A Project Report

on

SIGN LANGUAGE INTERPRETER

Submitted in partial fulfillment of the

requirement for the award of the degree of

Bachelor of Technology in Computer Science and Engineering

GALGOTIAS UNIVERSITY

(Established under Galgotias University Uttar Pradesh Act No. 14 of 2011)

Under The Supervision of
Mr. Ravinder Ahuja
Assistant Professor
Department of Computer Science and Engineering

Submitted By

18SCSE1010074 - ANIL KUMAR SHARMA

18SCSE1010604 - PRADEEP PARAJULI

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

GALGOTIAS UNIVERSITY, GREATER NOIDA, INDIA

DECEMBER - 2021

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING
GALGOTIAS UNIVERSITY, GREATER NOIDA

CANDIDATE'S DECLARATION

I/We hereby certify that the work which is being presented in the project, entitled **" SIGN LANGUAGE INTERPRETER "** in partial fulfillment of the requirements for the award of the **BACHELOR OF TECHNOLOGY IN COMPUTER SCIENCE AND ENGINEERING** submitted in the **School of Computing Science and Engineering** of Galgotias University, Greater Noida, is an original work carried out during the period of **JULY-2021 to DECEMBER-2021**, under the supervision of **Mr. Ravinder Ahuja, Assistant Professor**, **Department of Computer Science and Engineering** of School of Computing Science and Engineering , Galgotias University, Greater Noida

The matter presented in the project has not been submitted by me/us for the award of any other degree of this or any other places.

18SCSE1010074 – ANIL KUMAR SHARMA

18SCSE1010604 – PRADEEP PARAJULI

This is to certify that the above statement made by the candidates is correct to the best of my knowledge.

Supervisor

Mr.Ravinder Ahuja

Assistant Professor

CERTIFICATE

The Final Thesis/Project/ Dissertation Viva-Voce examination of **18SCSE1010074 - ANIL KUMAR SHARMA, 18SCSE1010604- PRADEEP PARAJULI** has been held on __and his/her work is recommended for the award of **BACHELOR OF TECHNOLOGY IN COMPUTER SCIENCE AND ENGINEERING**.

Signature of Examiner(s) **Signature of Supervisor(s)**

Signature of Project Coordinator **Signature of Dean**

Date:

Place:

ACKNOWLEDGEMENT

Primarily We might thank God for having the ability to finish this mission with success .Then we would love to thank my mission manual Mr. Ravinder Ahuja ,whose treasured steerage has been those that helped me patch this mission and make it complete evidence success, his pointers and his commands has served because the foremost contributor in the direction of the finishing touch of the mission.

Then We would love to thank our mother and father and friends who've helped me with their treasured pointers and steerage has been beneficial in diverse levels of the finishing touch of the mission.

ABSTRACT

Sign Language is a medium for communication for many disabled people who are dumb or deaf, to communicate both people need to know sign language or need a translator, but everywhere they are not getting a translator. This project is a prototype that will work as a translator for them, so that they can communicate with anyone irrespective of knowledge of sign language. With the help of gestures, we can perform many actions like human-computer interaction, gesture-controlled home appliances, and many applications that use gestures as the trigger input. In this prototype, an image is obtained using a webcam. OpenCV helps in detecting gestures from images. And on a later stage gesture is converted to text. OpenCV is a python package tool that provides support with image processing technique.

The main focus of this work is to create a vision based system to identify Finger spelled letters of ASL. The reason 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, 36 different categories have been considered: 26 categories for English Alphabets (a-z) and 10 categories for Numerals .

Table of Contents

List of Figures and Tables

Acronyms

RNN	Recurrent Neural Networks
ML	Machine Learning
DL	Deep Learning
CNN	Convolution Neural Networks

CHAPTER-1

Introduction

1.1 Introduction of Project:

Machine Learning(ML) and Artificial Intelligence(AI) is now at the heart of innovation economy and thus the base for this project is also the same. In the recent past a field of AI namely Deep Learning has turned a lot of heads due to its impressive results in terms of accuracy when compared to the already existing Machine learning algorithms. The task of being able to generate a meaningful sentence from an image is a difficult task but can have great impact, for instance helping the deaf and dumb to have a better communication. The task of sign language interpretation is significantly harder than that of image classification, which has been the main focus in the computer vision community. A description for an image must capture the relationship between the sign in the image.

1.1.1 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 USA or India 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.

American Sign Language is a predominant sign language Since the only disability D&M people have is communication related and they cannot use spoken languages hence the only way for them to communicate is through sign language. Communication is the process of exchange of thoughts and messages in various ways such as speech, signals, behavior and visuals. Deaf and dumb(D&M) people make use of their hands to express different gestures to express their ideas with other people. Gestures are the nonverbally exchanged messages and these gestures are understood with vision. This nonverbal communication of deaf and dumb people is called sign language.

Fingerspelling	Word level sign vocabulary	Non-manual features
Used to spell words letter by letter .	Used for the majority of communication.	Facial expressions and tongue, mouth and body position.

Fig. 1: Sign language is a visual language and consists of 3 major components.

In our project we basically focus on producing a model which can recognize Finger spelling based hand gestures in order to form a complete word by combining each gesture. The gestures we aim to train are as given in the image below.

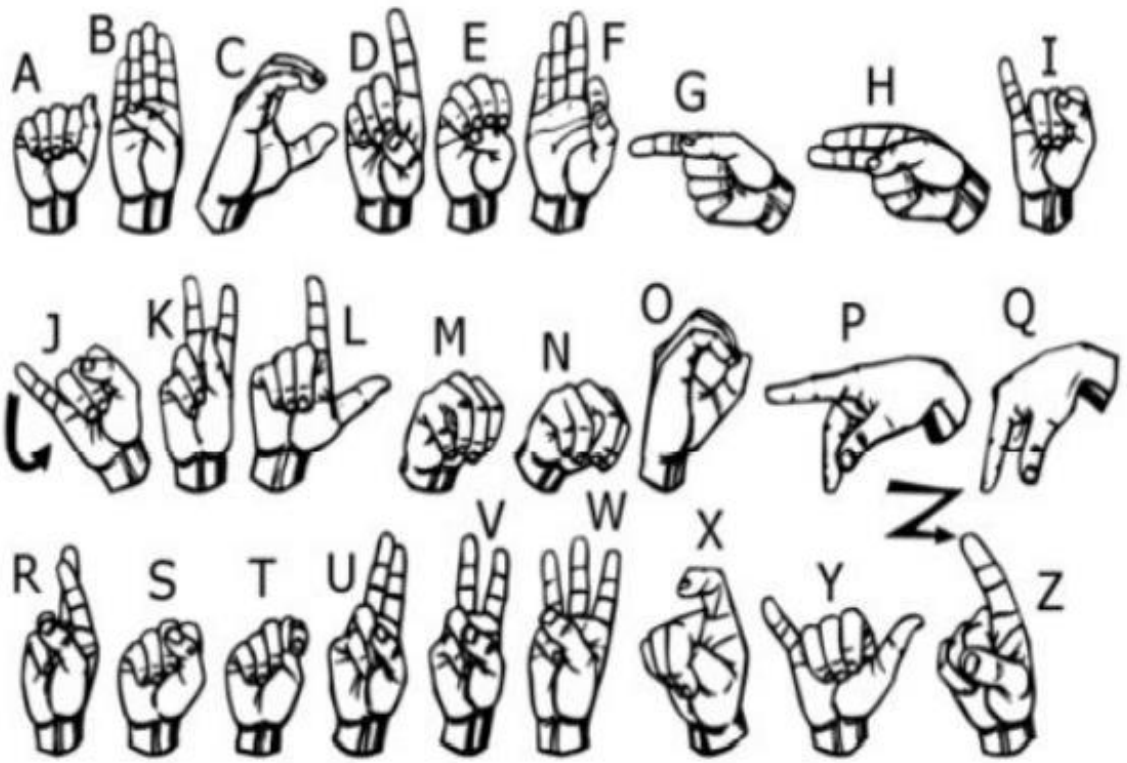


Fig. 2: American Sign Language.

There is no universal language for hearing individuals. Different nations have their own gesture-based communication, despite the fact that there are a few striking similitude among them. It is yet muddled the number of gesture-based communications that exists on the planet. A few dialects have lawful acknowledgment and some poor people. India's National Association of Deaf gauges that there are 18 million individuals in India with hearing weaknesses. This paper examines the execution of a framework that interprets Indian or American Sign Language motions to its English language understanding.

1.2 Motivation

For interaction between normal people and D&M people a language barrier is created as sign language structure which is different from normal text. So they depend on vision based communication for interaction.

If there is a common interface that converts the sign language to text the gestures can be easily understood by the other people. So research has been made for a vision based interface system where D&M people can enjoy communication without really knowing each other's language.

