criteria.
3. The protocol recalculates routes when network topology changes and minimizes the routing protocol traffic that it generates.
4. It provides a multi-level hierarchy called "area routing," so that information about the topology within a defined area of the AS is hidden from routers outside this area. This helps in an additional level of routing protection and a reduction in routing protocol traffic.

- **Connecting GNS3 topology to host computer**
    1. With the help of ‘Microsoft Loop-back adapter’, we can connect GNS3 router

to our host PC.

2. Virtual LoopBack adapter serves as physical connection from PC.

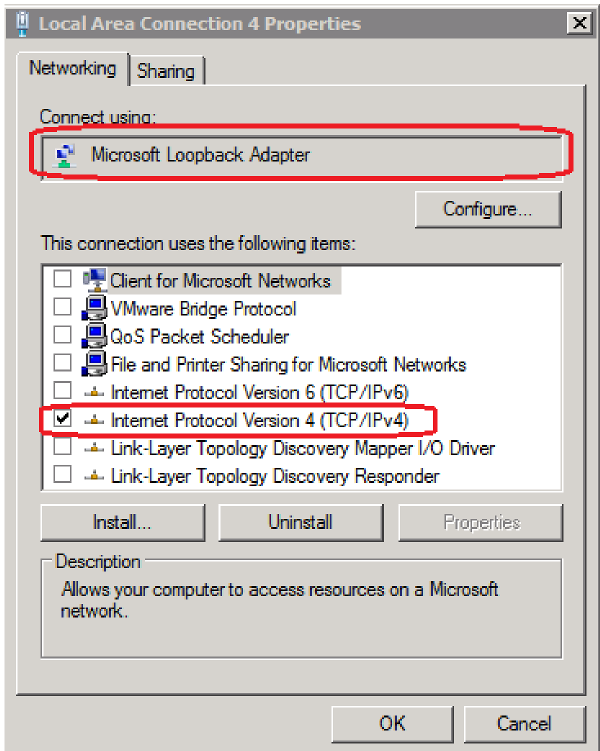


Figure 5.8: Configuring newly created adapter.

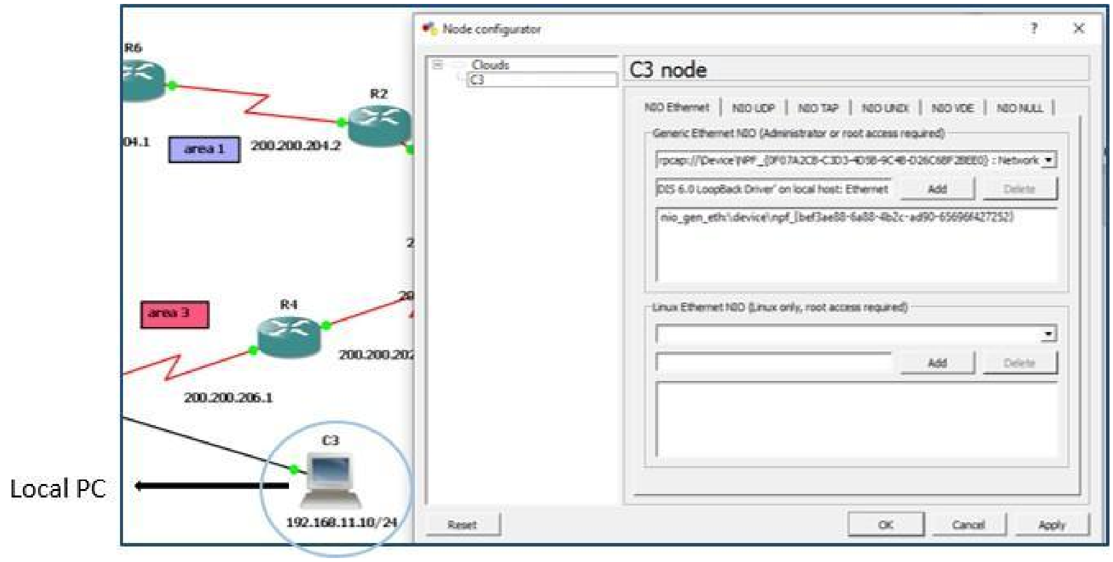


Figure 5.9: Attaching the host(created in topology)to our local PC.

- **Creating virtual server using Oracle VM VirtualBox.**

1. Virtual disk is created and Virtual machine is added.
2. Virtual CD/DVD disk file is chosen and browsed for the ISO image for providing bootable environment for the OS.

