**CONVERSION OF SIGN LANGUAGE (VIDEO) INTO TEXT FOR DEAF AND DUMB PEOPLE.**

**Minor Project II**

Submitted by:

**Arjit Khurkhuriya   (9916103034)**

**Ankita Sharma (9916103082)**

**Ojasvi Saxena (9916103125)**

**Gaurav Singh (9916103186)**

Under the supervision of:

**Mrs. Ambalika Sarkar.**

****

**Department of CSE/IT**

**Jaypee Institute of Information Technology University, Noida**

**MAY 2019**

**ACKNOWLEDGEMENT**

I would like to place on record my deep sense of gratitude to Mrs. Ambalika Sarkar, Assistant Professor, Jaypee Institute of Information Technology, India for her generous guidance, help and useful suggestions.

I express my sincere gratitude to Mr Anurag Srivastava and Dr. Sanjeev Patel, Dept. of Computer Science, India, for his/her stimulating guidance, continuous encouragement and supervision throughout the course of present work.

I also wish to extend my thanks to Mr.Gaurav Singh , Mr. Ojasvi Saxena , Ms. Ankita Sharma and other classmates for their insightful comments and constructive suggestions to improve the quality of this project work.

**Signatures of Students**

Arjit Khurkhuriya (9916103034)

Ankita Sharma (9916103082)

Ojasvi Saxena (9916103125)

Gaurav Singh (9916103186)

***ABSTRACT***

Inability to speak is considered to be true disability.People with this disability  use different modes to communicate with others, there are number of methods available for their communication one such common method of communication is  sign language.

Developing sign language application for deaf people can be very important,as they’ll be able to communicate easily with even those who don’t understand sign language. Our project aims at taking the basic step in bridging the communication gap between normal people, deaf and dumb people using sign language.

The main focus of this work is to create a vision based system to identify sign language gestures from the video sequences.There as on for choosing a system based on vision relates to the fact that it provides a simpler and more intuitive way of communication between a human and a computer. In this report,28 different gestures have been considered.

**TABLE OF CONTENTS**

*Abstract*

*Acknowledgement*

*List of Tables*

*List of Figures*

*List of Abbreviations*

*List of Nomenclature*

**Chapter-1     INTRODUCTION. 1**

**Chapter-2    REQUIREMENT ANALYSIS. 2-3**

2.1 Python 3.x

                          2.2 Tensorflow 1.5

                           2.3 Matplotlib

                           2.4 OpenCV 3.4

                           2.5 Numpy

**Chapter-3     IMPLEMENTATION. 4-6**

3.1 Convolution

   3.2 Pooling

3.3 Subsampling

                           3.4 Activation

3.5 Fully Connected

**Chapter-4 EXPERIMENT RESULT AND ANALYSIS 7-10**

4.1 Data Set Used.

4.2 Video to Frame

4.3 Result

**Chapter-5 CONCLUSION AND FUTURE SCOPE 11**

**Chapter-6 REFERENCES 12**

**Chapter-7 POWER POINT PRESENTATION 13 - 28**

**LIST OF TABLES AND FIGURES**

FIG 1 DATA SET OF IMAGES.

FIG 2 VIDEO TO IMAGE.

FIG 3 DATA SET OF VIDEOS.

FIG 4 RESULT 1

FIG 5 RESULT 2

FIG 6 GANTT CHART

**1.INTRODUCTION**

Motion of any body part like face, hand is a form of gesture. Here for gesture recognition we are using image processing and computer vision. Gesture recognition enables computer to understand human actions and also acts as an interpreter between computer and human. This could provide potential to human to interact naturally with the computers without any physical contact of the mechanical devices. Gestures are performed by deaf and dumb community to perform sign language. This community used sign language for their communication when broadcasting audio is impossible, or typing and writing is difficult, but there is the vision possibility. At that time sign language is the only way for exchanging information between people. Normally sign language is used by everyone when they do not want to speak, but this is the only way of communication for deaf and dumb community. Sign language is also serving the same meaning as spoken language does. This is used by deaf and dumb community all over the world but in their regional form like ISL, ASL. Sign language can be performed by using Hand gesture either by one hand or two hands. It is of two type Isolated sign language and continuous sign language.

