“HAND GESTURE TO TEXT AND SPEECH CONVERTER”

Major Project (Phase-II) Report Submitted

To

Chhattisgarh Swami Vivekanand Technical University Bhilai, India

for

The award of the degree

of

BACHELOR OF TECHNOLOGY

In

COMPUTER SCIENCE & ENGINEERING

By

Anjali Jha	Sudhanshu Shukla	Simran Banarase
B.Tech. (CSE)	B.Tech. (CSE)	B.Tech. (CSE)
Roll No.-303302219017	Roll No.-303302219108	Roll No.-303302219103
En. No.-BH3700	En. No.-BH3791	En. No.-BH3786

Under the Guidance of
Mrs. Preeti Tuli
Asst. Professor
Department of Computer Science & Engineering
S.S.I.P.M.T, Raipur

Department of Computer Science & Engineering
Shri Shankaracharya Institute of Professional Management & Technology Raipur (C.G.)

Session: 2022-23

DECLARATION BY THE CANDIDATE

We the undersigned solemnly declare that the thesis entitled **"Hand Gesture to Text and Speech Converter"** is based our own work carried out during the course of our study under the supervision of **Mrs. Preeti Tuli**, Asst. Prof., Department of Computer Science and Engineering, Shri Shankaracharya Institute of Professional Management & Technology, Raipur (C.G.), India.

(Signature of Candidate 1)	**(Signature of Candidate 2)**	**(Signature of Candidate 3)**
Name: Anjali Jha	**Name: Sudhanshu Shukla**	**Name: Simran Banarase**
Roll No.:303302219017	Roll No.: 303302219108	Roll No.: 303302219103
En. No.: BH3700	En. No.: BH3791	En. No.: BH3786

Department of Computer Science & Engineering

Shri Shankaracharya Institute of Professional Management & Technology

Raipur (C.G.)

CERTIFICATE OF THE SUPERVISOR

This is to certify that the incorporation in the thesis "**Hand Gesture to Text and Speech Converter"** is a record of project work carried out by Anjali Jha, bearing Roll No. 303302219017 Enrollment No. BH3700, Sudhanshu Shukla, bearing Roll No. 303302219108 Enrollment No. BH3791, Simran Banarase, bearing Roll No. 303302219103 Enrollment No. BH3786 under my guidance and supervision for the fulfillment of the award of degree of Bachelor of Technology in the faculty of Computer Science & Engineering of Chhattisgarh Swami Vivekanand Technical University, Bhilai (C.G.), India.

To the best of my knowledge and belief the report

i. Embodies the work of the candidate themselves

ii. Has duly been completed

iii. Fulfills the requirement of the ordinance relating to the Bachelor of Technology degree of the University

iv. Is up to the desired standard both in respect of contents and language for being referred to the examiners.

(Signature of Supervisor)

Mrs. Preeti Tuli

Asst. Professor, Dept of CSE

S.S.I.P.M.T, Raipur, C.G

Forwarded to

Chhattisgarh Swami Vivekanand Technical University, Bhilai

(Signature of HOD)

Dr. J.P.Patra

HOD, Dept of CSE

S.S.I.P.M.T, Raipur, C.G

(Signature of the Principal)

Dr. Alok Kumar Jain

S.S.I.P.M.T, Raipur, C.G

CERTIFICATE BY THE EXAMINERS

This is to certify that the project thesis entitled **"Hand Gesture to Text and Speech Converter"** was submitted by Anjali Jha student of B.Tech. (CSE) (Roll No. 303302219017, Enrollment No. BH3700), Sudhanshu Shukla student of B.Tech. (CSE) (Roll No. 303302219108 Enrollment No. BH3791), Simran Banarase student of B.Tech. (CSE) (Roll No. 303302219103 Enrollment No. BH3786) has been examined by the undersigned as a part of the examination and is hereby recommended for the award of the degree of **Bachelor of Technology in Computer Science and Engineering** of Chhattisgarh Swami Vivekanand Technical University, Bhilai (C.G.) India.

Internal Examiner	**External Examiner**
Date:	**Date:**

ACKNOWLEDGEMENT

We thank the almighty for giving us the encourage and perseverance in completing **"Hand Gesture to Text and Speech Converter"**. This project itself is an acknowledgement for all those people who have given us their heartfelt co-operation in making this project a grand success. We extend our sincere thanks to **Mr. Nishant Tripathi, Chairman (B.G.) of our college,** for providing sufficient infrastructure and good environment in the college to complete our course.

We are thankful to our **Principal, Dr. Alok Kumar Jain** for providing the necessary infrastructure and labs and also permitting to carry out this project. With extreme jubilance and deepest gratitude, we would like to thank **Head of the Computer Science and Engineering Department, Dr. J.P.Patra** for his constant encouragement.

We are thankful to our project coordinator **Mr. Yogesh Kumar Rathore, Assistant Professor, CSE Department,** for his support and valuable suggestions regarding project work. We are greatly indebted to project guide **Mrs. Preeti Tuli, Assistant Professor, CSE Department,** for providing valuable guidance and support at every stage of this project work. We are profoundly grateful towards the unmatched services rendered by her. Our special thanks to all the faculty of Computer Science and Engineering and peers for their valuable advises at every stage of this work.

(Signature of Candidate 1)	(Signature of Candidate 2)	(Signature of Candidate 3)
Name: Anjali Jha	**Name: Sudhanshu Shukla**	**Name: Simran Banarase**
Roll No.: 303302219017	Roll No.: 303302219108	Roll No.: 303302219103
En. No.: BH3700	En. No.: BH3791	En. No.: BH3786

ACKNOWLEDGEMENT –AICTE IDEA Lab

We have taken efforts in this project. However, it would not have been possible without the kind support and help of AICTE-IDEA Lab at SSIPMT, Raipur. We would like to extend our sincere thanks to all the gurus, mentors and support staff of Idea lab.

LIST OF ABBREVIATIONS

SRS	System Requirement Specification
DFD	Data Flow Diagram
ANN	Artificial Neural Network
CNN	Convolutional Neural Network
UML	Unified Modelling Language
ER	Entity Relationship
ER-D	Entity Relationship Diagram
ASL	American Sign Language
LDA	Linear Discriminant Analysis

LIST OF FIGURES

ABSTRACT

Sign Language Recognition is one of the most growing fields of research area. Many new techniques have been developed recently in this area. The Sign Language is mainly used for communication of deaf-dumb people. The proposed system contains four modules such as: pre-processing and hand segmentation, feature extraction, sign recognition and sign to text. By using image processing the segmentation can be done. Some of the features are extracted such as Eigen values and Eigen vectors which are used in recognition. The Linear Discriminant Analysis (LDA) algorithm was used for gesture recognition and recognized gesture is converted into text and voice format. The proposed system helps to dimensionality reduction.

Keywords: - Speech, Hand Gesture,, Image Processing

TABLE OF CONTENTS

CHAPTER-I

INTRODUCTION

1. Introduction

American Sign Language (ASL) is natural syntax that has the same etymological homes as being speaking languages, having completely different grammar, ASL can be express with destiny of actions of the body. In native America, people who are deaf or can't see, it's a reliable source of absurdity. There is not any formal or familiar form of sign language. Different signal languages are speculating in particular areas.