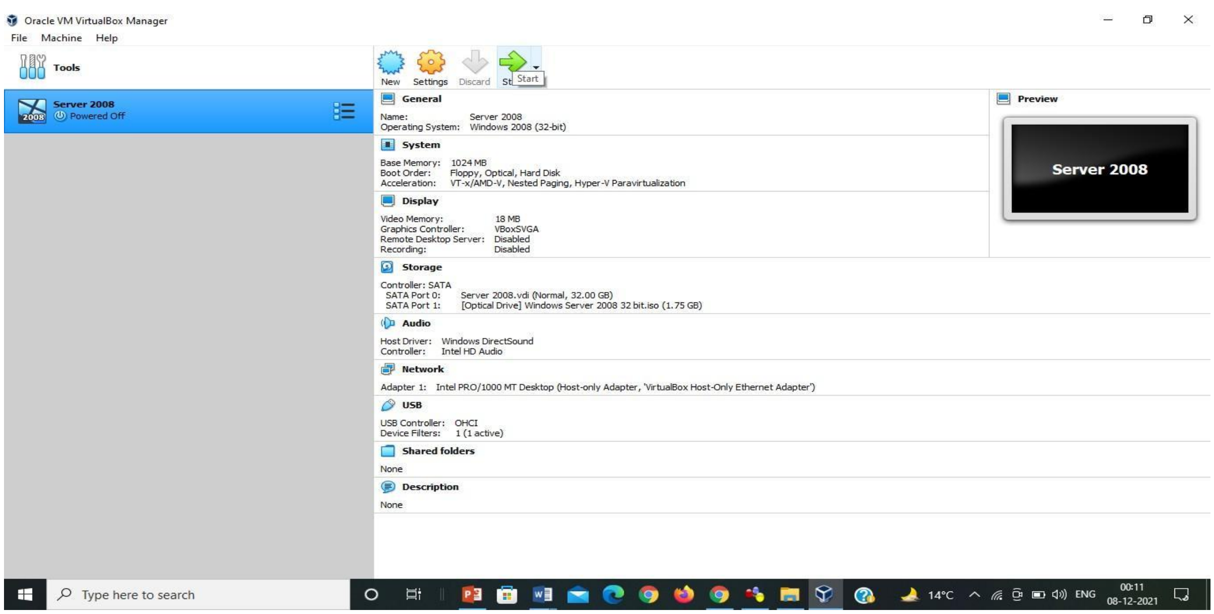


Figure 5.10 Creating a virtual Machine named Server 2008.

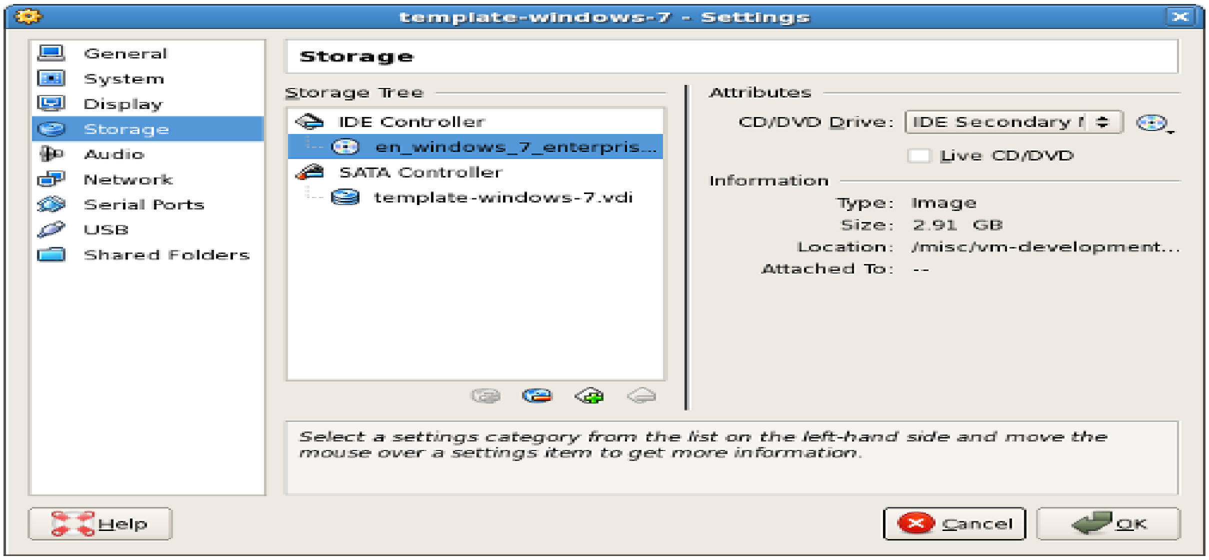


Figure 5.11 Browsing the windows server for booting the OS.

- **Adding website on Server's IIS Manager**

Internet Information Services (IIS) is a flexible, general-purpose web server from Microsoft that runs on Windows systems to serve requested HTML pages or files.

An IIS web server accepts requests from remote client computers and returns responses.

This basic functionality allows web servers to share and deliver information across networks (LAN and WAN)

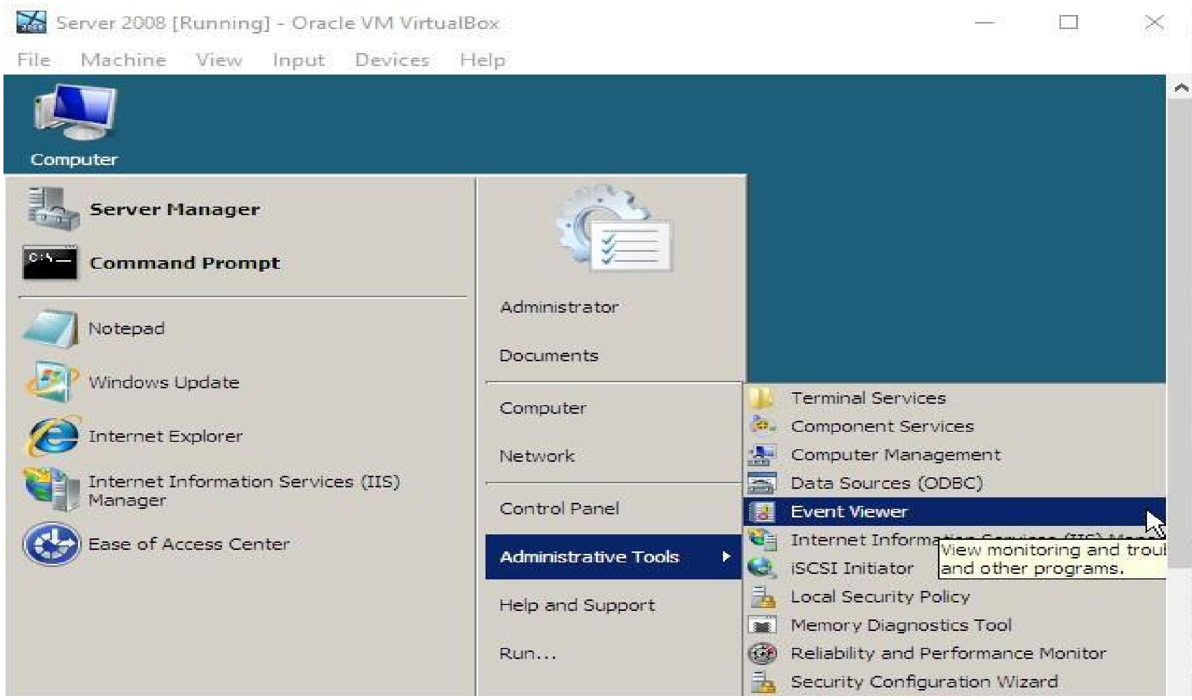


Figure 5.12: Internet Information Services (IIS)