The aim is to develop a user friendly human computer interfaces (HCI) where the computer understands the human sign language. There are various sign languages all over the world, namely American Sign Language (ASL), French Sign Language, British Sign Language (BSL), Indian Sign language, Japanese Sign Language and work has been done on other languages all around the world.

1.3 Problem Definition:

In day to day life we have seen lots of images on internet and almost everywhere like news, articles. Sometimes images having some short amount of description about it but out of them some images are just images and nothing extra as we are human we can figure out what's in it. So we are trying to build a model that will take input image from user and then machine gives a suitable caption. So due to this our model can analyze thousands of images and then it will be used for test purposes to know what images says. It is very helpful in Artificial Intelligence to recognize images and gives responses to request sends comes from users.

According to survey over 5% of world of have hearing loss which is around 430 million. Almost all of them have difficulties in communication.

1. 98% of them have did not have education of sign language.
2. 70% of families do not use official sign language to communicate with their kids.
3. 68% of working age deaf people are unemployed.
4. 1 out of 4 deaf left their job due to discrimination they face in their job.

1.4 Tool and Technology Used

In the report we first consider the task of image classification separately. We try to classify the images of the dataset using various classifiers. We try to apply some linear classifiers. The accuracy with these models was much less than expected since a high loss factor at the time of classification will amplify the loss even further at the time of caption generation. We then try to train a simple Convolutional Neural Network and achieve decent results within few hours of training. Thus, by the end of this section we conclude that CNN are a good fit to be used as the image encoder for the model.

1.4.1 Python:

Python is an interpreted, high-level, general-purpose programming language. Created by Guido van Rossum and first released in 1991, Python's design philosophy emphasizes code readability with its notable use of significant whitespace. Its language constructs and object-oriented approach aim to help programmers write clear, logical code for small and large-scale projects.

1.4.2 Jupyter Notebook:

The Jupyter notebook is an open source web application that allows you to create and share documents that contain live code, equations, visualizations and narrative text. Uses include: data cleaning and transformation, numerical simulation, statistical modeling, data visualization, machine learning and much more.

CHAPTER-2

2.1 Literature Survey

Real time vision-based system for hand gesture recognition for human computer interaction in many applications. The system can recognize 35 different hand gestures given by Indian and American Sign Language or ISL and ASL at faster rate with virtuous accuracy. RGB-to-GRAY segmentation technique was used to minimize the chances of false detection. Authors proposed a method of improvised Scale Invariant Feature Transform (SIFT) and same was used to extract features. The system is model using MATLAB. To design and efficient user-friendly hand gesture recognition system, a GUI model has been implemented.

In the recent years there has been tremendous research done on the hand gesture recognition. With the help of literature survey done we realized the basic steps in hand gesture recognition are:-

- Data acquisition
- Data preprocessing
- Feature extraction
- Gesture classification

2.2 Data acquisition:

The different approaches to acquire data about the hand gesture can be done in the following ways:

2.2.1 Use of sensory devices

It uses electromechanical devices to provide exact hand configuration, and position. Different glove based approaches can be used to extract information .But it is expensive and not user friendly.

2.2.2 Vision based approach

In vision based methods computer camera is the input device for observing the information of hands or fingers. The Vision Based methods require only a camera, thus realizing a natural interaction between humans and computers without the use of any extra devices. These systems tend to complement biological vision by describing 7 artificial vision systems that are implemented in software and/or hardware. The main challenge of vision-based hand detection is to cope with the large variability of human hand's appearance due to a huge number of hand movements, to different skin-color possibilities as well as to the variations in view points, scales, and speed of the camera capturing the scene.

The vision based hand gesture recognition system is shown in fig.:

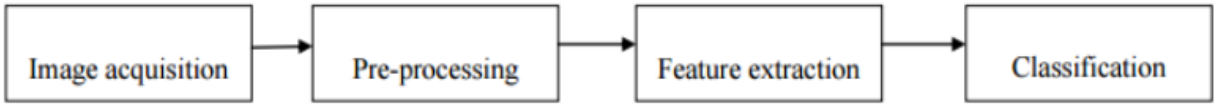


Fig. 3: Block Diagram of vision based recognition system

Vision based analysis, is based on the way human beings perceive information about their surroundings, yet it is probably the most difficult to implement in a satisfactory way. Several different approaches have been tested so far.

1. One is to build a three-dimensional model of the human hand. The model is matched to images of the hand by one or more cameras, and parameters corresponding to palm orientation and joint angles are estimated. These parameters are then used to perform gesture classification.

2. Second one to capture the image using a camera then extract some feature and those features are used as input in a classification algorithm for classification.

2.3 Data preprocessing and Feature extraction for vision based approach

- In [1] the approach for hand detection combines threshold-based color detection with background subtraction. We can use Adaboost face detector to differentiate between faces and hands as both involve similar skin-color.
- We can also extract necessary image which is to be trained by applying a filter called Gaussian blur. The filter can be easily applied using open computer vision also known as OpenCV and is described in [3].
- For extracting necessary image which is to be trained we can use instrumented gloves as mentioned in [4]. This helps reduce computation time for preprocessing and can give us more concise and accurate data compared to applying filters on data received from video extraction.
- We tried doing the hand segmentation of an image using color segmentation techniques but as mentioned in the research paper skin color and tone is highly dependent on the lighting conditions due to which output we got for the segmentation we tried to do were no so great. Moreover we have a huge number of symbols to be trained for our project many of which look similar to each other like the gesture for symbol ‘V’ and digit ‘2’, hence we decided that in order to produce better accuracies for our large number of symbols, rather than 8 segmenting the hand out of a random background we keep background of hand a stable single color so that we don’t need to segment it on the basis of skin color. This would help us to get better results.

2.4 Gesture classification :

- In [1] Hidden Markov Models (HMM) is used for the classification of the gestures .This model deals with dynamic aspects of gestures. Gestures are extracted from a sequence of video images by tracking the skin-color blobs corresponding to the

hand into a body– face space centered on the face of the user. The goal is to recognize two classes of gestures: deictic and symbolic. The image is filtered using a fast look–up indexing table. After filtering, skin color pixels are gathered into blobs. Blobs are statistical objects based on the location (x,y) and the colourimetry (Y,U,V) of the skin color pixels in order to determine homogeneous areas.

- In [2] Naïve Bayes Classifier is used which is an effective and fast method for static hand gesture recognition. It is based on classifying the different gestures according to geometric based invariants which are obtained from image data after segmentation. Thus, unlike many other recognition methods, this method is not dependent on skin color. The gestures are extracted from each frame of the video, with a static background. The first step is to segment and label the objects of interest and to extract geometric invariants from them. Next step is the classification of gestures by using a K nearest neighbor algorithm aided with distance weighting algorithm (KNNDW) to provide suitable data for a locally weighted Naïve Bayes‟ classifier.
- According to paper on "Human Hand Gesture Recognition Using a Convolution Neural Network" by Hsien-I Lin , Ming-Hsiang Hsu, and Wei-Kai Chen graduates of Institute of Automation Technology National Taipei University of Technology Taipei, Taiwan, they construct a skin model to extract the hand out of an image and then 9 apply binary threshold to the whole image. After obtaining the threshold image they calibrate it about the principal axis in order to center the image about it. They input this image to a convolutional neural network model in order to train and predict the outputs. They have trained their model over 7 hand gestures and using their model they produce an accuracy of around 95% for those 7gesture.

CHAPTER 3

3.1 Functionality and Concept of Project

3.1.1 System Requirements

The system requires a few important hardware and software to run this project.

3.1.1.1 Hardware: HD camera, minimum 1GB RAM, 200MB ROM, 1GHz processor. This project is supported by all-new generation laptops. By connecting a USB camera it can be also run on Desktop PC. The camera captures the images at speed of 25 frames per second.

3.1.1.2 Software: Python 3.6 and above, pip packages. The system architecture of this project shows the flow of control.

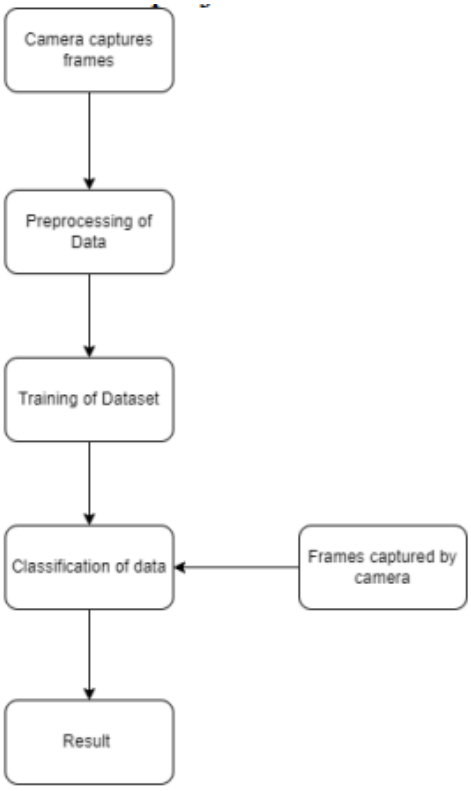


Fig. 4: Sign language interpreter flowchart.