Idea behind this project is to make a machine learning project which can

help Deaf people to understand what the other person is saying with the help of a computer based

software . For this we use machine learning algorithms like Recursive Neural Networks to train our

data set . We use 28 different gestures to recognise alphabets of the English Language. This will help

us to recognise all the alphabet using hand movement in front of a web cam. The data set used consists of Indian Sign Language(ISA) Gestures, with around 3000 images belonging to 28 gestures categories.

1.1   Sign Language

Deaf people around the world communicate using sign language as distinct  from spoken language in their every day a visual language that uses a system of manual, facial and body movements as the means of communication. Sign language is not an universal language,and different sign languages are used in different countries, like the many spoken languages all over the world. Some countries such as Belgium, the UK, the USAorIndia may have more than one sign language. Hundreds of sign languages are in used around the world, for instance, Japanese Sign Language,British Sign Language(BSL),Spanish Sign Language,  Turkish Sign Language.

**REQUIREMENTS ANALYSIS**

***1.TENSORFLOW***

***2. MATPLOTLIB***

***3.OPEN CV***

***4.NUMPY***

***5.KERAS***

***TENSOR FLOW***

To understand tensors well, it’s good to have some working knowledge of linear algebra and vector calculus. You already read in the introduction that tensors are implemented in TensorFlow as multidimensional data arrays, but some more introduction is maybe needed in order to completely grasp tensors and their use in machine learning

***MATPLOTLIB***

         Matplotlib is a library for making 2D plots of arrays in Python. Although it has its origins in emulating the MATLAB graphics commands, it is independent of MATLAB, and can be used in a Pythonic, object oriented way. Although Matplotlib is written primarily in pure Python, it makes heavy use of NumPy and other extension code to provide good performance even for large arrays

***OPEN CV***

OpenCV (Open Source Computer Vision Library) is an open-source BSD- licensed library that includes several hundreds of computer vision algorithms. The document describes the so-called OpenCV 2.x API, which is essentially a C++ API, as opposite to the C-based OpenCV 1.x API. OpenCV has a modular structure, which means that the package includes several shared or static libraries.

***NUMPY***

NumPy is module for Python. The name is an acronym for "Numeric Python" or "Numerical Python". It is pronounced /ˈnʌmpaɪ/ (NUMpy) or less often / ˈnʌmpi (NUM-pee)). It is an extension module for Python, mostly written in C. This makes sure that the precompiled mathematical and numerical functions and functionalities of Numpy guarantee great execution speed.

***.KERAS***

**Keras** is an open-source neural network library written in Python. It is capable of running on top of TensorFlow, Microsoft Cognitive Toolkit, Theano, or PlaidML. Designed to enable fast experimentation with deep neural networks, it focuses on being user-friendly, modular, and extensible. It was developed as part of the research effort of project ONEIROS (Open-ended Neuro-Electronic Intelligent Robot Operating System), and its primary author and maintainer is François Chollet, a Google engineer. Chollet also is the author of the XCeption deep neural network model.

**IMPLEMENTATION**

***CONVOLUTIONAL NEURAL NETWORK***

  Convolutional   Neural Network (CNN) Neural networks, as its name suggests, is a machine learning technique which is modeled after the brain structure. It comprises of a network of learning units called neurons. These neurons learn how to convert input signals (e.g. picture of a cat) into corresponding output signals (e.g. the label “cat”), forming the basis  of automated recognition. A convolutional neural network (CNN, or ConvNet) is a type of feed­forward artificial neural network in which the connectivity pattern between its neurons is inspired by the organization of the animal visual cortex. CNNs have repetitive blocks of neurons that are applied across space (for images) or time (for audio signals etc). For images, these blocks of neurons can be interpreted as 2D convolutional kernels, repeatedly applied over each patch of the image. For speech, they can be seen as the 1D convolutional kernels applied across time ­windows. At training time, the weights for these repeated blocks are 'shared', i.e. the weight gradients learned over various image  patches are averaged.

There are four steps in CNN..