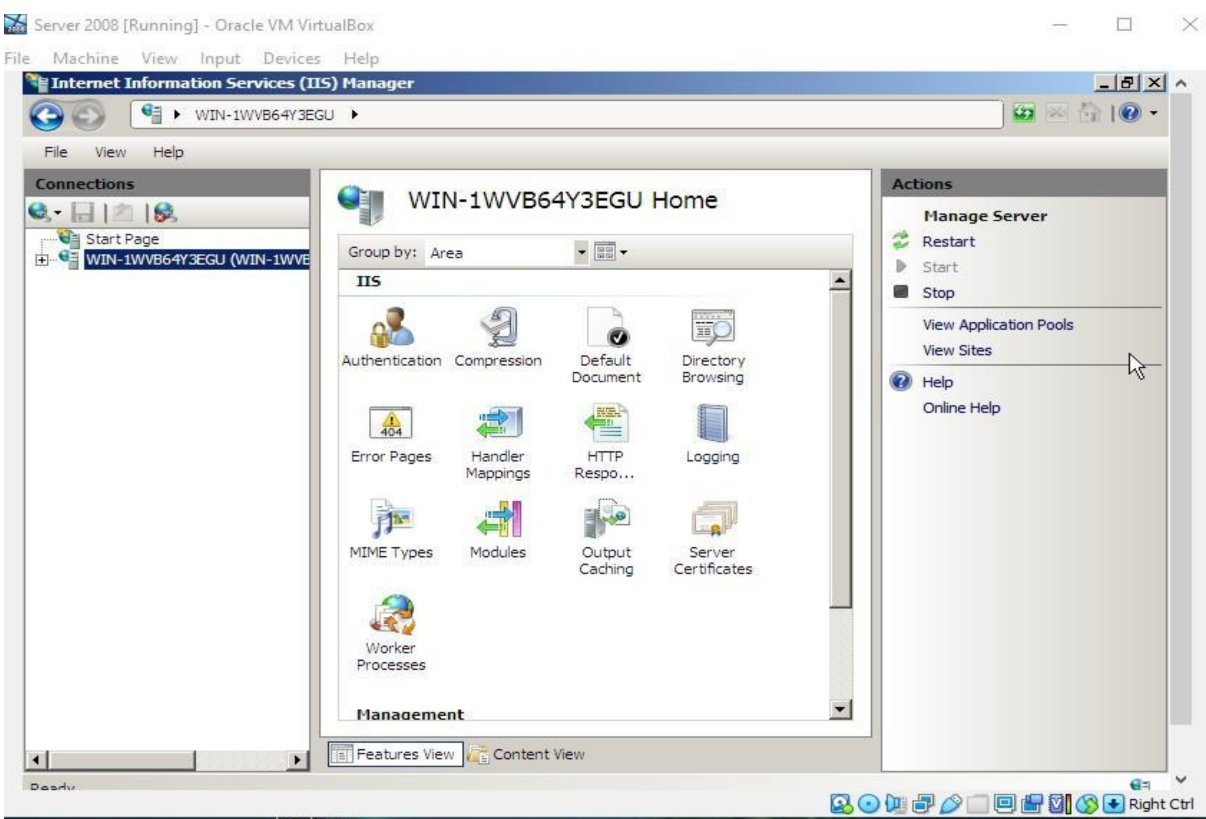


Figure 5.13: Accepting requests from remote client computers

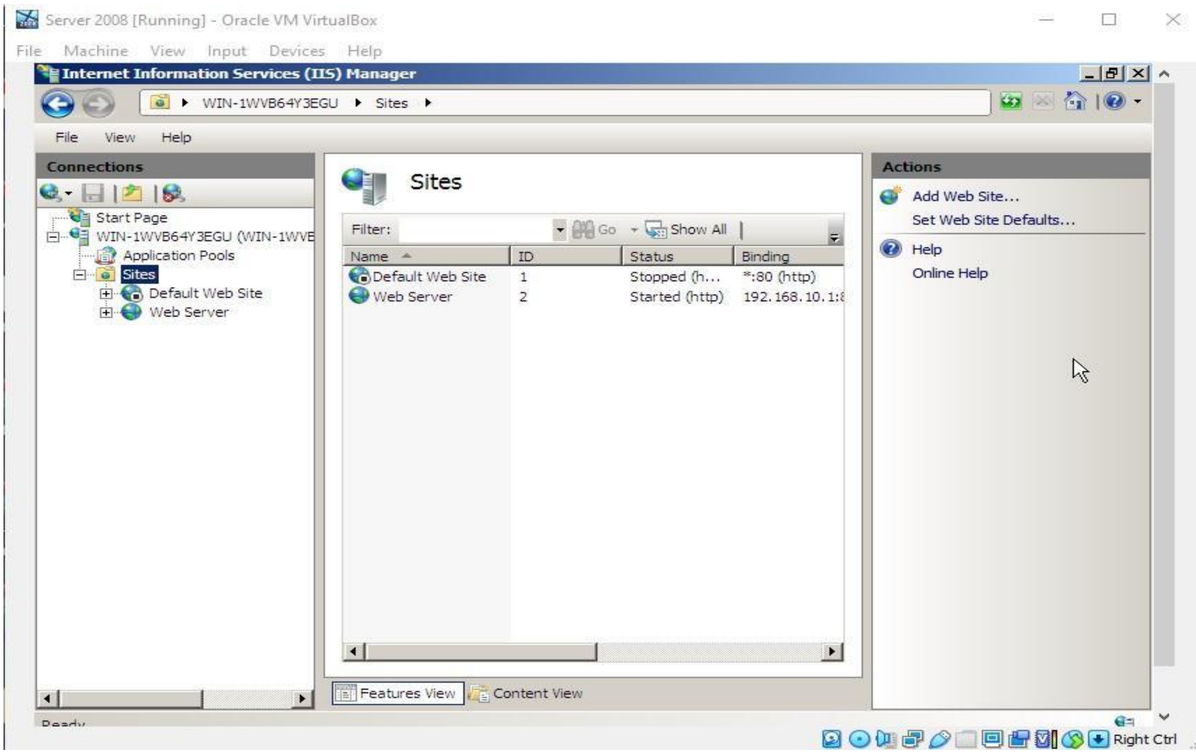


Figure 5.14: sharing and delivering information across networks