3.2 Image Processing

3.2.1 Introduction to Image Processing

Image processing is a way to transform a photo into a digital form and perform other operations on it in order to produce an improved image or gain useful information from it. The input is an image, such as a video frame and the output can also be an object or picture. This is a type of signal propagation. The system of image processing contains different tools such as image acquisition, image enhancement, image restoration, color image processing, wavelet and multiresolution processing, segmentation and object recognition.

Each tool will be defined as follows: image acquisition is capturing an image and digitizing it and then analyzing the problem domain then follow the steps according to the problem. Image enhancement is implemented through time and frequency to enhance the image according to the requirements. Image restoration is storing specific parts of an image using a Point Spread Function. The colour image processing tool is used when the image is black and white. Wavelets and multiresolution processing are used if the images are to be rendered in different degrees or wavelets of resolution. Compression reduces the size of images using specific function. Morphological processing is an external structure of the image using dilation and erosion. Segmentation is implemented by splitting an image into different parts. Object recognition used to recognise and save the image description.

Image processing is faster and more cost-effective. It takes less time to process as well as less film and other equipment to photograph. The Processing of images is more environmentally friendly. No chemicals need to be processed or fixed to capture and process digital images. Printing inks are essential element when digital images are printed. The Microsoft computer vision Application Programming

Interface (API) cloud-based tool allows developers to access advanced image processing and data algorithms, transfer pictures or specify image Uniform Resource Locators (URLs), analyze visual content in numerous ways that support inputs and user selections [87]. Amazon Recognition is a cloud software which is used to incorporatephoto and video analysis to users' applications. It can recognise objects, persons, text, scenes and events of an image or video, or any content that is inappropriate. SimpleCV is an open source computer vision platform that allows users to access various high-powered computer vision libraries such as OpenCV without thinking about bit depths, file formats, colour spaces, buffer management, individual values or matrix versus bitmap storage. Photoshop is a software used to edit digital images.

3.2.2 Computer Vision Systems

Computer vision is a field that aims to enable computers to interpret, recognize and process objects in the same manner as human vision. It is similar to giving intelligence and instincts to a human computer. In fact, it is a difficult task to recognize computer images of different objects. Computer vision is closely associated with artificial intelligence because machines need to understand what they see and then interpret or act appropriately.

Computer vision architecture involves processing digital images via different stages successively. The first stage is image acquisition which captures an image and digitalizes it and then analyses it according to the problem domain. Image Processing is the second stage which is a method to transform an object into a digital form and perform certain operations on it in order to produce an enhanced photo or obtain

useful information from it. Image processing is also a form of signal dispensing where the input is an image, such as a video frame or image, and the output can be an object or image-related features. The third stage is image analysis, which extracts a piece of information, and data processing. This method is typically necessary to ensure that certain assumptions suggested by the system are satisfied before a computer vision approach can be applied to image data. Feature Extraction is the fourth stage in computer vision systems which extract features of the object and are derived from the image data at different levels of complexity. The fifth stage is detection/segmentation where a decision is made at some point in the processing whether points or regions of the image require further processing. High-level processing is the sixth stage where the input is usually a small set of data at this level such as a set of points or an image area that should contain a specific object. Lastly, decision making consist of releasing the final decision needed for the application.

A sparse 3D point model of a large complex scene can be reconstructed from hundreds of partially overlapping photographs. Stereo matching algorithms can create a detailed 3D model of a building facade consisting of hundreds of photographs taken from the internet. Object tracking algorithms can track a person walking in a street. Face detection algorithms combined with colour-based clothing and hair detection algorithms are able to identify individuals in an image. Combining Computer-Generated Imagery (CGI) with live action videos by monitoring the source video feature points measure 3D camera motion and scene form. Automatic authentication in the form of fingerprint recognition and face detection is also the domain of Computer Vision.

The applications of computer vision include Optical Character Recognition (OCR)-interpreting handwritten letter codes and Automatic Number Plate Recognition

(ANPR). An example of a computer vision application is a machine inspection where the quick quality inspection of aircraft wings or auto body parts or X-ray vision defects in steel casting using stereo vision with special lighting. It is also used in retail to classify items for automated checkout lanes. It is used in 3D model creation (photogrammetric) where completely automated 3D photographic aerial models are used in applications like Bing Maps. Moreover, the field of medical imaging utilises computer vision currently and is applied in several ways including capturing preoperative and intra operative images as well as to perform long-term brain morphology studies in individuals as they age.

3.2.3 Artificial Intelligence

AI is the ability of a machine to perform cognitive tasks and act intelligently. The field of AI tries to understand intelligent entities. AI is a new discipline that began in 195 With a help of AI, it is possible for machines to learn from their own experience, adapt to new inputs and perform human-like tasks. AI is widely used in finance, education, healthcare, transportation fields and in other industries such as computer vision, medical diagnosis, robotics and remote sensing.

The father of computer science and AI is Alan Turing who proposed a ‘Turing test’ in 1950 which was designed to provide an operational definition of intelligence. If a machine passes this Turing test, it is said to be intelligent. But no machines have completely passed this test as of yet. There are other indicators of intelligence such as Intelligence Quotient (IQ) tests and brain size, but none of them convey intelligence in machines. According to Daniel Gilbert, there is one fundamental element in which our minds differ from the minds of animals and computers; it can

experience something that has not yet happened. Intelligence is not defined by behaviour but rather by prediction. Humans can read at a high speed by predicting the future of a sentence at a high rate. It is only when your brain predicts badly that you suddenly feel blocked. Humans are not the best decision makers, and this is what AI needs to make better decisions for users. Factors affecting human decision making are loss aversion, sunk-cost effects, farming effort and omission bias.

Computers and robots can exceed the human ability at some tasks that are considered to be 'intelligent' using techniques such as data mining and pattern recognition etc. Lower cognitive tasks that are natural for humans can be extremely complex for machines. For example, a vision system and object recognition, partially concealed objects, same object, different shape, colour, texture and size consistency.

Weak AI are machines that are able to act as if they are intelligent, but their thinking is simulated thinking and not real. Strong AI are machines that act as if they are intelligent and they are thinking. Unfortunately, we still only have Weak AI. If a machine passes the Turing test, it is considered as a Strong AI.

3.2.4 Artificial Neural Network

ANN is defined as an interconnected assembly of nodes like the neural structure of the human brain and can solve different types of problems in an easy manner. The brain works by learning from experiences. ANN is a system that processes information in a similar manner to the biological nervous system. The key aspect of this system is the unique structure of the information processing system. The system is composed of a large number of unified processing elements working together to solve certain issues. It is specifically configured for data classification or pattern

recognition applications via learning processes. The architecture of an ANN is composed of three main layers including an input layer, the hidden layer (one layer or more) and the output layer.

ANN can be trained using a supervised or unsupervised approach. In a supervised approach, ANN is simply trained by matched input and output while the unsupervised approach is an attempt to obtain the ANN to realize the structure of input data. There are several benefits associated with using ANN such as self-learning and large data handling. The advantage of using an ANN is ANN has the ability to learn and train data models for non-linearand complicated relationships. Different applications may be used by an ANN such as image processing, object detection and forecasting.

3.2.5 Deep Learning

Deep learning is a machine learning based model that instructs systems to perform the task the humans likely to do. For instance, deep learning is the basic technology behind the automated cars, helping them to sense the traffic signals and pedestrians. It is also the main idea behind the recognition of audio and voices in different devices such cell phones and tablets. Deep learning becoming famous because it is doing the tasks which could not be performed earlier. A deep learning model is based on the layers of the data which could be pictures, text, or audio, into different and small classification layers. Artificial Intelligence could provide 100% accurate results with close to the human level accuracy and even exceeding the human pace. These models are trained by using large data sets and machine learning techniques such as CNN or ANN which contains many classification layers.

In machine learning techniques the system would guide how to use the model accurately on the graphics, audios, and text. Deep learning models give precision based accurate results even exceeding the human level. These models are framed according to the data given and transforming that data into artificial neural based systems containing large layers of classified data. Deep learning attains more precision and accuracy ever than before which help it to meet the users' expectations. It is used in useful applications such as automated cars. The advances in the past years have shown that artificial intelligence can even surpass the humans in classifying images.

Deep learning needs large amount of classified data. For instance, developing automated cars hundreds of thousands of images and videos. Deep learning requires an excessive amount of power. Elite GPUs have an equal design that is proficient for deep learning. Cloud computing and clusters when combined takes less time as compared before when it took weeks.