For a case, British Sign Language (BSL) is an entirely different language from an ASL, and USA people who familiarise with ASL would not easily understand BSL. Some nations adopt capabilities of ASL of their sign languages. Sign language is a way of verbal exchange via human beings diminished by speech and listening to loss. Around 360 million human beings globally be afflicted via unable to hearing loss out of which 328000000 are adults and 32000000 children. hearing impairment extra than 40 decibels in the better listening to ear is referred as disabling listening to loss.

Thus, with growing range of people with deafness, there is moreover a rise in demand for translators. Minimizing the verbal exchange gap among listening to impaired and regular humans turns into a want to make certain effective conversation among all. Sign language translation is one of the amongst most growing line of research nowadays and its miles the maximum natural manner of communication for the humans with hearing impairments. A hand gesture recognition gadget can offer an opportunity for deaf people to talk with vocal humans without the need of an interpreter. The system is built for the automated conversion of ASL into textual content and speech. A massive set of samples has been utilized in proposed device to understand isolated phrases from the same old American sign language which may be concerned about the use of virtual camera.

Sign language is a visual language and consists of 3 major components -

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.

Considering all the sign language alphabets and terms, the database includes one thousand special gesture images. The proposed system intends to understand some very fundamental elements of signal language and to translate them to text and audio.

American Sign Language is a visible language. Along with the signing, the thoughts techniques linguistic data through the vision. The form, placement, motion of hands, in addition to facial expressions, frame movements, every play essential factor in convey facts. Sign language isn't a normal language each the entire USA. It Has its very own signal 6 language, and areas have vernaculars, like the numerous languages are spoken anywhere inside the globally speaking language, the detection rate by the ASL language as in compare to the grammatical accuracy is of 90 % percentage of institutions commonly use Indian sign language. The amazing elements of India it [ISL] has a bit difference in signing however the grammar is identical at a few stages in the U.S.A. The Deaf humans in India remember the fact that it’s plenty better than one-of-a-kind sign languages on the grounds that it's far a natural method for them, they observe via the herbal interaction with the human beings around them. The stages of sign language acquisition are equal as spoken languages, the toddlers begin with the aid of rambling with their hands. Since India doesn't have many Institutions for growing Indian sign language there is lack of understanding a number of the human beings and some Institution indicates to select ASL over ISL without right knowledge.

In our project we primarily focus on producing a model which can recognize Fingerspelling 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 -

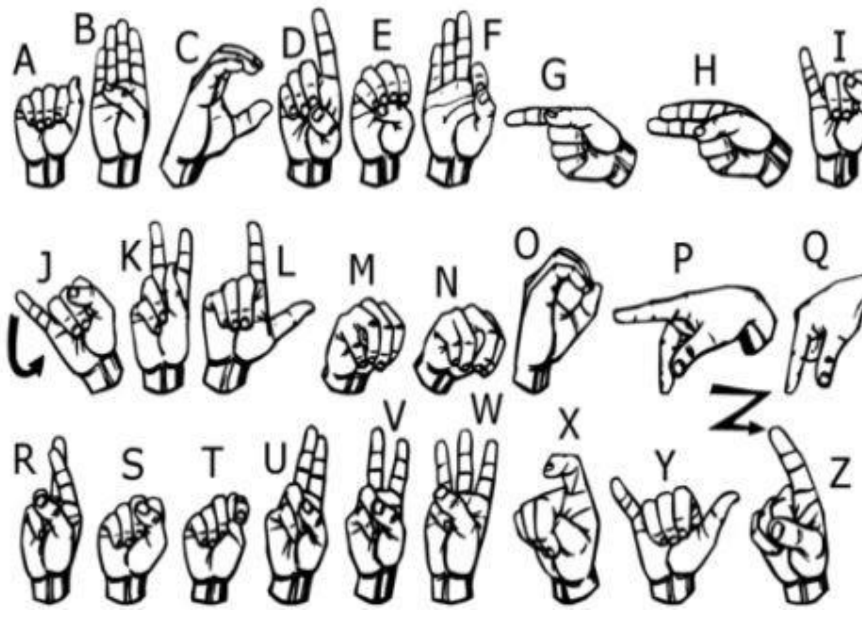


Fig. 1 Sign Language Gesture (ASL)

A hand gesture to speech and text converter is an innovative technology that enables users to communicate using hand gestures, which are converted into spoken words and written text. This technology leverages computer vision, machine learning, and natural language processing techniques to interpret and understand the gestures made by individuals, providing them with an alternative and accessible means of communication. By recognizing and interpreting hand gestures in real-time, the converter translates the gestures into spoken language and displays the corresponding text output, facilitating effective communication between individuals.

The hand gesture to speech and text converter holds significant potential in various domains, including accessibility, assistive technology, education, gaming, and more. It offers an inclusive communication solution for individuals with speech or hearing impairments, allowing them to express themselves and engage in conversations with others. Additionally, the converter can assist people with physical disabilities or limited mobility by enabling them to control devices, perform tasks, and interact with their surroundings using hand gestures.

The technology behind the hand gesture to speech and text converter involves several key components. First, a robust gesture recognition system is employed to accurately detect and interpret hand gestures. This system utilizes computer vision algorithms and machine learning models trained on large datasets to recognize various hand shapes, movements, and patterns. Once a gesture is recognized, it is mapped to a corresponding speech output using natural language processing algorithms. Simultaneously, the gesture is converted into written text, which is displayed on a screen or device interface.

The application of hand gesture to speech and text conversion extends beyond individual communication. It can be integrated into educational tools for teaching sign language or enhancing language learning, enabling learners to practice and improve their communication skills. In gaming and virtual reality, the converter offers immersive and interactive experiences, allowing players to control characters, perform in-game actions, and communicate with other players using hand gestures. Moreover, in professional settings, the converter can assist public speakers, presenters, and lecturers in delivering impactful presentations by using hand gestures to control slides, trigger multimedia content, and emphasize key points.

However, the development and implementation of a hand gesture to speech and text converter come with their own set of challenges. These challenges include achieving accurate gesture recognition, accommodating variations in gestures across individuals and cultures, minimizing noise and interference, ensuring real-time processing, resolving ambiguity in gesture interpretation, and designing an intuitive user interface. Overcoming these challenges requires advanced algorithms, continuous research, user feedback, and interdisciplinary collaboration.

In conclusion, the hand gesture to speech and text converter is a transformative technology that enables individuals to communicate effectively using hand gestures. It has the potential to enhance accessibility, assistive technology, education, gaming, and various other domains. With ongoing advancements, this technology can open new possibilities for inclusive communication and interaction, making a positive impact on the lives of individuals worldwide.

CHAPTER-II

LITERATURE REVIEW & PROBLEM IDENTIFICATION

2.1 Literature Review

A various hand gestures were recognized with different methods by different researchers in which were implemented in different fields. The recognition of various hand gestures were done by vision based approaches, data glove based approaches, soft computing approaches like Artificial Neural Network, Fuzzy logic, Genetic Algorithm and others like PCA, Canonical Analysis, etc. The recognition techniques are divided into three broad categories such as Hand segmentation approaches, Feature extraction approaches and Gesture recognition approaches. "Application research on face detection technology uses Open CV technology in mobile augmented reality" introduces the typical technology. Open source computer vision library, Open CV for short is a cross-platform library computer vision based on open source distribution. The Open CV, with C language provides a very rich visual processing algorithm to write it part and combined with the characteristics of its open source. Data gloves and Vision based method are commonly used to interpret gestures for human computer interaction. The sensors attached to a glove that finger flexion into electrical signals for determining the hand posture in the data gloves method. The camera is used to capture the image gestures in the vision based method. The vision based method reduces the difficulties as in the glove based method.