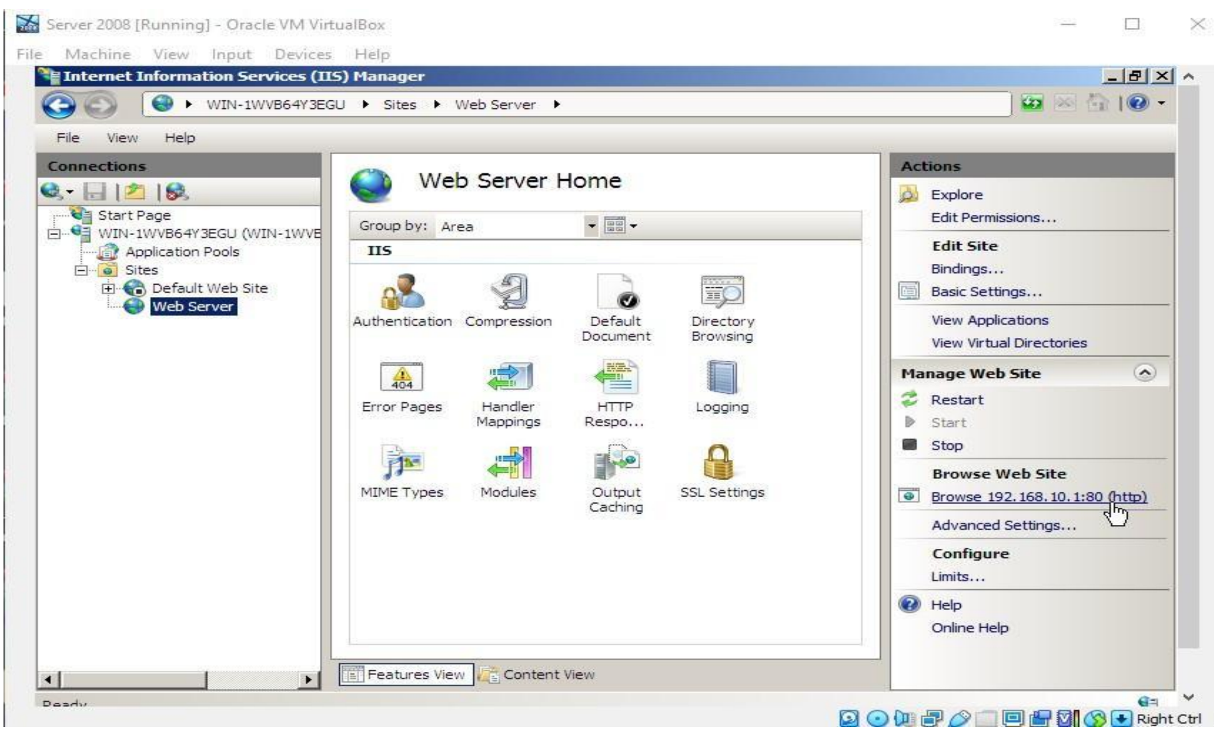


Figure 5.15: sharing and delivering information across networks

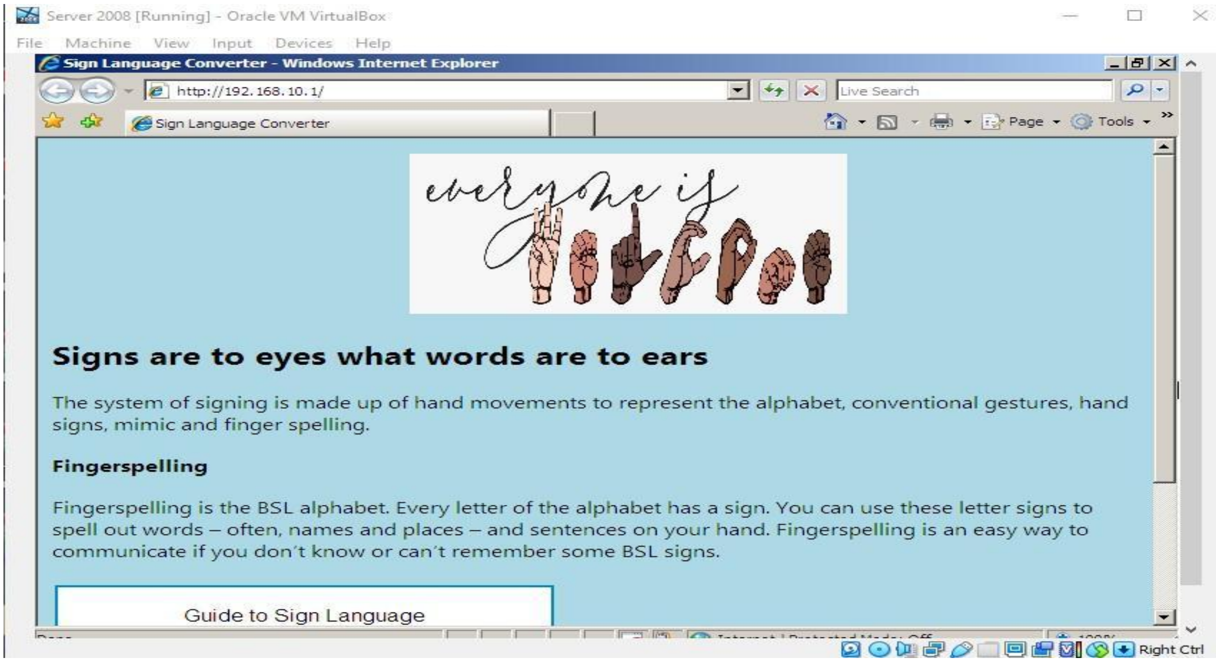


Figure 5.16: Browsing the website on Virtual server.