1.Convolution

2. Subsampling

3. Activation

4. Full  Connectedness

**1.** **Convolution**

The first layers that receive an input signal are called convolution filters. Convolution is a process where the network tries to label the input signal by referring to what it has learned in the past. If the input signal looks like previous cat images it has seen before, the “cat” reference signal will be mixed into, or convolved with, the input signal. The resulting output signal is then passed  on to the next layer.

Convolution has the nice property of being translational invariant . Intuitively, this means that each convolution filter represents a feature of interest (e.g whiskers, fur), and the CNN algorithm learns which features comprise the resulting reference. The output signal strength is not dependent on where the features are located, but simply whether the features are present. Hence, a cat could be sitting in different positions, and the CNN algorithm  would still be able to recognize it.

**2**. **Subsampling**

Inputs from the convolution layer can be “smoothened” to reduce the sensitivity of the filters to noise and variations. This smoothing process is called subsampling , and can be achieved by taking averages or taking the maximum over a sample of the signal. Examples of subsampling methods (for image signals) include reducing the size of the image, or reducing the color contrast  across red, green, blue (RGB) channels

**3**. **Pooling**

A  pooling  layer is  another building  block of a CNN

Its function is to progressively reduce the spatial size of the representation to reduce the amount of parameters and computation in the network. Pooling layer  operates on each feature map independently. The most common approach used in pooling is max pooling in which maximum of a region taken as its representative. For example in the following diagram  a 2x2 region is replaced by the maximum value in it.

4. **Activation**

The activation layer controls how the signal flows from one layer to the next, emulating how neurons are fired in our brain. Output signals which are strongly associated with past references would activate more neurons, enabling  signals to be propagated more efficiently for identification. CNN is compatible with a wide variety of complex activation functions to model signal propagation, the most common function being the Rectified Linear Unit (ReLU),  which is favored for its faster training speed.

                                                5. **Fully  Connected**

The last layers in the network are fully connected, meaning that neurons of preceding layers are connected to every neuron in subsequent layers. This mimics high level reasoning where all possible pathways from the input to output  are considered.

**6. Implementation**

Algorithms used in training CNN are analogous to studying for exams using flash cards. First, you draw several flashcards and check if you have mastered the concepts on each card. For cards with concepts that you already know,

discard them. For those cards with concepts that you are unsure of, put them back into the pile. Repeat this process until you are fairly certain that you know enough concepts to do well in the exam. This method allows you to focus on less familiar concepts by revisiting them often. Formally, these algorithms are called gradient descent algorithms for forward pass learning. Modern deep learning algorithm uses a variation called stochastic gradient descent, where instead of drawing the flashcards sequentially, you draw them at random. If similar topics are drawn in sequence, the learners might over­estimate how well they know the topic. The random approach helps to minimize  any form of bias in the learning of topics