As deep learning is consisting of neural networks, so it is also known as deep neural networks. The expression "deep" typically mentions to the quantity of concealed layers in the neural system. Usually neural networks just contain 2-3 concealed layers, while deep systems can have upto of 150 layers. To implement the deep learning models, they must train them. For training these models they need large number of labelled data and neural networks. This will help them to learn the features directly from data without any kind of human interaction.

The CNN is one of the most famous deep neural networks algorithms. It stands for Conventional Neural Network. It involves classified layers of input data and uses 2D convolutional layers to process 2D data.

Contrastive Divergence (CD) algorithm is different training method to approximate Maximum-Likelihood (ML) learning algorithm which represents the relationship between weights and itserror, and it called the gradient. This method implemented to learn the weight of the Restricted Boltzmann Machines (RBMs) with gradient ascent. The formula of this method is shown as follows:

$$\Delta w_{ij}(t+1) = w_{ij}(t) + \eta \frac{\partial \log(p(v))}{\partial w_{ij}}$$

Deep learning applications are utilized in projects from computerized heading to clinical devices. Automobiles companies are using machine learning models identify traffic signs etc. Due to use of Deep learning accidents of walking people has significantly decreased. In aerospace and defence deep learning is utilized to recognize objects from satellites that find special regions and distinguish guarded or unguarded areas for troops. Cancer analysts are trying to identify malignant growth cells using deep learning models and artificial intelligence. UCLA teams have manufactured a microscope that includes a high-dimensional informational sets used to train a deep learning application to precisely recognize cancer cells. Use of Deep learning models helps workers in their field area where there is heavy machinery by identifying people and things in the safe and unsafe zones. Deep learning is being used in the recognizing the audio and voice, such as the devices that detect your speech and give the results according to it. These all functions are done by deep-learning.

3.2.6 Convolutional Neural Network

A Convolutional neural network (CNN) is a type of artificial neural network specifically designed for image recognition. A neural network following the activity of human brain neurons is a patterned hardware and/or software system. CNN is also defined as a different type of multi-layer neural network and each layer of a CNN converts one amount of activations to another through a function. CNN is a special architecture used for deep learning CNN is frequently used in recognizing scenes and objects, and to carry out image detection, extraction and segmentation.

CNN can be categorised in two phases, namely Training and Inference. To build a CNN- based architecture, it applies three key types of layers: Convolutional Layer, Pooling Layer and the Fully Connected Layer. The first layer is a convolutional layer which is the main block of CNN. It takes many filters that are applied to the given image and creates different activation features in the picture. The second layer is pooling which is used to downsample. It will obtain input from non-linear activation and the output will depend on the window size. The last layer is fully connected where a target is identified to determine the category of final output. Due to the three layers, which removes the necessity for feature extraction by using image processing tools, the image data is learned directly by CNN. CNN causes the recognition results to be unique and it might be retrained easily for new recognition missions while it is allowed to build on the pre-existing network. All the following factors have made the usage of CNN significant in the last few years.

If a correct filter is applied to the temporal and spatial dependency in an image, it can be effectively captured by CNN. The number of parameters (weights) will increase rapidly in a neural network with fully connected neurons as the size of the input increases. A convolutional neural network reduces the number of parameters with fewer connections, mutual weights and down sampling. Weight sharing is another major feature of CNNs. CNNs are an efficient extractor for a completely new task or for problems in photo performance, text, audio, video recognition and classification functions. A Convolutional neural network also reduces the number of parameters with fewer connections, mutual weights, and down sampling. Besides that, CNNs remove the need for manual processing of features then discovers the features direct.

CNN Algorithm contains convolutional layers that are represented by an input called map I, many filters K and biases b. In the images case , It may have as input which is an image with height H, width W and C = 3 channels which is red, blue and green such that $I \in RH \times W \times C$ Consequently for many D filters will have $K \in Rk1 \times k2 \times C \times D$ and biases $b \in RD$, one for each filter. The output from this convolution process is shown as follows:

$$(I * K)_{ij} = \sum_{m=0}^{k_1-1} \sum_{n=0}^{k_2-1} \sum_{c=1}^{C} K_{m,n,c} \cdot I_{i+m,j+n,c} + b$$

The convolution procedure implemented previously is the same as the cross-correlation, exclude that the kernel is flipped horizontally and vertically. For simplicity purposes, It should utilize the argument where the input image is grayscale such as single channel C = 1 .The Equation will be transformed as follows:

$$(I * K)_{ij} = \sum_{m=0}^{k_1-1} \sum_{n=0}^{k_2-1} K_{m,n} \cdot I_{i+m,j+n} + b$$

Search engines, recommender systems and social media are the primary fields to use a CNN in identification and classification of objects. Social media, identification procedures and surveillance are using face recognition which is worth mentioning separately. This image recognition section involves more complex images such as pictures that could have human or other living beings, including animals, fish and insects. A banking insurance using optical character recognition has been designed to process symbols that are written and printed. The medical image involves a whole lot of additional data analysis that will spur the initial recognition of the image. A CNN medical image classification detects microorganisms with higher accuracy than the human eye on the X-ray or MRI images. Drug discovery is another important area of health care that uses CNNs extensively. CNN is one of the most innovative implementations used in various fields.

Unlike regular Neural Networks, in the layers of CNN, the neurons are arranged in 3 dimensions: width, height, depth. The neurons in a layer will only be connected to a small region of the layer (window size) before it, instead of all of the neurons in a fully-connected manner. Moreover, the final output layer would have dimensions (number of classes), because by the end of the CNN architecture we will reduce the full image into a single vector of class scores.

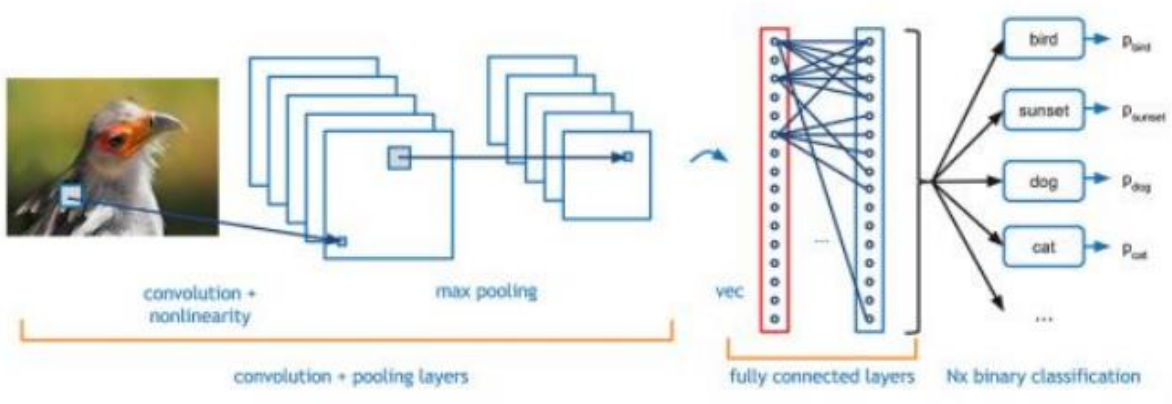


Fig .5 : Convolution Neural Network

1. **Convolution Layer:** In convolution layer we take a small window size [typically of length 5*5] that extends to the depth of the input matrix. The layer consists of learnable filters of window size.

   During every iteration we slid the window by stride size [typically 1], and compute the dot product of filter entries and input values at a given position. As we continue this process well create a 2-Dimensional activation matrix that gives the response of that matrix at every spatial position. That is, the network will learn filters that activate when they see some type of visual feature such as an edge of some orientation or a blotch of some color

2. **Pooling Layer:** We use pooling layer to decrease the size of activation matrix and ultimately reduce the learnable parameters. There are two type of pooling :

a) **Max Pooling:** In max pooling we take a window size [for example window of size 2*2], and only take the maximum of 4 values. Well lid this window and continue this process, so well finally get a activation matrix half of its original Size.

b) **Average Pooling :** In average pooling we take average of all values in a window.

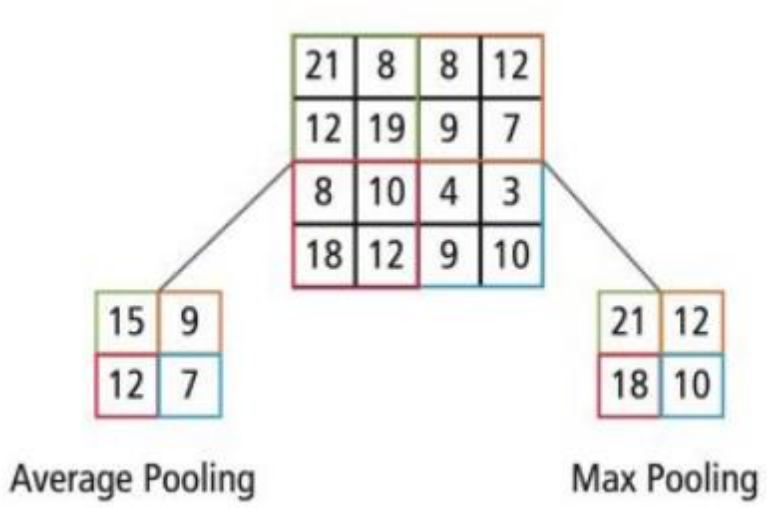


Fig. 6: Types of pooling

3. Fully Connected Layer : In convolution layer neurons are connected only to a local region, while in a fully connected region, well connect the all the inputs to neurons.