- **Pinging the Website from local PC and routers in the topology.**

  As the network becomes fully connected i.e the routers are interconnected with the local PC , website being hosted on a server (using the virtual server) could be accessed from any of the components connected in the network (routers as well as the local host).

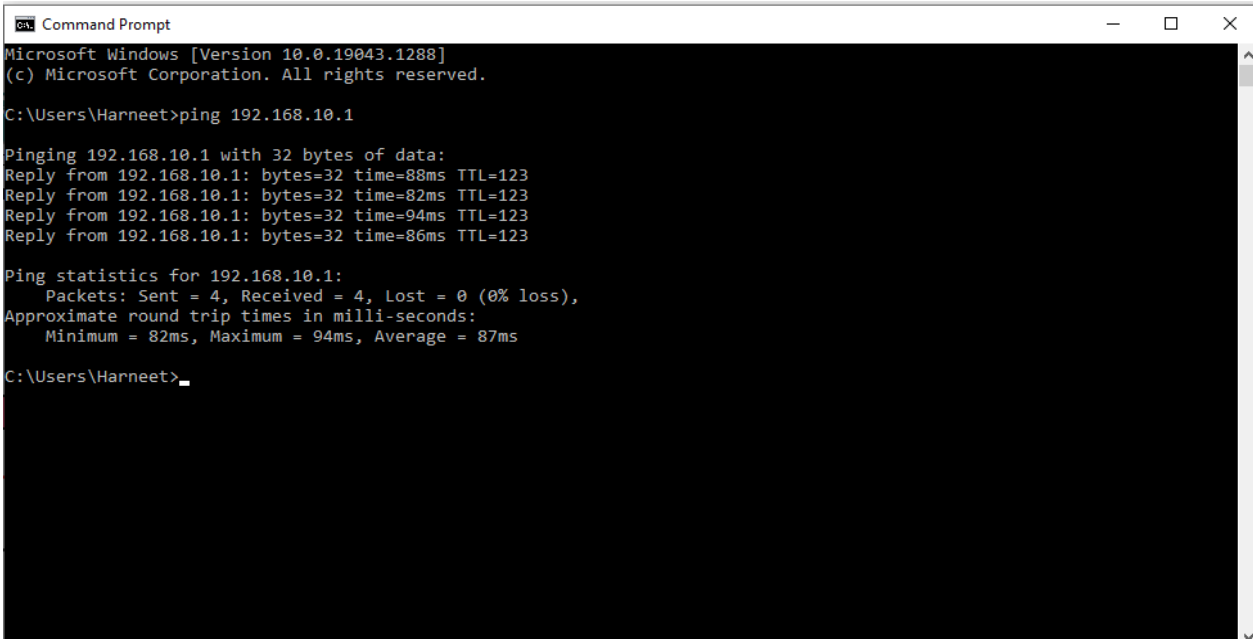


Figure 5.17: Pinging the website using IP address(as assigned) on local PC.