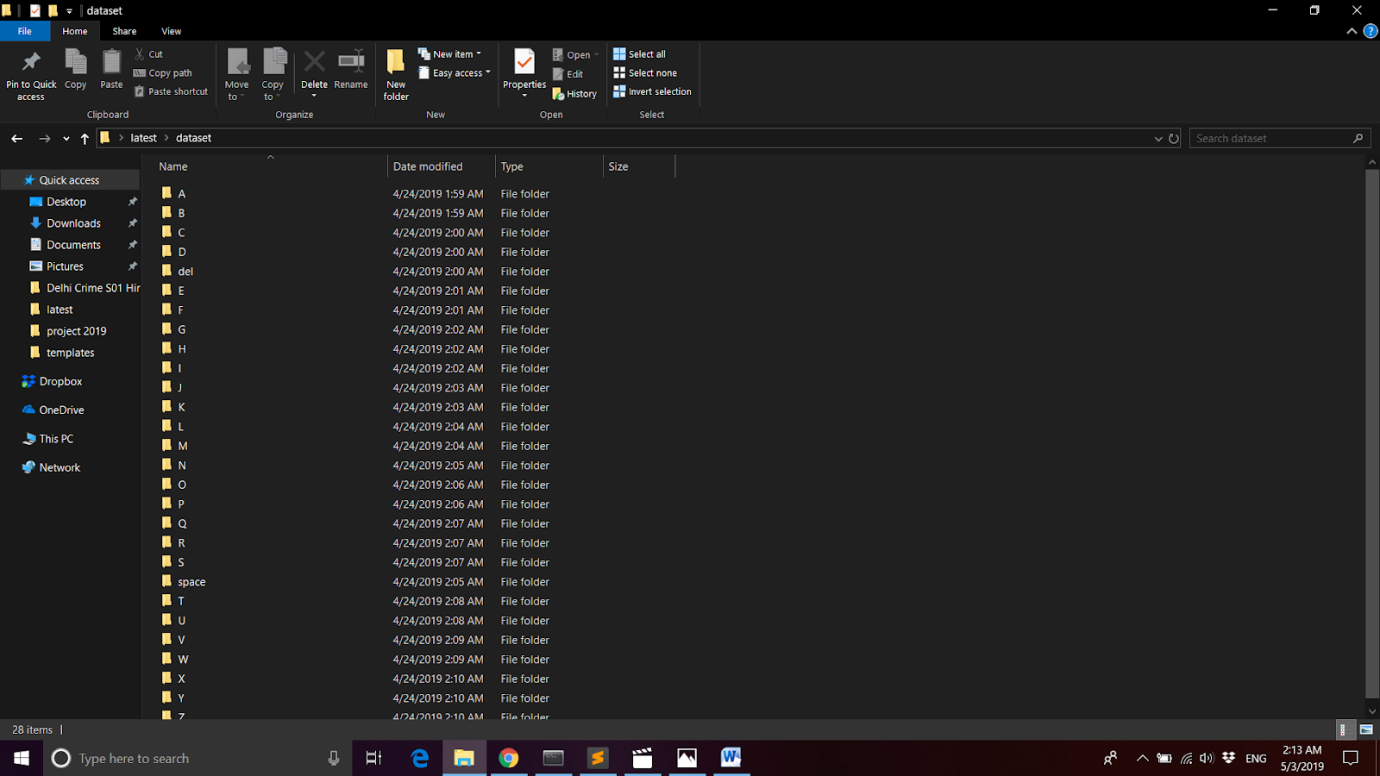
Learning algorithms require feedback. This is done using a validation set where the CNN would make predictions and compare them with the true labels or ground truth.

**EXPERIMENT RESULTS AND ANALYSIS**

After performing the experiment many number of times , the following result has been showing