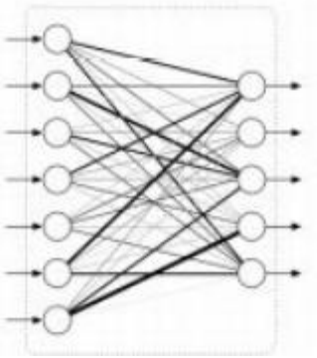


Fig. 7: Fully connected layer

4. Final Output Layer: After getting values from fully connected layer, well connect them to final layer of neurons[having count equal to total number of classes], that will predict the probability of each image to be in different classes.

TensorFlow :

Tensorflow is an open source software library for numerical computation. First we define the nodes of the computation graph, then inside a session, the actual computation takes place. TensorFlow is widely used in Machine Learning.

3.2.7 Keras :

Keras is a high-level neural networks library written in python that works as a wrapper to TensorFlow. It is used in cases where we want to quickly build and test the neural network with minimal lines of code. It contains implementations of commonly used neural network elements like layers, objective, activation functions, optimizers, and tools to make working with images and text data easier.

3.2.8 OpenCV :

OpenCV(Open Source Computer Vision) is an open source library of programming functions used for real-time computer-vision. It is mainly used for image processing, video capture and analysis for features like face and object recognition. It is written

in C++ which is its primary interface, however bindings are available for Python, Java, MATLAB/OCTAVE.

3.3 GESTURE CLASSIFICATION

The approach which we used for this project is :

Our approach uses two layers of algorithm to predict the final symbol of the user.

3.3.1 Algorithm Layer 1:

1. Apply gaussian blur filter and threshold to the frame taken with opencv to get the processed image after feature extraction.

2. This processed image is passed to the CNN model for prediction and if a letter is detected for more than 50 frames then the letter is printed and taken into consideration for forming the word.

3. Space between the words are considered using the blank symbol.

		Predicted																								
		A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y
	A	147	0	0	0	0	0	0	0	0	0	0	0	1	2	0	0	0	0	0	0	0	0	2	0	0
	B	0	139	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	11	0	0	0
	C	0	0	152	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	D	0	0	0	145	0	0	0	0	0	0	8	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	E	0	0	0	0	152	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	F	0	0	0	0	0	135	0	0	0	0	0	4	0	0	0	0	0	1	0	0	2	10	0	0	0
C	G	0	0	0	0	0	0	150	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
o	H	1	0	0	0	0	0	7	143	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	1
r	I	0	0	0	33	0	0	0	0	108	0	2	0	0	0	0	0	0	0	0	7	1	0	0	0	0
r	J	0	0	0	0	0	0	0	0	0	153	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
e	K	0	0	0	0	0	0	0	0	0	0	153	0	0	0	0	0	0	0	0	0	0	0	0	0	0
c	L	0	0	0	0	0	0	0	0	0	0	0	153	0	0	0	0	0	0	0	0	0	0	0	0	0
t	M	0	0	0	0	0	0	0	0	0	0	2	0	152	0	0	0	0	0	0	0	0	0	0	0	0
	N	0	0	0	0	0	0	0	0	0	0	0	0	0	152	0	0	0	0	0	0	0	0	0	0	0
V	O	0	0	0	0	0	0	0	0	0	0	0	0	0	0	154	0	0	0	0	0	0	0	0	0	0
a	P	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	153	0	0	0	0	0	0	0	0	0
l	Q	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	2	147	1	0	0	0	0	0	0	0
u	R	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	150	0	0	0	0	0	0	0
e	S	0	0	0	0	1	0	0	0	0	0	0	0	0	1	10	0	0	0	132	0	0	0	0	8	0
s	T	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	151	0	0	0	0	0
	U	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	35	0	0	115	0	0	0	0
	V	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	151	1	0	0
	W	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	149	0	0
	X	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	148	0
	Y	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	151
	Z	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Fig. 8: Algorithm 1

3.3.2 Algorithm Layer 2:

1. We detect various sets of symbols which show similar results on getting detected.
2. We then classify between those sets using classifiers made for those sets only.

		Predicted Values																							
		A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y
	A	147	0	0	0	0	0	0	0	0	0	0	0	1	2	0	0	0	0	0	0	0	0	2	0	0
	B	0	139	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	11	0	0	0
	C	0	0	152	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	D	0	0	0	153	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	E	0	0	0	0	152	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	F	0	0	0	0	0	135	0	0	0	0	0	4	0	0	0	0	0	0	0	0	3	10	0	0	0
Correct Values	G	0	0	0	0	0	0	150	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
	H	1	0	0	0	0	0	7	143	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	1
	I	0	0	0	0	0	0	0	0	150	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
	J	0	0	0	0	0	0	0	0	0	153	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	K	0	0	0	0	0	0	0	0	0	0	153	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	L	0	0	0	0	0	0	0	0	0	0	0	153	0	0	0	0	0	0	0	0	0	0	0	0	0
	M	0	0	0	0	0	0	0	0	0	0	2	0	152	0	0	0	0	0	0	0	0	0	0	0	0
	N	0	0	0	0	0	0	0	0	0	0	0	0	0	152	0	0	0	0	0	0	0	0	0	0	0
Value	O	0	0	0	0	0	0	0	0	0	0	0	0	0	0	154	0	0	0	0	0	0	0	0	0	0
	P	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	153	0	0	0	0	0	0	0	0	0
	Q	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	2	147	1	0	0	0	0	0	0	0
	R	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	150	0	0	0	0	0	0	0
	S	0	0	0	0	1	0	0	0	0	0	0	0	0	0	10	0	0	0	133	0	0	0	0	8	0
	T	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	151	0	0	0	0	0
	U	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	150	0	0	0	0
	V	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	151	1	0	0
	W	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	149	0	0
	X	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	148	0
	Y	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	151
	Z	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Fig. 9: Algorithm 1 + Algorithm 2

Layer 1:

CNN Model :

1. 1st Convolution Layer: The input picture has resolution of 128x128 pixels. It is first processed in the first convolutional layer using 32 filter weights (3x3 pixels each). This will result in a 126X126 pixel image, one for each Filter-weight.kk**2. 1st**

2. 1st Pooling Layer: The pictures are down sampled using max pooling of 2x2 i.e. we keep the highest value in the 2x2 square of array. Therefore, our picture is down sampled to 63x63 pixels.

3. 2nd Convolution Layer: Now, these 63 x 63 from the output of the first pooling layer is served as an input to the second convolutional layer. It is processed in the

second convolutional layer using 32 filter weights (3x3 pixels each).This will result in a 60 x 60 pixel image.

4. 2nd Pooling Layer: The resulting images are down sampled again using max pool of 2x2 and are reduced to 30 x 30 resolutions of images.

5. 1st Densely Connected Layer: Now these images are used as an input to a fully connected layer with 128 neurons and the output from the second convolutional layer is reshaped to an array of 30x30x32 =28800 values. The input to this layer is an array of 28800 values. The output of these layer is fed to the 2nd Densely Connected Layer. We are using a dropout layer of value 0.5 to avoid overfitting.

6. 2nd Densely Connected Layer: Now the output from the 1st Densely Connected Layer is used as an input to a fully connected layer with 96 neurons.

7. Final layer: The output of the 2nd Densely Connected Layer serves as an input for the final layer which will have the number of neurons as the number of classes we are classifying (alphabets + blank symbol).

Activation Function :

We have used ReLu (Rectified Linear Unit) in each of the layers (convolutional as well as fully connected neurons). ReLu calculates max(x,0) for each input pixel. This adds nonlinearity to the formula and helps to learn more complicated features. It helps in removing the vanishing gradient problem and speeding up the training by reducing the computation time.

Pooling Layer : We apply Max pooling to the input image with a pool size of (2, 2) with relu activation function. This reduces the amount of parameters thus lessening the computation cost and reduces overfitting.

Dropout Layers: The problem of overfitting, where after training, the weights of the network are so tuned to the training examples they are given that the network doesn't perform well when given new examples. This layer "drops out" a random set of activations in that layer by setting them to zero. The network should be able to provide the right classification or output for a specific example even if some of the activations are dropped out[5].