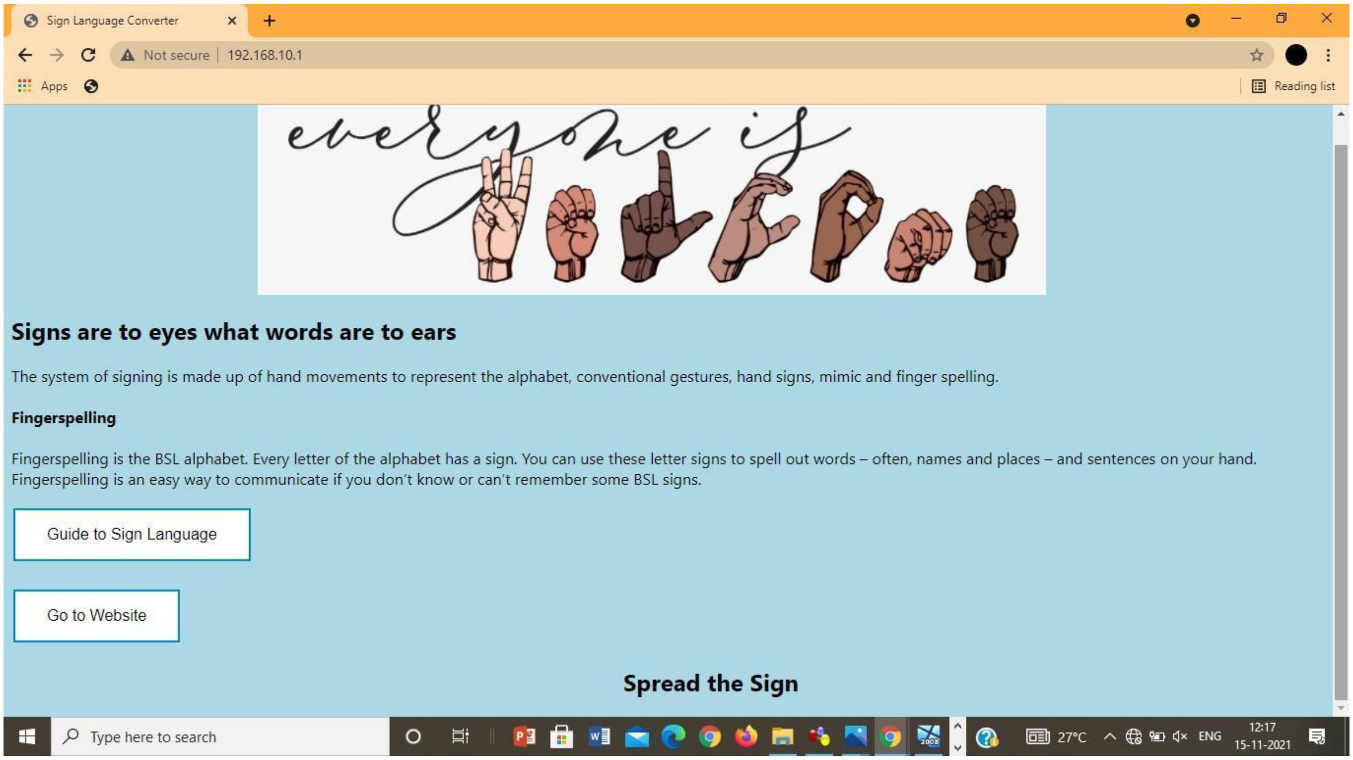


Figure 5.18: Final Representation of User Interface

5.2 Snapshots of system

5.2.1 Sign Language Recognition

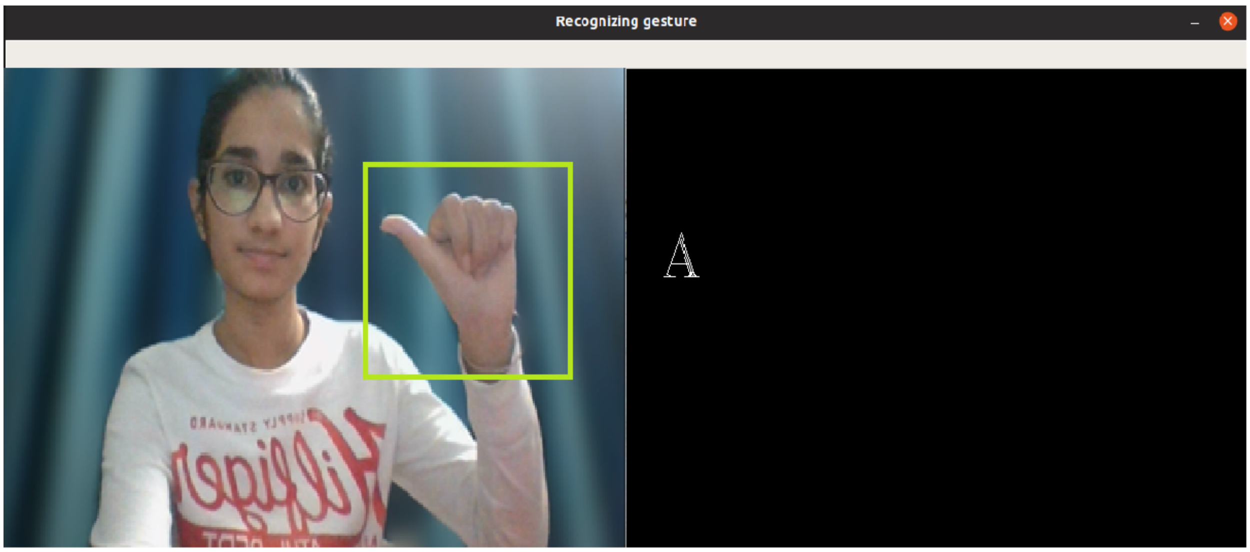


Figure 5.19: Screenshot of the result obtained for letter A

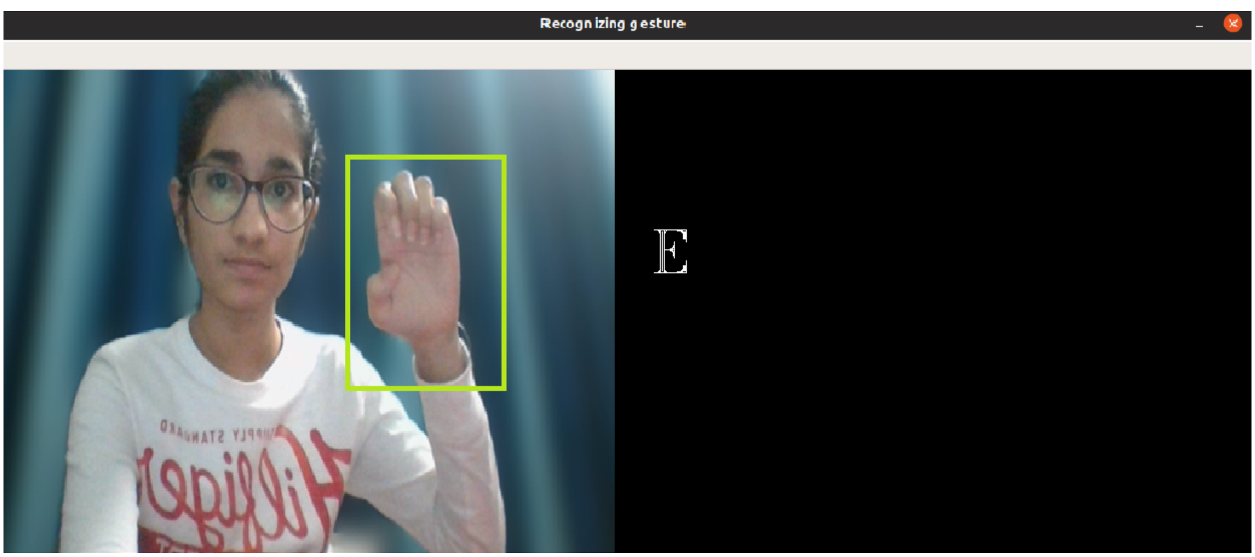


Figure 5.20: Screenshot of the result obtained for letter E

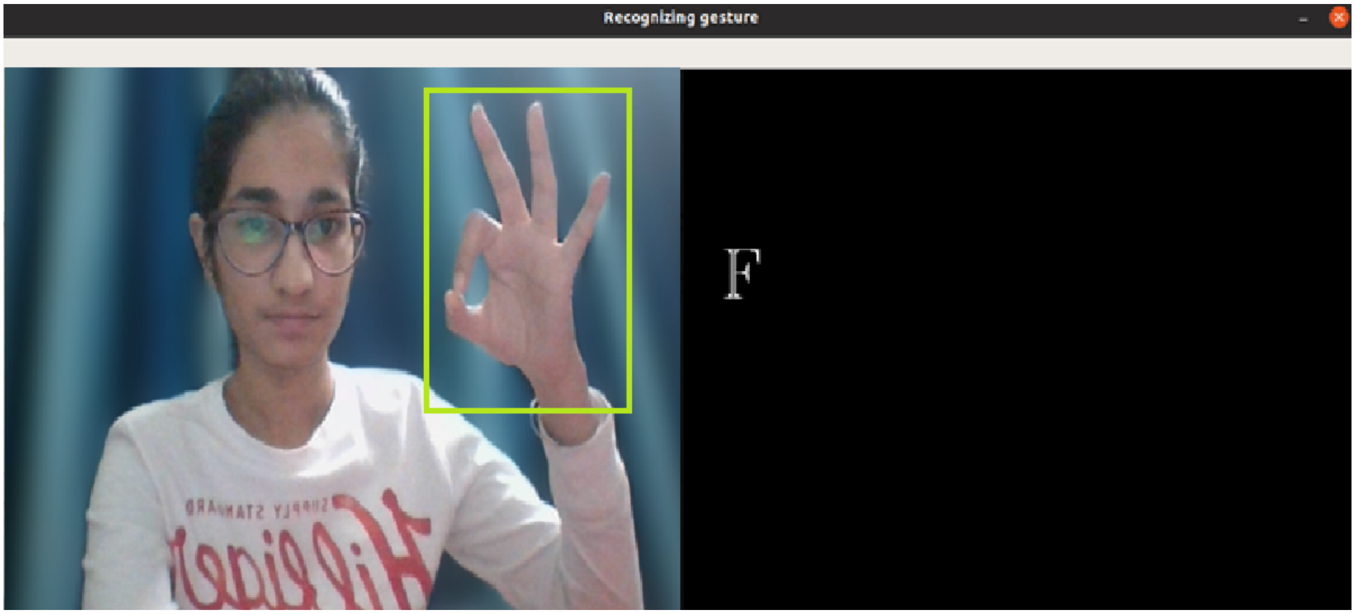


Figure 5.21: Screenshot of the result obtained for letter F

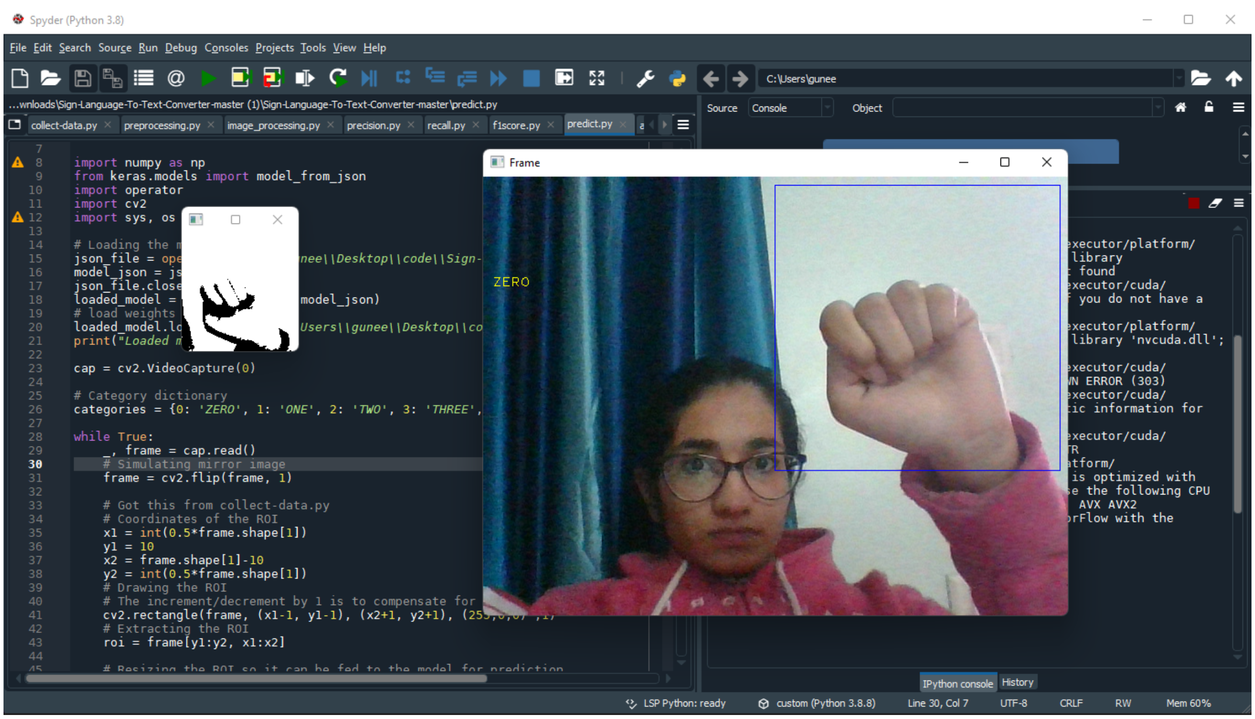


Figure 5.22: Sign Detection

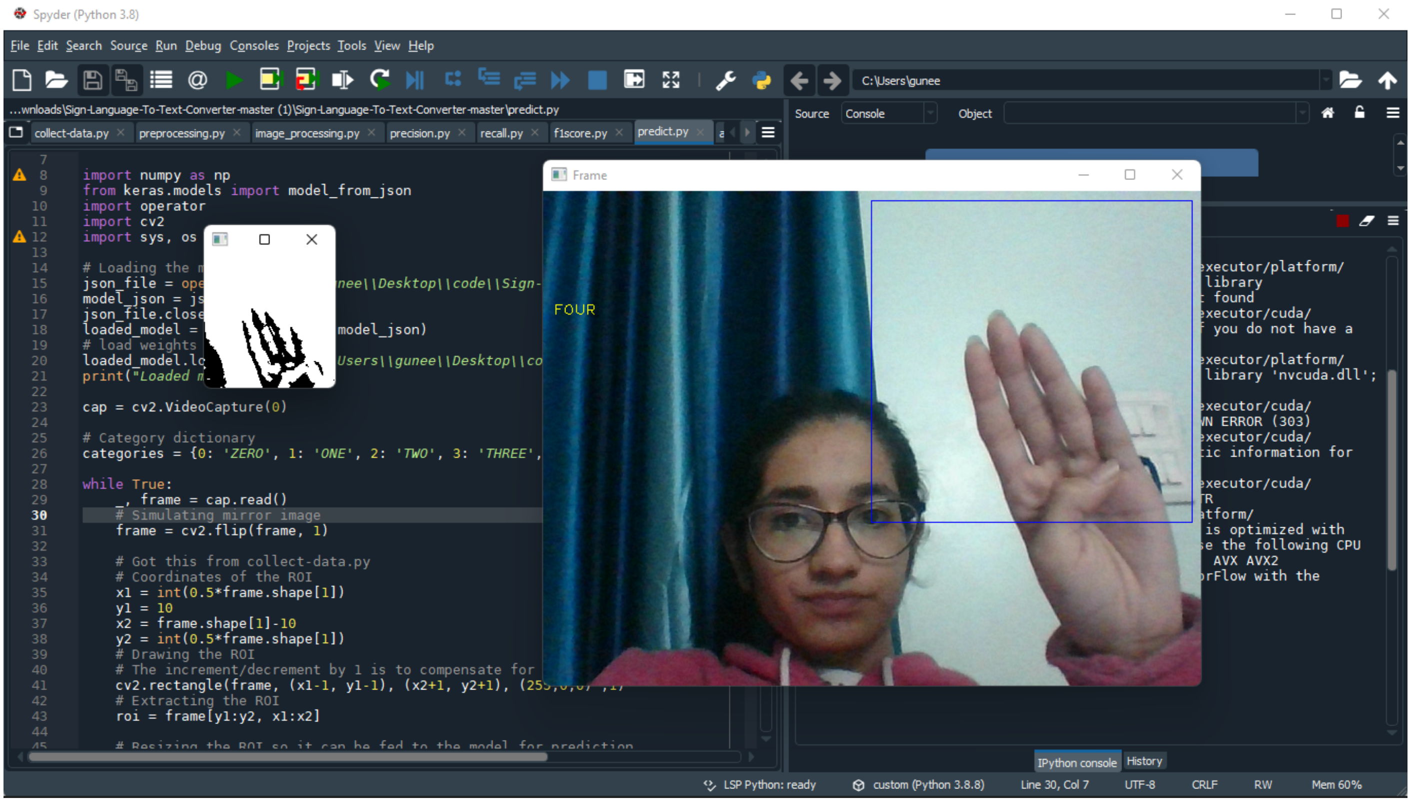


Figure 5.23 Sign Detection

CHAPTER 6:CONCLUSIONS AND FUTURE SCOPE

Conclusions:

This project demonstrates that with an extremely high degree of accuracy, CNN can be used to solve computer vision problems. A translator, which translates the fingerspelling sign language, is obtained.

The project is able to solve a subset of the Sign Language translation problem as the Sign languages are spoken more in context rather than as finger spelling languages . The main objective has been achieved, that is, the need for an interpreter has been eliminated.

Future Scope:

- Nowadays, applications need several kinds of images as sources of information for elucidation and analysis. Several features are to be extracted so as to perform various applications. When an image is transformed from one form to another such as digitizing, scanning, and communicating, storing, etc. degradation occurs.