***DATASET USED***

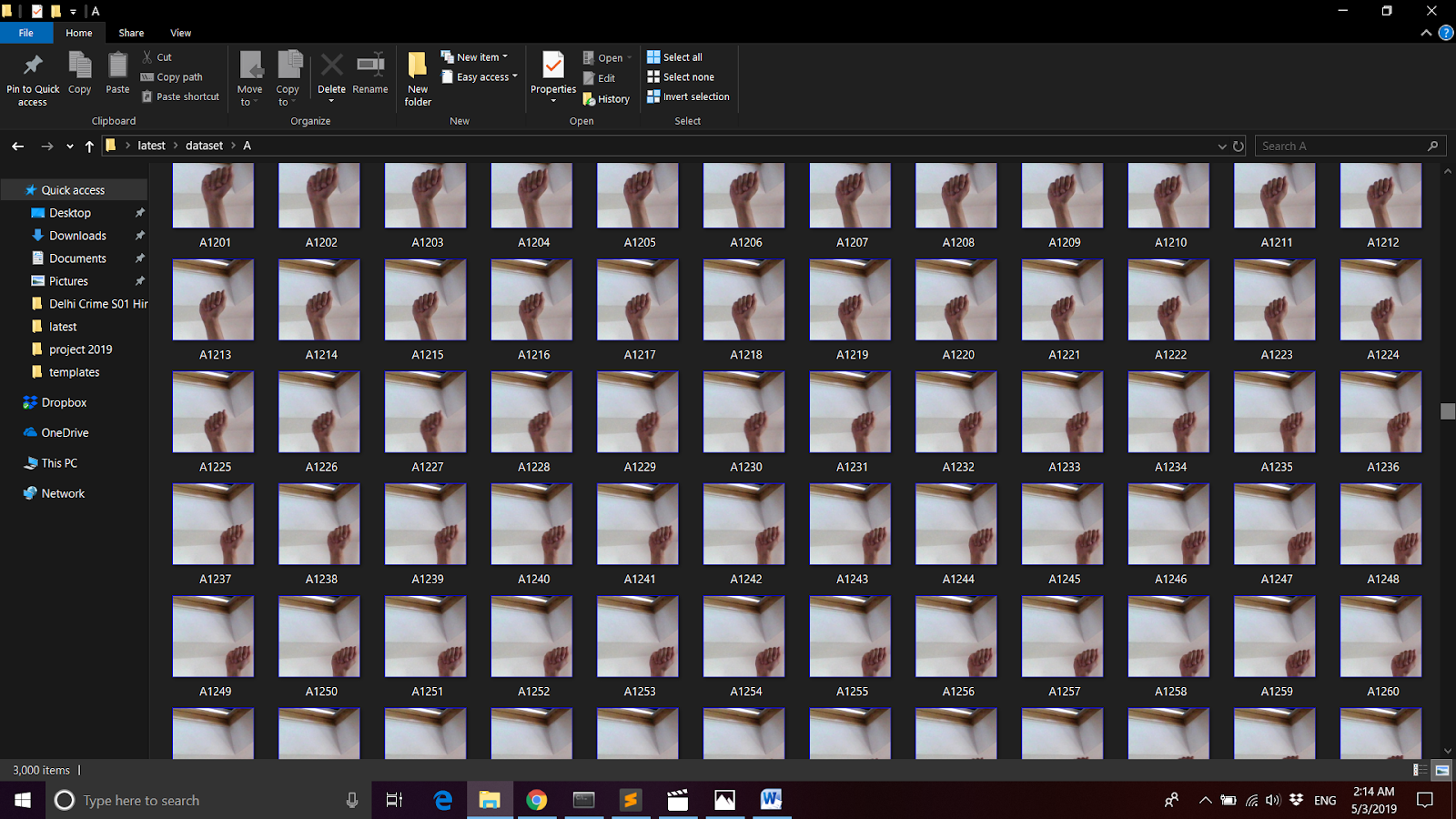
In the dataset various videos with a similar background has been used for the training of dataset by breaking the video into the frames. The following sample is used,