Optimizer: We have used Adam optimizer for updating the model in response to the output of the loss function. Adam combines the advantages of two extensions of two stochastic gradient descent algorithms namely adaptive gradient algorithm (ADA GRAD) and root mean square propagation(RMSProp)

Layer 2: We are using two layers of algorithms to verify and predict symbols which are more similar to each other so that we can get us close as we can get to detect the symbol shown. In our testing we found that following symbols were not showing properly and were giving other symbols also:

1. For D : R and U
2. For U : D and R
3. For I : T, D, K and I
4. For S : M and N So to handle above cases we made three different classifiers for classifying these sets:

3.4 System Implementations

3.4.1 Hand Gestures Input

In this experiment, hand gestures are fed as input into CNN. Figure 1, Figure 4 and Figure 7 show twelve random hand gestures recorded in short distance with a plain background using a holoscopic imaging camera system. Some motions are 2D while others are 3D. The images are pre-processed before extracting videos in terms of some steps:

1- For Figure 1, the resolution of the camera used is full High Definition (HD) while for Figure 4 and Figure7 the resolution is 4K.

2- The camera used in this experiment is a holoscopic imaging camera system with multi lenses. The number of lenses shown in Figure 1 is 47 for x-axis whereas for y-axis it is 84. For Figure 4 and Figure 7, the number of lenses is decreased to 31 on the xaxis is and 55 on the y-axis.

3- The generated images are converted from RGB to grey and images need to be resized to 135 × 75.

4- In figure 2, the image is rotated 0.30 degrees to adjust the image position while for figure 5 and figure 8 are rotated 180.20 degrees.

5- Divide lens into seven segments i.e. 7 × 7, the X segment is a constant of 4 while Y is changeable to 2, 4 and

6- Create twelve separate directories for three different left, centre and right images and convert them from RGB to grey colour. Lastly, resize these grey images to size 135 × 75.

7- Combine each left, centre and right images for three people in one directory.

8- Combine the three images i.e. left, centre and right to get one image with a size 405 × 75 in Figure 3, Figure 6 and Figure 9.

3.4.2 Data Set Generation

The dataset consists of BGR images. First captured each frame shown by webcam of the machine. In each frame Region of interest (ROI) is denoted by a blue square on the top-right side as shown in the image below.

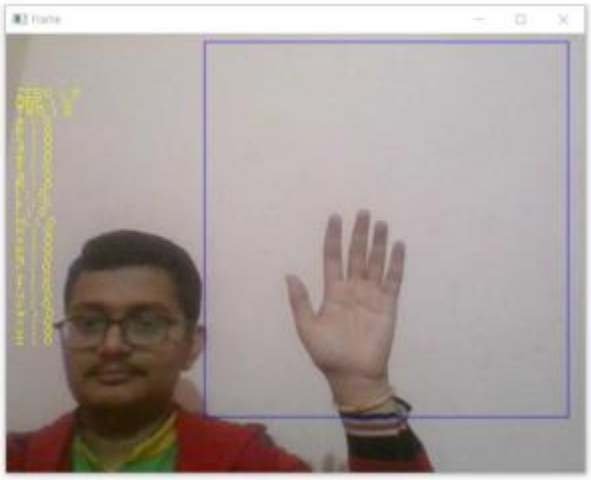


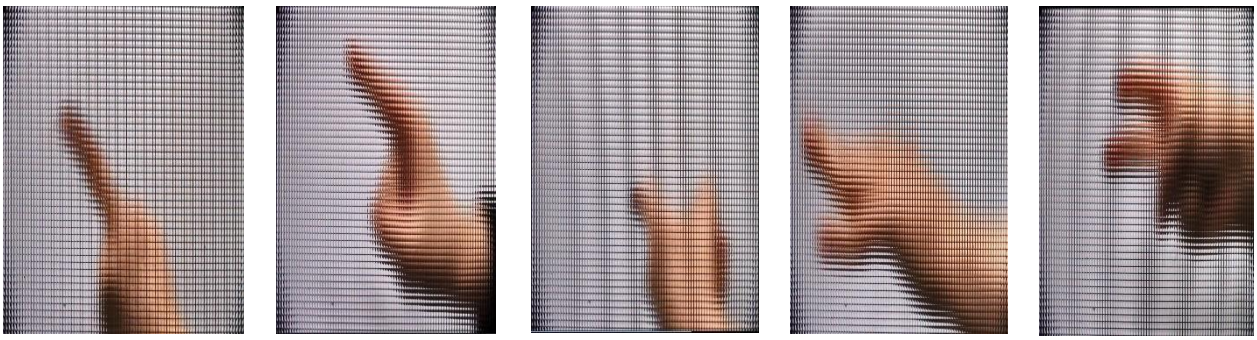
Fig. 10: Pre- extraction first person's hand motions in short distance.

From each frame, we extract ROI which contains hand. ROI is RBG so it is converted to grayscale as shown below.

Fig. 11: Gaussian blur filter is applied to greyscale ROI

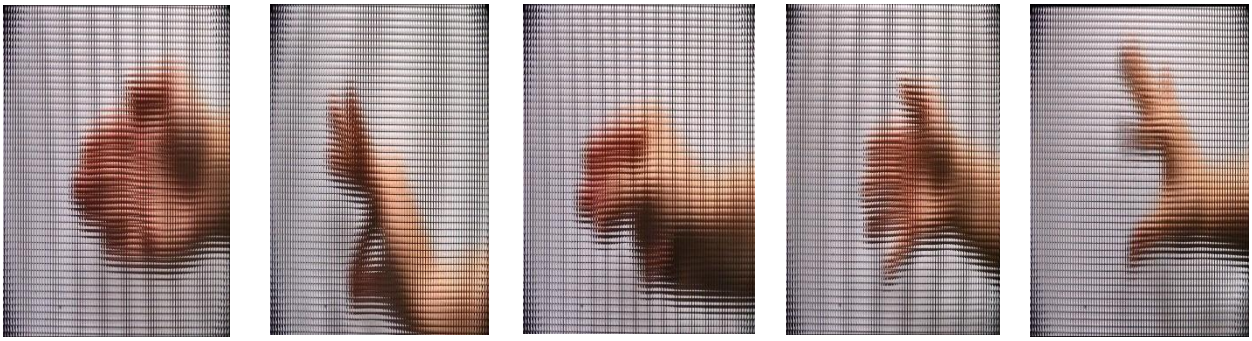
Then Gaussian blur filter is applied to grayscale ROI which is later used for extracting various features of our image. After applying ROI looks like below.