"Hand gesture recognition and voice conversion system for dumb people" proposed lower the communication gap between the mute community and additionally the standard world. The projected methodology interprets language into speech. The system overcomes the necessary time difficulties of dumb people and improves their manner. Compared with existing system the projected arrangement is simple as well as compact and is possible to carry to any places. This system converts the language in associate text into voice that's well explicable by blind and ancient people. The language interprets into some text kind displayed on the digital display screen, to facilitate the deaf people likewise. In world applications, this system is helpful for deaf and dumb of us those cannot communicate with ancient person. Conversion of RGB to gray scale and gray scale to binary conversion introduced in the intelligent sign language recognition using image processing. Basically any colour image is a combination of red, green, blue colour. A computer vision system is implemented to select whether to differentiate objects using colour or black and white and, if colour, to decide what colour space to use (red, green, blue or hue, saturation, luminosity).

In the recent years there has been tremendous research done on the hand gesture recognition.

With the help of literature survey, we realized that the basic steps in hand gesture recognition are: -

- Data acquisition
- Data pre-processing
- Feature extraction
- Gesture classification

2.1.1 Data acquisition:

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

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.

Vision based approach:

In vision-based methods, the computer webcam is the input device for observing the information of hands and/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, thereby reducing cost. These systems tend to complement biological vision by describing artificial vision systems that are implemented in software and/or hardware. The main challenge of vision-based hand detection ranges from coping with the large variability of the human hand's appearance due to a huge number of hand movements, to different skin-color possibilities as well as to the variations in viewpoints, scales, and speed of the camera capturing the scene.

2.1.2 Data Pre-Processing and Feature extraction for vision-based approach:

Researchers have explored various techniques for speech synthesis and text-to-speech conversion to generate natural and intelligible speech output. Concatenative synthesis and statistical parametric synthesis are two common approaches used in this context. Li et al. (2015) employed a concatenative synthesis method that combines pre-recorded speech units to generate speech output. Schroeter et al. (2012) and Tachibana et al. (2016) utilized statistical parametric synthesis models to generate expressive and high-quality speech from text inputs. These studies highlight the importance of selecting appropriate speech synthesis techniques for accurate and natural-sounding speech generation.

NLP algorithms are employed to extract meaning from hand gestures and generate coherent textual output. Nguyen et al. (2019) proposed a gesture-based communication system that incorporated gesture-based language models and semantic analysis to improve the accuracy and meaningfulness of the converted text. Perez et al. (2017) developed a similar approach, utilizing NLP techniques to map hand gestures to specific words or phrases. These studies demonstrate the potential of NLP in enhancing the quality and accuracy of the converted text, enabling more effective communication.

Several existing systems have implemented hand gesture to speech and text conversion interfaces. For instance, Li et al. (2018) developed a real-time system that converts American Sign Language (ASL) gestures to speech. The system combined computer vision techniques with ASL gesture recognition algorithms and employed text-to-speech synthesis to generate spoken output. Rahman et al. (2019) proposed a gesture-based communication system that translated hand movements to text, which was then converted to speech using text-to-speech synthesis.

- In the approach for hand detection combines threshold-based colour detection with background subtraction. We can use AdaBoost face detector to differentiate between faces and hands as they both involve similar skin-color.

- We can also extract necessary image which is to be trained by applying a filter called Gaussian Blur (also known as Gaussian smoothing). The filter can be easily applied using open computer vision (also known as OpenCV) .

- For extracting necessary image which is to be trained we can use instrumented gloves. This helps reduce computation time for Pre-Processing and gives 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 skin colorur 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 segmenting the hand out of a random background we keep background of hand a stable single colour so that we don't need to segment it on the basis of skin colour. This would help us to get better results.

2.1.3 Gesture Classification:

Various techniques have been explored in the literature to accurately detect and interpret hand gestures. Computer vision-based approaches have shown promising results, utilizing techniques such as feature extraction, object tracking, and machine learning algorithms. For example, Zhang et al. (2012) proposed a real-time hand gesture recognition system based on depth data and skeleton tracking. Wachs et al. (2011) developed a computer vision-based method using a hierarchical template matching approach for robust hand gesture recognition. These studies demonstrate the effectiveness of computer vision techniques in recognizing hand gestures for subsequent conversion.

In 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 centred 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 colour pixels are gathered into blobs. Blobs are statistical objects based on the location (x, y) and the colorimetry (Y, U, V) of the skin color pixels in order to determine homogeneous areas.

In 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 colour. 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.

2.2 Problem Identification and Application:

The problem statement centres around the concept of a camera-based sign language recognition system for the deaf, which would transform sign language gestures to text and subsequently text to speech. Our goal is to create a user-friendly and straightforward solution.

Dumb individuals communicate via hand signs, thus normal folks have a hard time understanding what they're saying. As a result, systems that recognise various signs and deliver information to ordinary people are required.

By harnessing the capabilities of hand gesture recognition technology, a hand gesture to speech and text converter application empowers individuals with enhanced communication abilities, promotes accessibility, and facilitates intuitive and interactive interactions with various devices and systems

Assistive Technology:

The application can serve as an assistive technology tool for individuals with physical disabilities or limited mobility. People with conditions such as paralysis or motor impairments can use hand gestures to control and interact with devices, perform tasks, and communicate their needs effectively.

Multilingual Communication:

Hand gesture recognition can provide an alternative means of communication, especially in multicultural and multilingual settings. By using standardized hand gestures, individuals from different linguistic backgrounds can communicate and understand each other, breaking down language barriers.

Accessibility for People with Disabilities:

One of the primary applications of a hand gesture to speech and text converter is to enhance accessibility for individuals with disabilities, particularly those with speech or hearing impairments. The application enables them to communicate effectively by converting their hand gestures into speech and text output, allowing for more inclusive and independent communication.

Gesture-Based Control and Interaction:

In addition to communication, a hand gesture to speech and text converter can facilitate gesture-based control and interaction with devices and systems. Users can use hand gestures to navigate through menus, control multimedia playback, interact with virtual or augmented reality environments, or operate smart home devices.

Educational Tools:

The application can be utilized as an educational tool for teaching sign language or enhancing language learning. It can provide real-time feedback and guidance to learners, enabling them to practice and improve their hand gesture communication skills.

Gaming and Virtual Reality:

In the gaming and virtual reality domain, a hand gesture to speech and text converter can enable more immersive and interactive experiences. Players can use hand gestures to control characters, perform in-game actions, and communicate with other players, enhancing the overall gaming or virtual reality experience.

Presentation and Public Speaking:

Public speakers, lecturers, or presenters can benefit from using hand gesture to speech and text converter applications to enhance their presentations. They can use hand gestures to control slides, trigger multimedia content, or highlight key points, while the application converts their gestures into speech, assisting them in delivering their message effectively.

2.3 System Requirement Analysis

The SRS is a specification for a specific software product, program, or set of applications that perform particular functions in a specific environment.