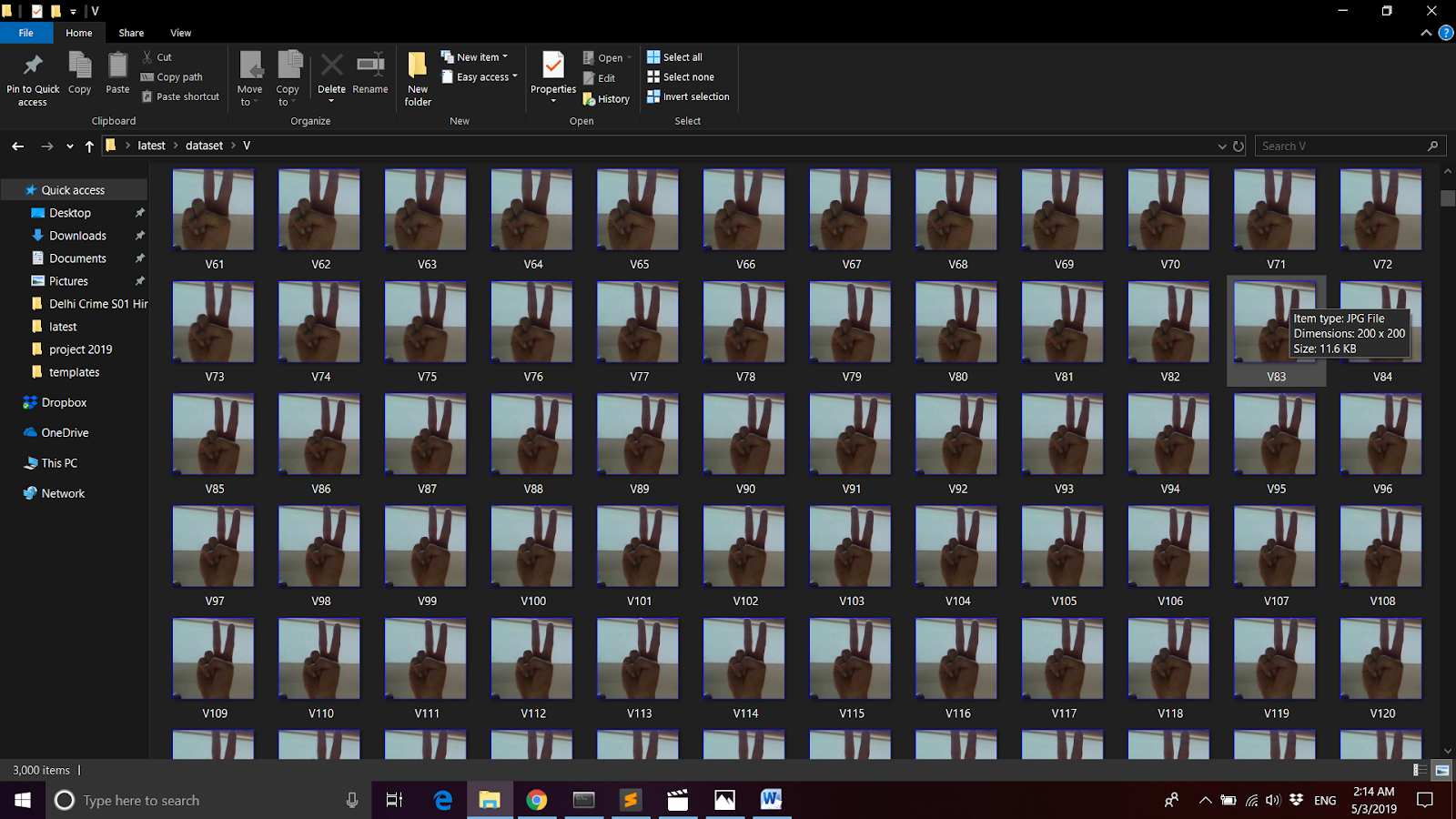


***2. VIDEO TO FRAME***

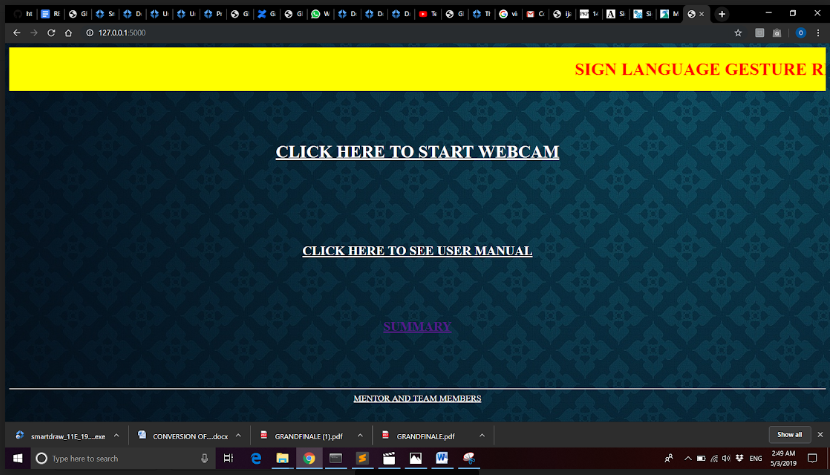
In the second step , for recognition of our gesture the input

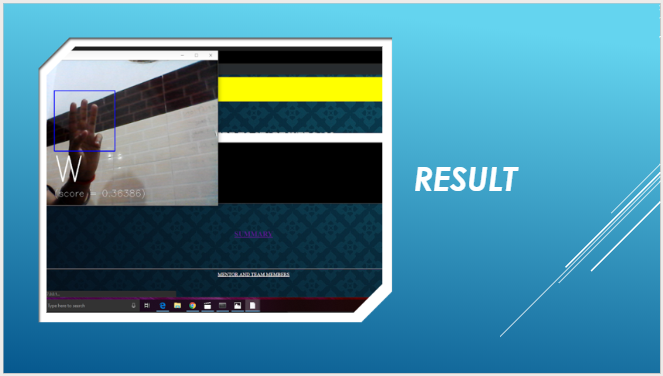
video is break into several images.