1-Pre- extraction first person's hand motions in short distance



a) Sweep motion b) Shrink motion c) Circular motion d) Squeeze motion

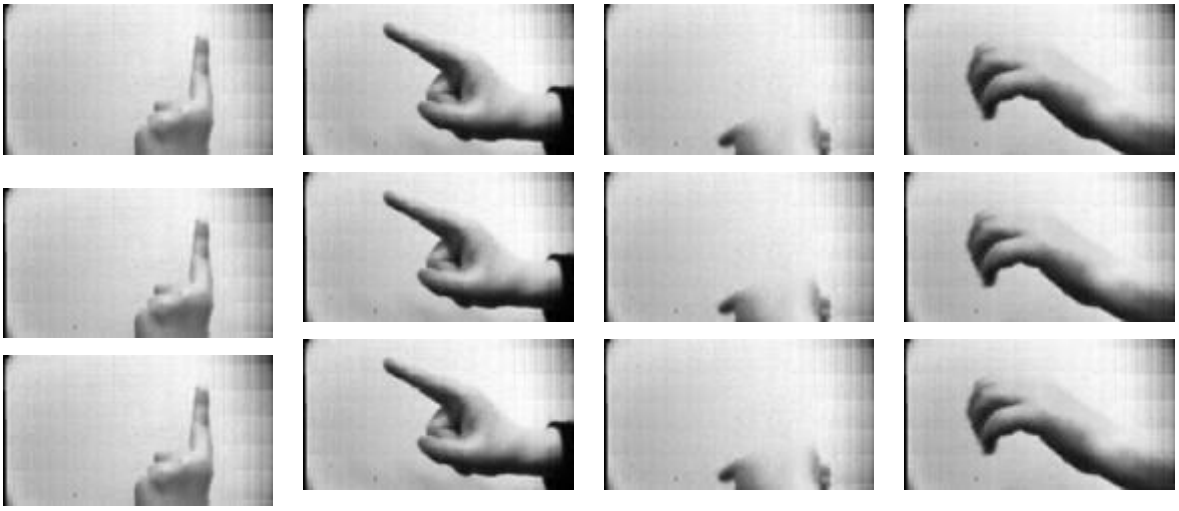
e) 2 Fingers Shrink



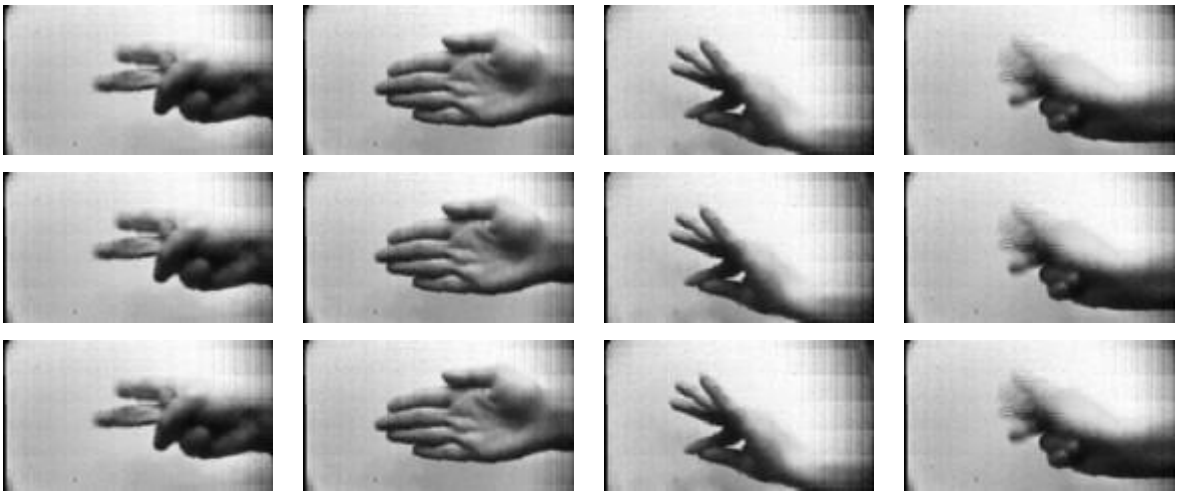
f) Back/Forth g) Rub motion h) Click motion i) Dance motion j) Pinch motion

Fig. 12: Pre- extraction first person's hand motions in short distance.

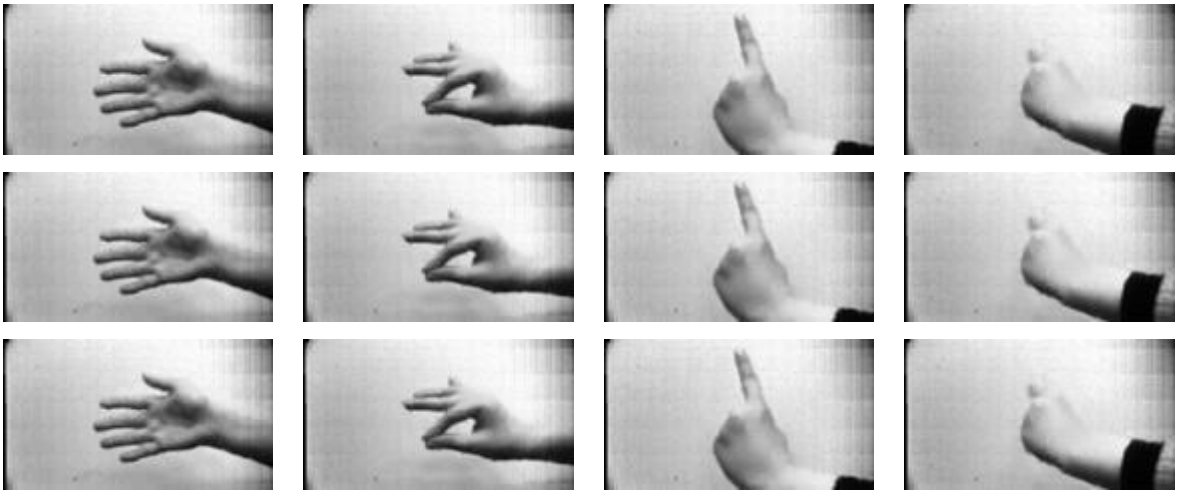
2-Post-extraction first person's hand motions in short distance single (LCR)



a) Sweep motion b) Shrink motion c) Circular motion d) Squeeze motion



e) 2 Fingers Shrink f) Back/Forth g) Rub motion h) Click motion



i) Dance motion j) Pinch motion k) write motion l) Click motion 2

Fig. 13: Post- extraction first person's hand motions in short distance

3-Post-extraction first person's hand motions in short distance combined (LCR)

a) Sweep motion

b) Shrink motion

c) Circular motion

d) Squeeze motion

e) 2 Fingers Shrink

f)Back/Forth

g) Rub motion

h) Click motion

i)Dance motion

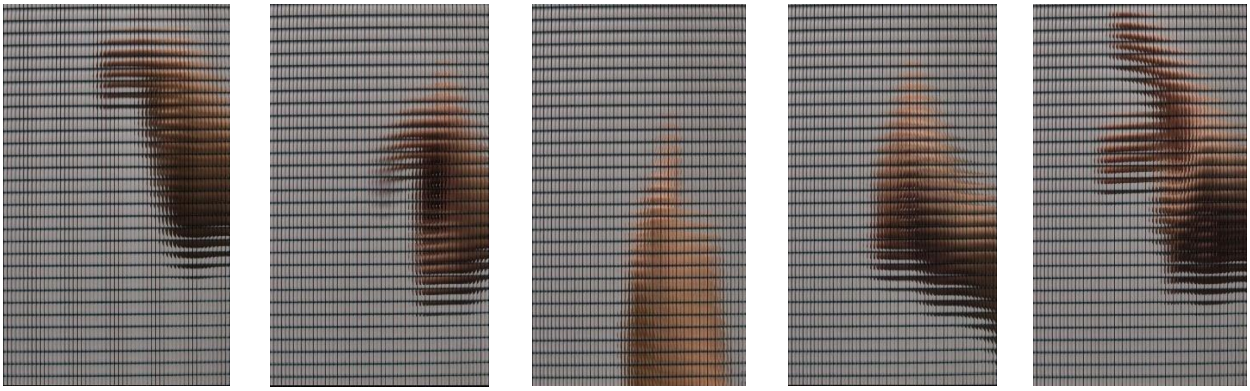
j)Pinch motion

write motion

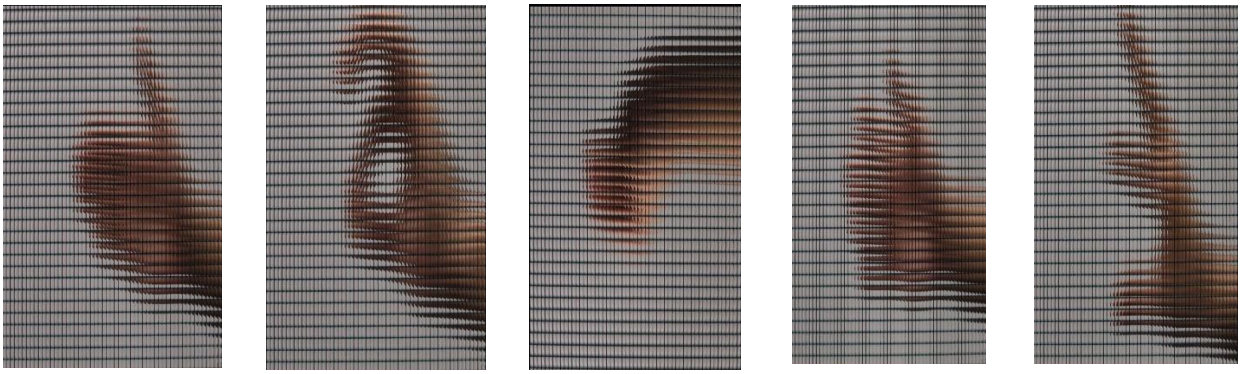
l) Click motion 2

Fig. 14: Post- extraction first person's hand motions in short distance

4-Pre- extraction second person's hand motion in short distance



a) Sweep motion b) Shrink motion c) Circular motion d) Squeeze motion e) 2 Fingers Shrink



f) Back/Forth g) Rub motion h) Click motion i) Dance motion
j) Pinch motion

Fig. 15: Pre- extraction second person's hand motion in short distance

3.4.3 Finger spelling sentence formation Implementation :

1. Whenever the count of a letter detected exceeds a specific value and no other letter is close to it by a threshold we print the letter and add it to the current string(In our code we kept the value as 50 and difference threshold as 20).
2. Otherwise we clear the current dictionary which has the count of detections of present symbol to avoid the probability of a wrong letter getting predicted.
3. Whenever the count of a blank(plain background) detected exceeds a specific value and if the current buffer is empty no spaces are detected.
4. In other case it predicts the end of word by printing a space and the current gets appended to the sentence below.

3.4.4 Autocorrect Feature:

A python library Hunspell_suggest is used to suggest correct alternatives for each (incorrect) input word and we display a set of words matching the current word in which the user can select a word to append it to the current sentence. This helps in reducing mistakes committed in spellings and assists in predicting complex words.