2.3.1 Software Requirements

- **VS Code**
  Visual Studio Code is a streamlined code editor with support for development operations like debugging, task running, and version control. It aims to provide just the tools a developer needs for a quick code-build-debug cycle and leaves more complexworkflows to fuller featured IDEs, such as Visual Studio IDE.

- **Python**

  Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. Its high-level built-in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together.

2.3.2 Module Requirements

- **Numpy** :-

  NumPy is a Python library used for working with arrays. It also has functions for working in domain of linear algebra, fourier transform, and matrices.
- **Open Cv :-**

  OpenCV is a cross-platform library using which we can develop real-time computer vision applications. It mainly focuses on image processing, video capture and analysis including features like face detection and object detection.
- **Pillow:-**

  Python Imaging Library (expansion of PIL) is the de facto image processing package for Python language. It incorporates lightweight image processing tools that aids in editing, creating and saving images.
- **Tensorflow :-**

  TensorFlow is an open source framework developed by Google researchers to run machine learning, deep learning and other statistical and predictive analytics workloads.
- **Mediapipe :-**

  MediaPipe is an open-source framework for building pipelines to perform computer vision inference over arbitrary sensory data such as video or audio.
- **Keras :-**

  Keras is an Open Source Neural Network library written in Python that runs on top of Tensorflow.

2.3.3 Hardware Requirement

- Processors: i3,i5,i7 processor
- Disk space: 1 GB or more

- Operating systems: Windows 7 or later, macOS, and Linux

2.4 Keywords and Definitions:

2.4.1 Feature Extraction and Representation:

The representation of an image as a 3D matrix having dimension as of height and width of the image and the value of each pixel as depth (1 in case of Grayscale and 3 in case of RGB). Further, these pixel values are used for extracting useful features using CNN.

2.4.2 Artificial Neural Network (ANN):

Artificial Neural Network is a connection of neurons, replicating the structure of human brain. Each connection of neuron transfers information to another neuron. Inputs are fed into first layer of neurons which processes it and transfers to another layer of neurons called as hidden layers. After processing of information through multiple layers of hidden layers, information is passed to final output layer.

To understand the concept of the architecture of an artificial neural network, we have to understand what a neural network consists of. In order to define a neural network that consists of a large number of artificial neurons, which are termed units arranged in a sequence of layers. Lets us look at various types of layers available in an artificial neural network.

Input Layer:

As the name suggests, it accepts inputs in several different formats provided by the programmer.

Hidden Layer:

The hidden layer presents in-between input and output layers. It performs all the calculations to find hidden features and patterns.

Output Layer:

The input goes through a series of transformations using the hidden layer, which finally results in output that is conveyed using this layer. The artificial neural network takes input and computes the weighted sum of the inputs and includes a bias.

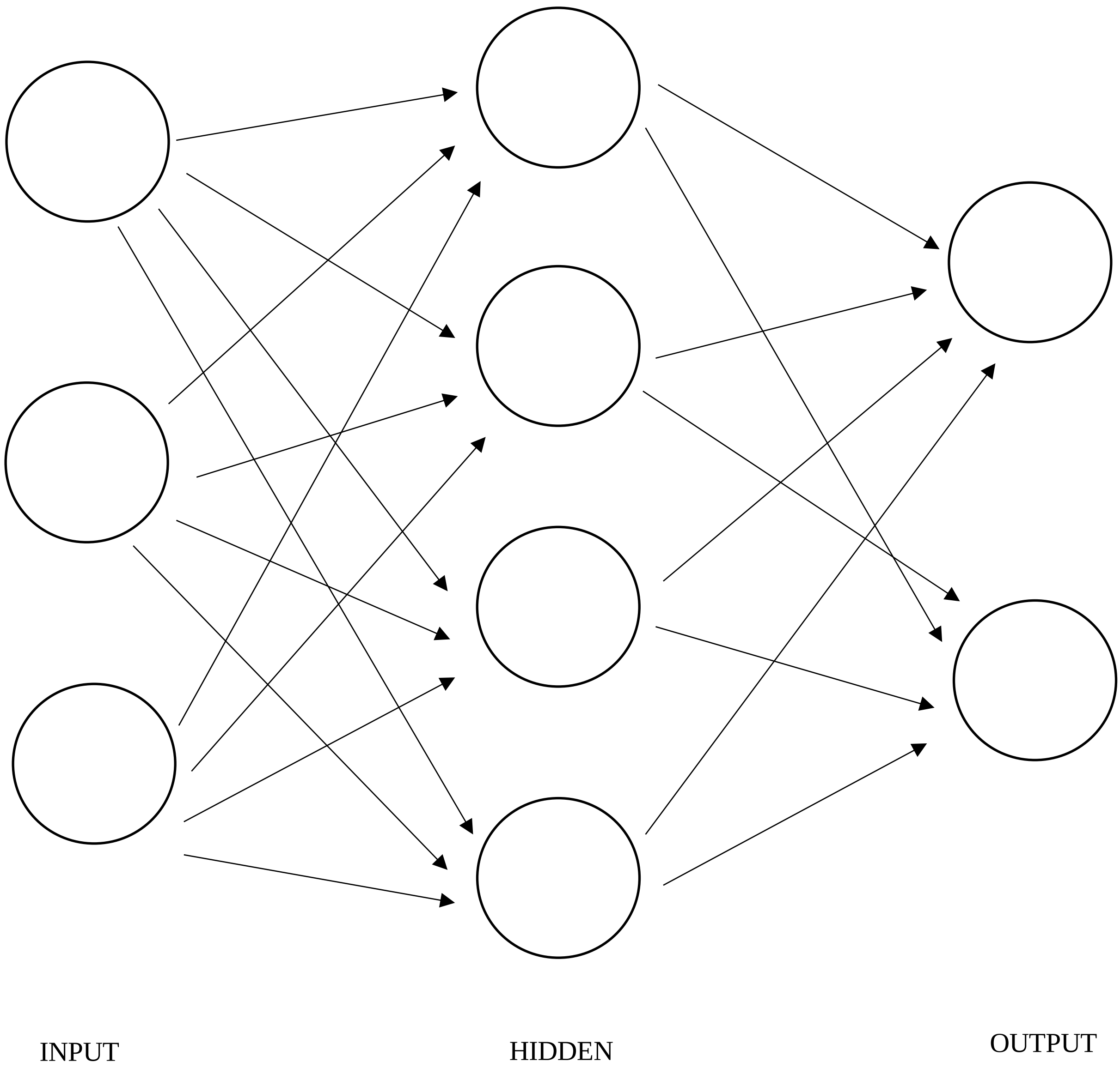


Fig. 2.4.2 Artificial Neural Network

These are capable of learning and have to be trained. There are different learning strategies:

1. Unsupervised Learning
2. Supervised Learning
3. Reinforcement Learning

2.4.3 Convolutional Neural Network (CNN):

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.