  
**RESULT-**

For every instance of the input the result is showing in the following way.





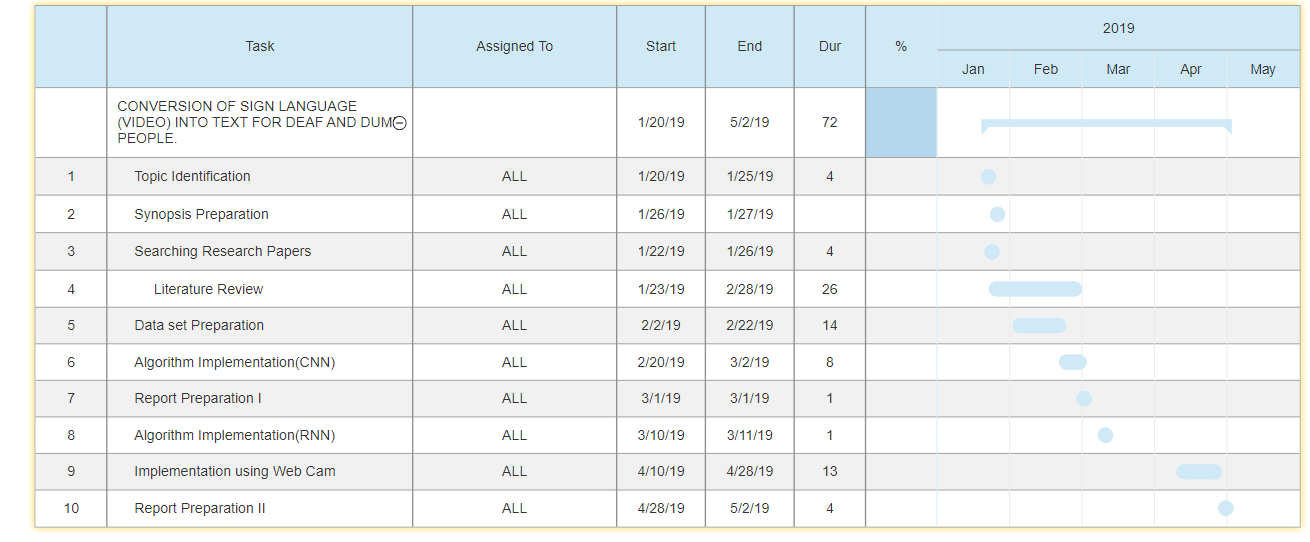
**CONCLUSION AND FUTURE SCOPE**

We wish to extend our work further in recognising continuous sign language gestures with better accuracy.

Hand gestures are a powerful way for human communication, with lots of potential applications in the area of human computer interaction. Vision ­based hand gesture recognition techniques have many proven advantages compared with traditional devices. However, hand gesture recognition is a difficult problem and the current work is only a small contribution towards achieving the results needed in the field of sign language gesture recognition. This report presented a vision ­based system able to interpret isolated hand gestures from the  Argentinian Sign Language(LSA).

Our proposed work plan will be:  Currently, we have worked on small dataset in future we are planning on increasing our dataset by increasing images i.e, we will try to improve accuracy by training more and more data and taking flexible string length as per requirement by user .In future we are trying to implement  model that convert video into text and further

Convert it  into audio which makes conversation more flexible between deaf and dumb people and others.



**REFERENCES**

[1] Mahesh Kumar N B1 “SIGN LANGUAGE CONVERTER”  International Journal of Applied Engineering Research ISSN 0973-4562 Volume 13, Number 9 (2018) pp. 7154-7161

[2]   M.Suresh Anand, A.Kumaresan“An Integrated Two Way ISL (Indian Sign Language) Translation System – A New Approach”International Journal of Computer ,2017.

[3]   Mayuresh Keni, Shireen Meher, Aniket Marathe“Sign Language Recognition System” International Journal of Scientific & Engineering Research, Volume 4, Issue 12, December-2013

[4]<http://www.academia.edu/Documents/in/Sign_Language_Translation_and_Interpreting_Studies>

[5]<http://airccse.org/journal/ijcses/papers/6415ijcses03.pdf>