3.4.5 Training and Testing :

We convert our input images(RGB) into grayscale and apply gaussian blur to remove unnecessary noise. We apply adaptive threshold to extract our hand from the background and resize our images to 128 x 128.

We feed the input images after preprocessing to our model for training and testing after applying all the operations mentioned above.

The prediction layer estimates how likely the image will fall under one of the classes. So the output is normalized between 0 and 1 and such that the sum of each values in each class sums to 1. We have achieved this using softmax function.

At first the output of the prediction layer will be somewhat far from the actual value. To make it better we have trained the networks using labeled data. The cross-entropy is a performance measurement used in the classification. It is a continuous function which is positive at values which is not same as labeled value and is zero exactly when it is equal to the labeled value. Therefore we optimized the cross-entropy by minimizing it as close to zero.

To do this in our network layer we adjust the weights of our neural networks. TensorFlow has an inbuilt function to calculate the cross entropy.

As we have found out the cross entropy function, we have optimized it using Gradient Descent in fact with the best gradient descent optimizer is called Adam Optimizer.

CHAPTER- 4
Results and Discussion

4.1 Result

We have achieved an accuracy of 95.8% in our model using only layer 1 of our algorithm , and using the combination of layer 1 and layer 2 we achieve an accuracy of 98.0%, which is a better accuracy then most of the current research papers on American sign language. Most of the research papers focus on using devices like kinect for hand detection. In [7] they build a recognition system for flemish sign language using convolutional neural networks and kinect and achieve an error rate of 2.5%. Ina recognition model is built using a hidden Markov model classifier and a vocabulary of 30 words and they achieve an error rate of 10.90%. In [9] they achieve an average accuracy of 86% for 41 static gestures in Japanese sign language. Using depth sensors map [10] achieved an accuracy of 99.99% for observed signers and 83.58% and 85.49% for new signers. They also used CNN for their recognition system. One thing should be noted that our model doesn't uses any background subtraction algorithm whiles some of the models present above do that. So once we try to implement background subtraction in our project the accuracies may vary. On the other hand most of the above projects use kinect devices but our main aim was to create a project which can be used with readily available resources.

4.2 Training Result

Epoch	Validation Loss	Validation Accuracy
1	0.7627	71.06
2	0.2387	91.25
3	0.1634	93.96
4	0.1295	94.08

4.3 Testing Result

The loss obtained on testing set was 0.1182 with an accuracy of 95.50%.

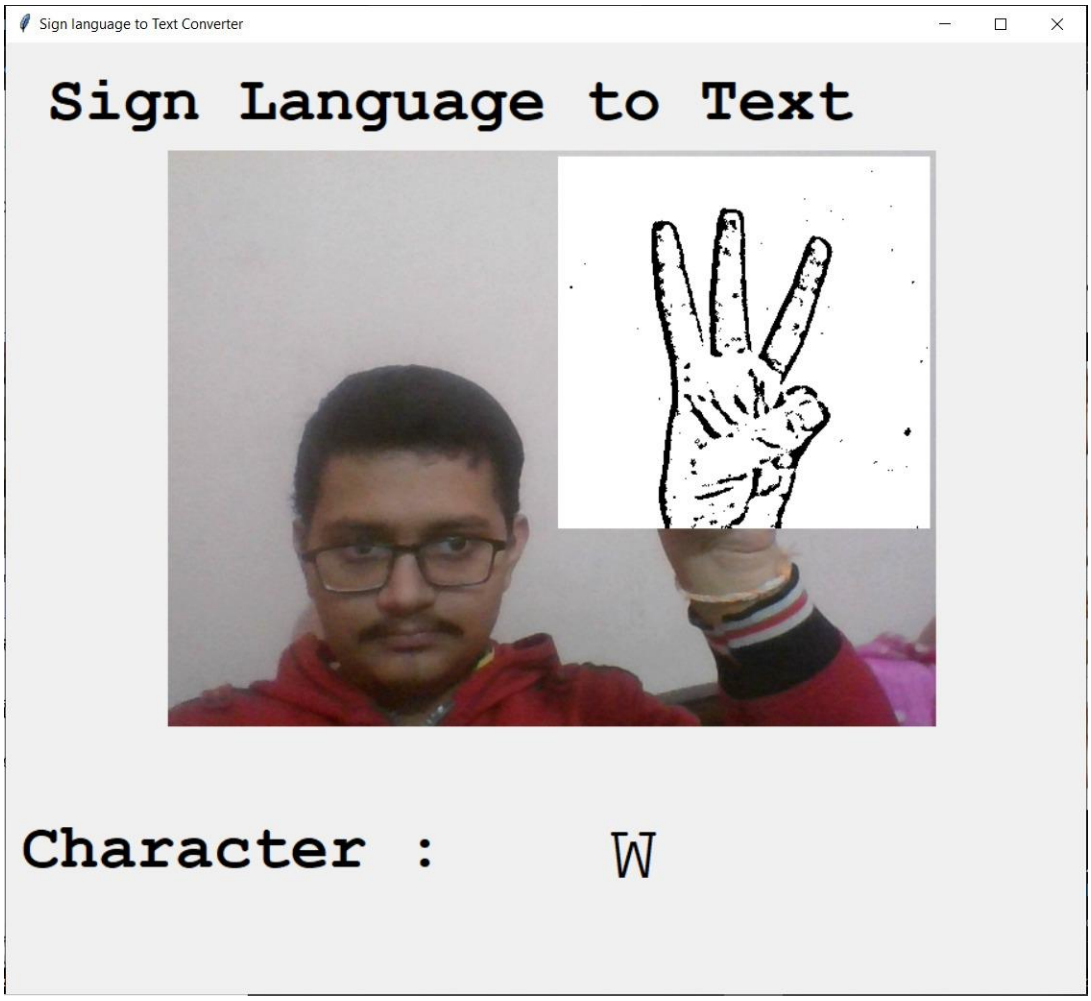


Fig. 16: Final result output

CHAPTER-5

Conclusion and Future Scope

5.1 Conclusion

In this report, a functional real time vision based American sign language recognition for D&M people have been developed for asl alphabets. We achieved final accuracy of 98.0% on our dataset. We are able to improve our prediction after implementing two layers of algorithms in which we verify and predict symbols which are more similar to each other. This way we are able to detect almost all the symbols provided that they are shown properly, there is no noise in the background and lighting is adequate.

5.2 Future Prospects

The task of image captioning can be put to great use for the visually impaired. The model proposed can be integrated with an android or ios application to work as a real-time scene descriptor. The accuracy of the model can be improved to achieve state of the art results by hyper tuning the parameters.

We are planning to achieve higher accuracy even in case of complex backgrounds by trying out various background subtraction algorithms. We are also thinking of improving the preprocessing to predict gestures in low light conditions with a higher accuracy.

Reference

[1]T. Yang, Y. Xu, and “A. , Hidden Markov Model for Gesture Recognition”, CMU-RI-TR-94 10, Robotics Institute, Carnegie Mellon Univ.,Pittsburgh,PA.

[2]Pujan Ziaie, Thomas M¨uller , Mary Ellen Foster , and Alois Knoll“A Na¨ive Bayes Munich,Dept. of Informatics VI, Robotics and Embedded Systems,Boltzmannstr. 3, DE-85748 Garching, Germany.

[3]https://docs.opencv.org/2.4/doc/tutorials/imgproc/gausian_median_blur_bilateral_filter/gausian_median_blur_bilateral_filter.html

[4]Mohammed Waleed Kalous, Machine recognition of Auslan signs using PowerGloves: Towards large-lexicon recognition of sign language.

[5]aeshpande3.github.io/A-Beginner%27s-Guide-To-Understanding-Convolutional-Neural-Networks-Part-2/

[6]http://www-i6.informatik.rwth-aachen.de/~dreuw/database.php

[7] Pigou L., Dieleman S., Kindermans PJ., Schrauwen B. (2019) Sign Language Recognition Using Convolutional Neural Networks. In: Agapito L., Bronstein M., Rother C. (eds) Computer Vision - ECCV 2019 Workshops. ECCV 2014. Lecture Notes in Computer Science, vol 8925. Springer, Cham

[8]Zaki, M.M., Shaheen, S.I.: Sign language recognition using a combination of new vision based features. Pattern Recognition Letters 32(4), 572–577 (2011) 25

[9] N. Mukai, N. Harada and Y. Chang, "Japanese Fingerspelling Recognition Based on Classification Tree and Machine Learning," 2017 Nicograph International (NicoInt), Kyoto, Japan, 2017, pp. 19-24. doi:10.1109/NICOInt.2017.9

[10]Byeongkeun Kang , Subarna Tripathi , Truong Q. Nguyen ”Real-time sign