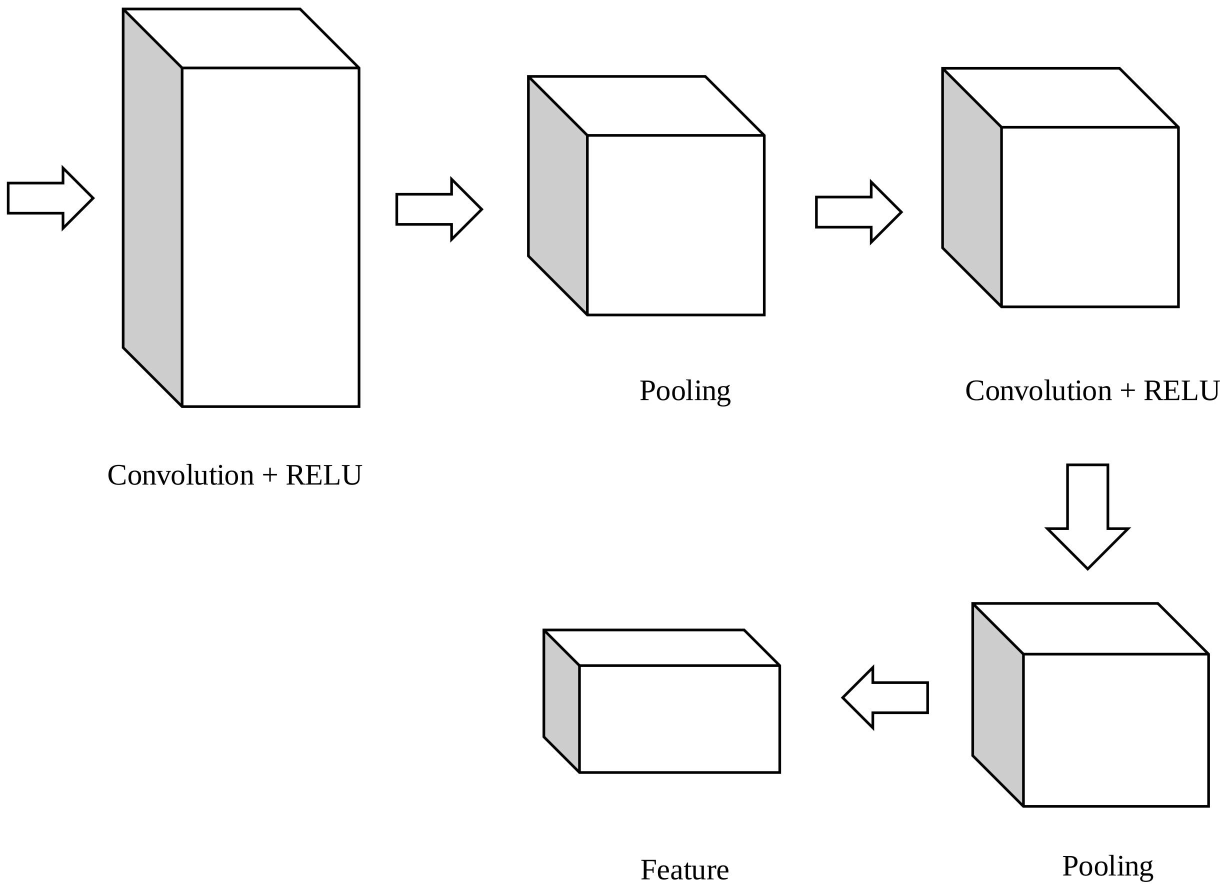


Fig. 2.4.3 Block diagram of CNN

2.4.3.1 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 we will 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 colour.

We use pooling layer to decrease the size of activation matrix and ultimately reduce the learnable parameters. There are two types 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 an activation matrix half of its original Size.
- **b. Average Pooling:** In average pooling, we take advantage of of all Values in a window.

12	20	30	0
8	12	2	0
34	70	37	4
112	100	25	12

20	30
112	37

13	8
79	20

Fig. 2.4.3.1 Convolution Layer

2.4.3.2 Fully Connected Layer:

In convolution layer, neurons are connected only to a local region, while in a fully connected region, we will connect all the inputs to neurons.

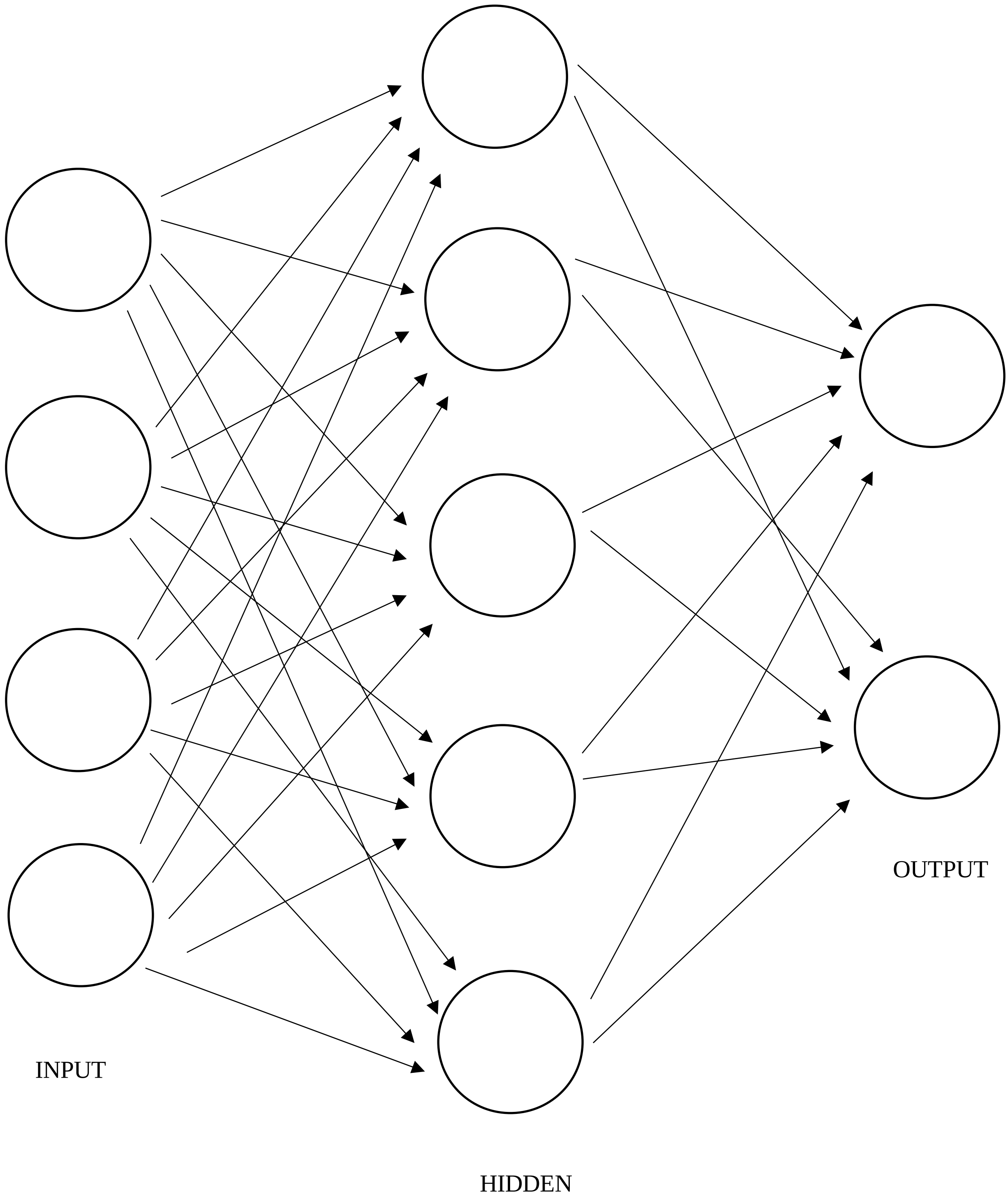


Fig. 2.4.3.2 Fully Connected Layer

2.4.3.3 Final Output Layer:

After getting values from fully connected layer, we will connect them to the 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.