- Therefore ,the output image has to undertake a process called image enhancement, which consists of a group of methods that seek to develop the visual presence of an image. Image enhancement is fundamentally enlightening the interpretability or awareness of information in images for human listeners and providing better input for other automatic image processing systems.
- Image then undergoes feature extraction using various methods to make the image more readable by the computer.Sign language recognition system is a powerful tool to prepare an expert's knowledge, edge detect and the combination of inaccurate information from different sources. the intent of convolution neural network is to get the appropriate classification

REFERENCES AND BIBLIOGRAPHY

This combined bibliography contains both references consulted in the previous editions of this book and a sampling of books on issues such as design theory, design in different disciplines, product development, project management techniques, optimization theory, applications of artificial intelligence, engineering ethics and the practice of engineering, and more. This is not a complete list of works: The literature on design and project management alone are both vast and rapidly expanding. Thus, keep in mind that this list represents only the tip of a very large iceberg of published work in design and in project management. Some of the works cited are just intellectually interesting, and some are books that students in particular will find useful for project work.

[1] N B, Mahesh K. 2018. "Conversion of Sign Language into Text." *International Journal of Applied Engineering Research ISSN 0973-4562 Volume 13, Number 9*. https://www.ripublication.com/ijaer18/ijaerv13n9_90.pdf

[2]Bantupalli, Kshitij, and Ying Xie. 2019. "American Sign Language Recognition using Deep Learning and Computer Vision." *IEEE*, (January). https://ieeexplore.ieee.org/abstract/document/8622141.

[3] Arsan, Taner, and Oguz Ulgen. 2015. "SIGN LANGUAGE CONVERTER." *International Journal of Computer Science & Engineering Survey (IJCSES) Volume* 6, *Number* 4 (August). https://www.researchgate.net/publication/282839736_Sign_Language_Converter.

[4] Grif, Mikhail G., and Olga O. Korolkova. 2011. "Development of computer sign language translation technology for deaf people." *IEEE*, (September). https://ieeexplore.ieee.org/abstract/document/6021115.

[5] Baranwal, Neha, and G. C. Nandi. 2015, "Continuous dynamic Indian Sign Language gesture recognition with invariant backgrounds by Kumud Tripathi", *Conference on Advances in Computing Communications and Informatics*, (September). https://ieeexplore.ieee.org/abstract/document/7275945

[6] Hu Peng, "Application Research on Face Detection Technology based on Open CV in Mobile Augmented Reality", International Journal of Signal Processing, Image Processing and Pattern Recognition, *Volume* 8, *Number* 2 (2015) http://article.nadiapub.com/IJSIP/vol8_no4/22.pdf