2.5 Data Flow Diagram

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It can be manual, automated, or a combination of both. It shows how data enters and leaves the system, what changes the information, and where data is stored.

The objective of a DFD is to show the scope and boundaries of a system as a whole. It may be used as a communication tool between a system analyst and any person who plays a part in the order that acts as a starting point for redesigning a system. The DFD is also called as a data flow graph or bubble chart.

The Data Flow Diagram has 4 components:

- <u>Process</u>:
  Input to output transformation in a system takes place because of process function. The symbols of a process are rectangular with rounded corners, oval, rectangle or a circle.
- <u>Data Flow:</u>
  Data flow describes the information transferring between different parts of the systems. The arrow symbol is the symbol of data flow. A relatable name should be given to the flow to determine the information which is being moved. Data flow also represents material along with information that is being moved.
- <u>Warehouse:</u>
  The data is stored in the warehouse for later use. Two horizontal lines represent the symbol of the store. The warehouse is simply not restricted to being a data file rather it can be anything like a folder with documents, an optical disc, a filing cabinet.
- <u>Terminator:</u>
  The Terminator is an external entity that stands outside of the system and communicates with the system.

A data flow diagram can dive into progressively more detail by using levels and layers, zeroing in on a particular piece. DFD levels are numbered 0, 1 or 2, and occasionally go to even Level 3 or beyond. The necessary level of detail depends on the scope of what you are trying to accomplish.

- **DFD level** 0 is also known as fundamental system model, or context diagram represents the entire software requirement as a single bubble with input and output data denoted by incoming and outgoing arrows. Then the system is decomposed and described as a DFD with multiple bubbles. Parts of the system represented by each of these bubbles are then decomposed and documented as more and more detailed DFDs. This process may be repeated at as many levels as necessary until the program at hand is well understood. It is essential to preserve the number of inputs and outputsbetween levels, this concept is called leveling by DeMacro. Thus, if bubble "A" has two inputs x 1 and x2 and one output y, then the expanded DFD, that represents "A" should have exactly two external inputs and one external output.
- **DFD Level** 1 provides a more detailed breakout of pieces of the Context Level Diagram. You will highlight the main functions carried out by the system, as you break down the high-level process of the Context Diagram into its subprocesses.
- **DFD Level** 2 then goes one step deeper into parts of Level 1. It may require more text to reach the necessary level of detail about the system's functioning.

The following observations about DFDs are essential:

- All names should be unique. This makes it easier to refer to elements in the DFD.
- Remember that DFD is not a flow chart. Arrows is a flow chart that represents the order of events; arrows in DFD represents flowing data. A DFD does not involve any order of events.
- Suppress logical decisions. If we ever have the urge to draw a diamond-shaped box in a DFD, suppress that urge! A diamond-shaped box is used in flow charts to represents decision points with multiple exists paths of which the only one is taken. This implies an ordering of events, which makes no sense in a DFD.

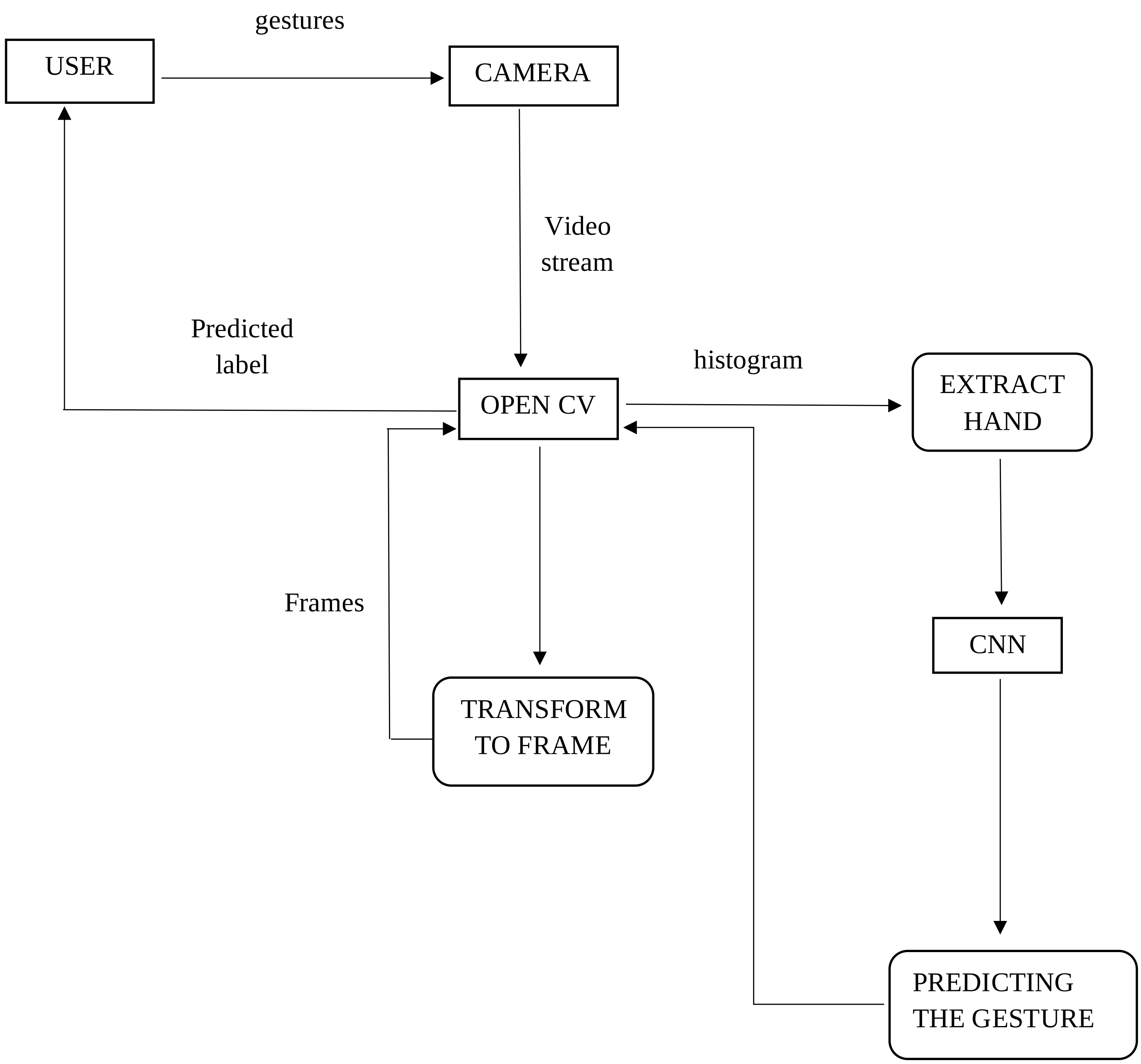


Fig. 2.5 Data Flow Diagram

CHAPTER-III

METHODOLOGY

3.1 Workflow Diagram

A workflow diagram (also known as a workflow) provides a graphic overview of the business process. Using standardized symbols and shapes, the workflow shows step by step how your work is completed from start to finish. It also shows who is responsible for work at what point in the process.

Designing a workflow involves first conducting a thorough workflow analysis, which can expose potential weaknesses. A workflow analysis can help you define, standardize and identify critical areas of your process.

A workflow chart is commonly used for documentation and implementation purposes since it provides a general overview of a business process. It's often the foundation for other documentation including flowcharts, data flow diagram, projects and more.

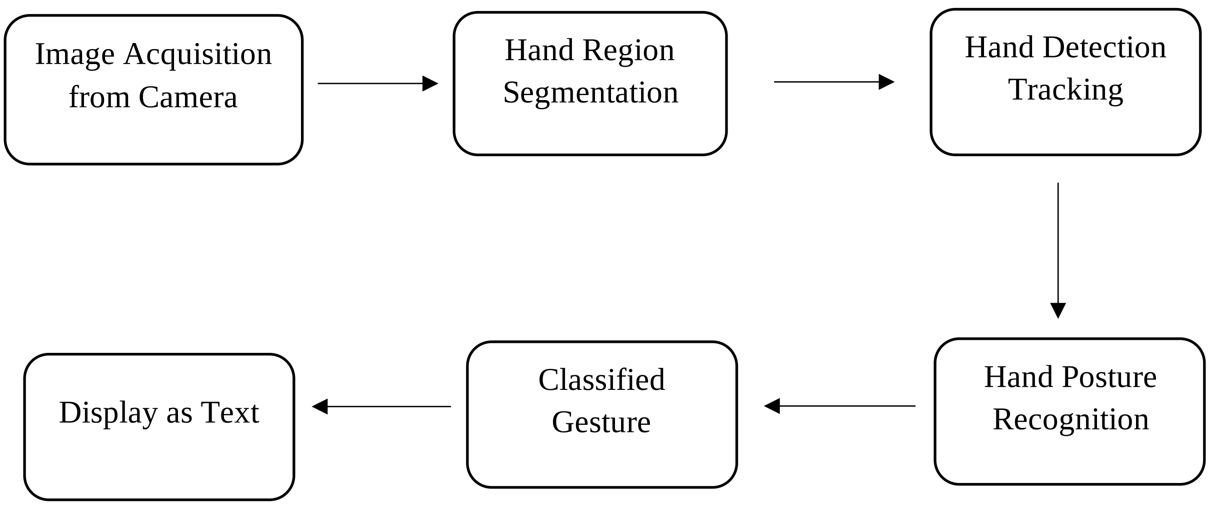


Fig. 3.1 Workflow Diagram

3.2 Data Set Generation:

For the project we tried to find already made datasets but we couldn't find dataset in the form of raw images that matched our requirements. All we could find were the datasets in the form of RGB values. Hence, we decided to create our own data set. Steps we followed to create our data set are as follows.

We used Open computer vision (OpenCV) library in order to produce our dataset.

Firstly, we captured around 800 images of each of the symbol in ASL (American Sign Language) for training purposes and around 200 images per symbol for testing purpose.

Then, we apply Gaussian Blur Filter to our image which helps us extract various features of our image. The image, after applying Gaussian Blur, looks as follows:

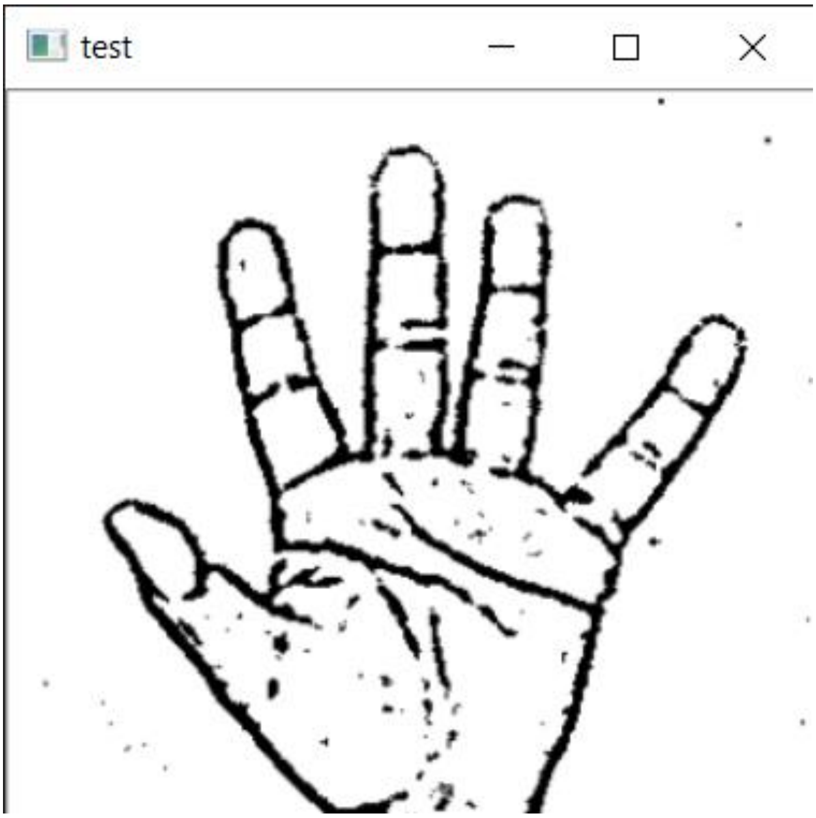


Fig. 3.2 Data Set Generation

3.3 Gesture Classification:

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

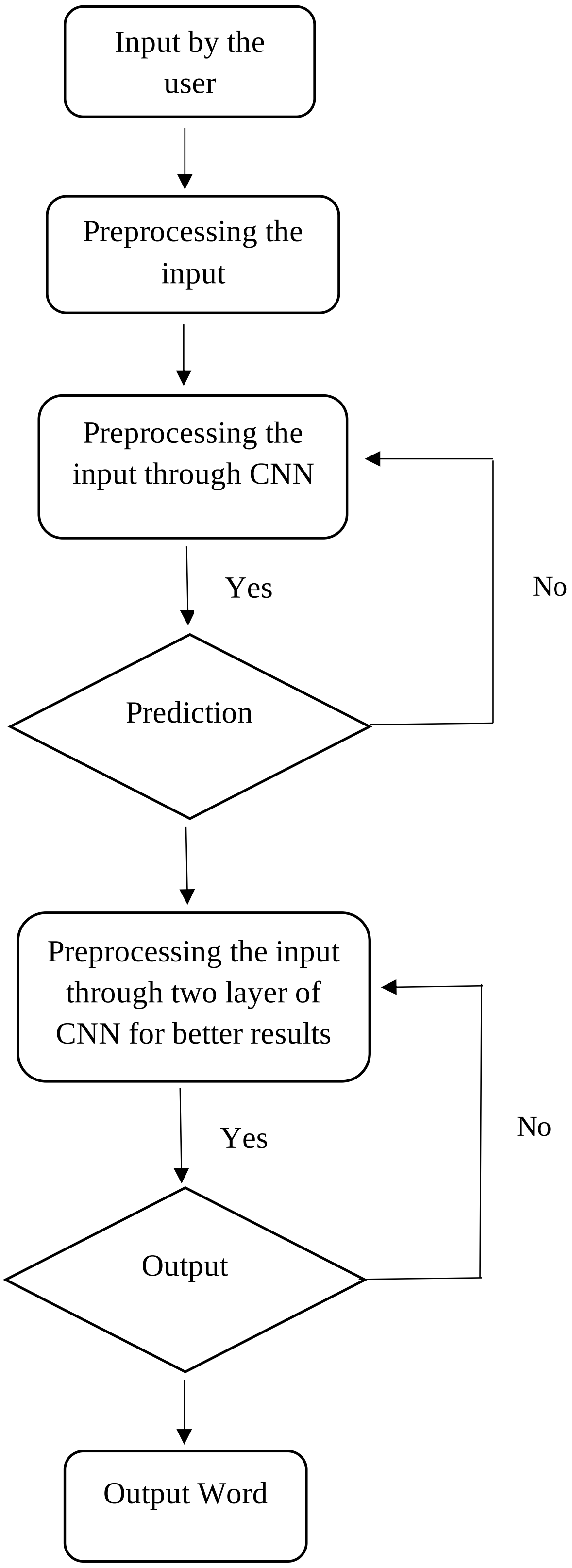


Fig. 3.3 Gesture Classification

Algorithm Layer 1:

- Apply Gaussian Blur filter and threshold to the frame taken with openCV to get the processed image after feature extraction.
- 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.
- Space between the words is considered using the blank symbol.

Algorithm Layer 2:

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

3.3.1 Layer 1:

CNN Model:

- **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-weights.
- **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.
- **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.
- **2nd Pooling Layer:** The resulting images are down sampled again using max pool of 2x2 and is reduced to 30 x 30 resolution of images.
- **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.
- **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.

- **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 problemand 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 optimizer combines the advantages of two extensions of two stochastic gradient descent algorithms namely adaptive gradient algorithm (ADA GRAD) and root mean square propagation (RMSProp).

3.3.2 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:

- For D : R and U
- For U : D and R
- For I : T, D, K and I
- For S : M and N

So, to handle above cases we made three different classifiers for classifying these sets:

- D, R, U}
- {T, K, D, I}
- {S, M, N}

3.4 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.

In other case it predicts the end of word by printing a space and the current gets appended to the sentence below.

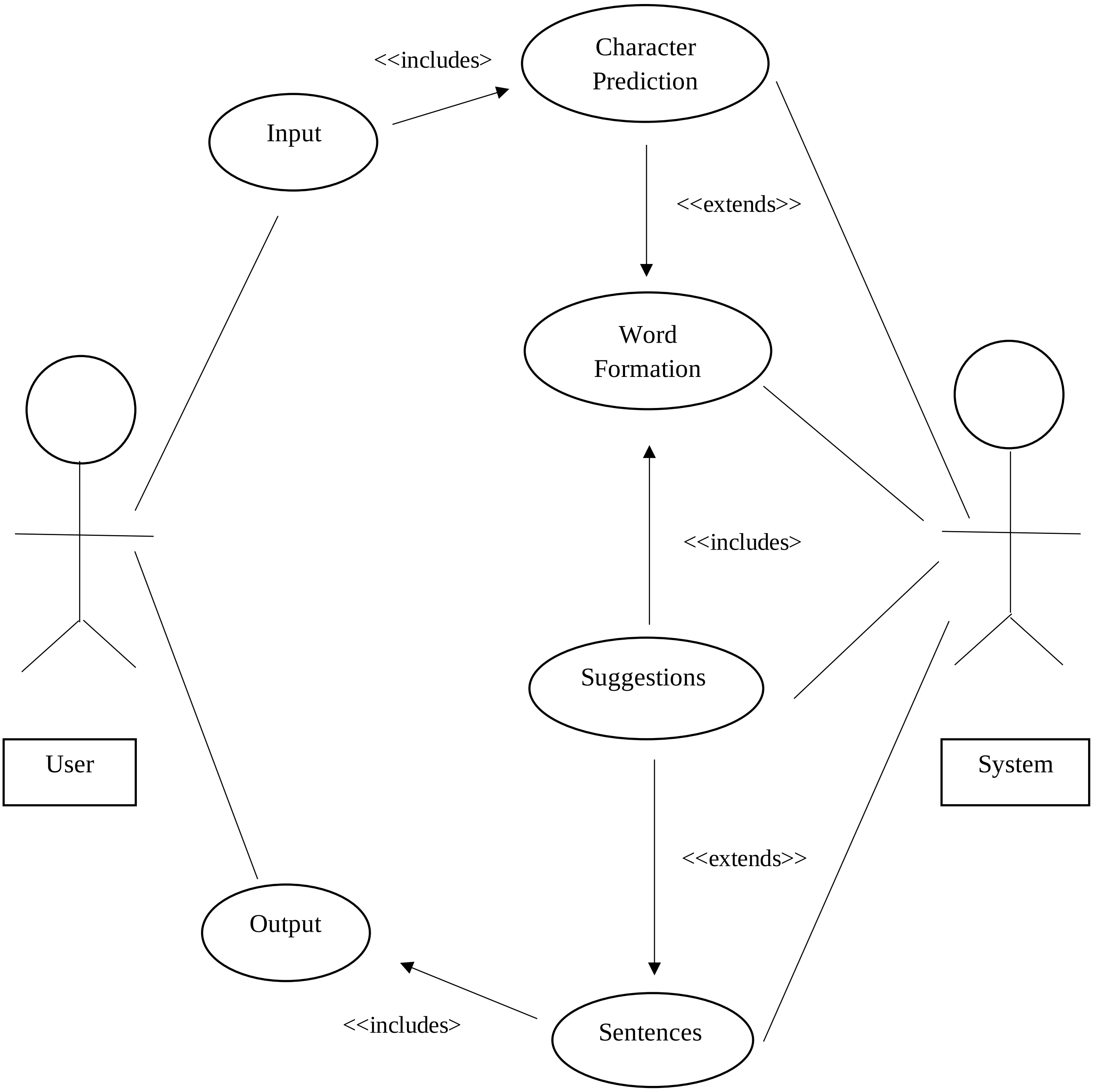


Fig. 3.4 Sentence Formation Implementation

3.5 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.

3.6 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 pre-processing 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 value 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 labelled 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 labelled value and is zero exactly when it is equal to the labelled 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.

3.7 E-R Diagram

ERD stands for entity relationship diagram. People also call these types of diagrams ER diagrams and Entity Relationship Models. An ERD visualizes the relationships between entities like people, things, or concepts in a database. An ERD will also often visualize the attributes of these entities.

By defining the entities, their attributes, and showing the relationships between them, an ER diagram can illustrate the logical structure of databases. This is useful for engineers hoping to either document a database as it exists or sketch out a design of a new database.

An ER diagram can help businesses document existing databases and thereby troubleshoot logic or deployment problems or spot inefficiencies and help improve processes when a business wants to undertake business process re-engineering. ERDs can also be used to design and model new databases and make sure that engineers can identify any logic or designflaws before they're implemented in production.

- Document an existing database structure
- Debug, troubleshoot, and analyze
- Design a new database
- Gather design requirements
- Business process re-engineering **(BPR)**

When documenting a system or process, looking at the system in multiple ways increases the understanding of that system. ERD diagrams are commonly used in conjunction with a data flow diagram to display the contents of a data store. They help us to visualize how data is connected in a general way, and are particularly useful for constructing a relational database.

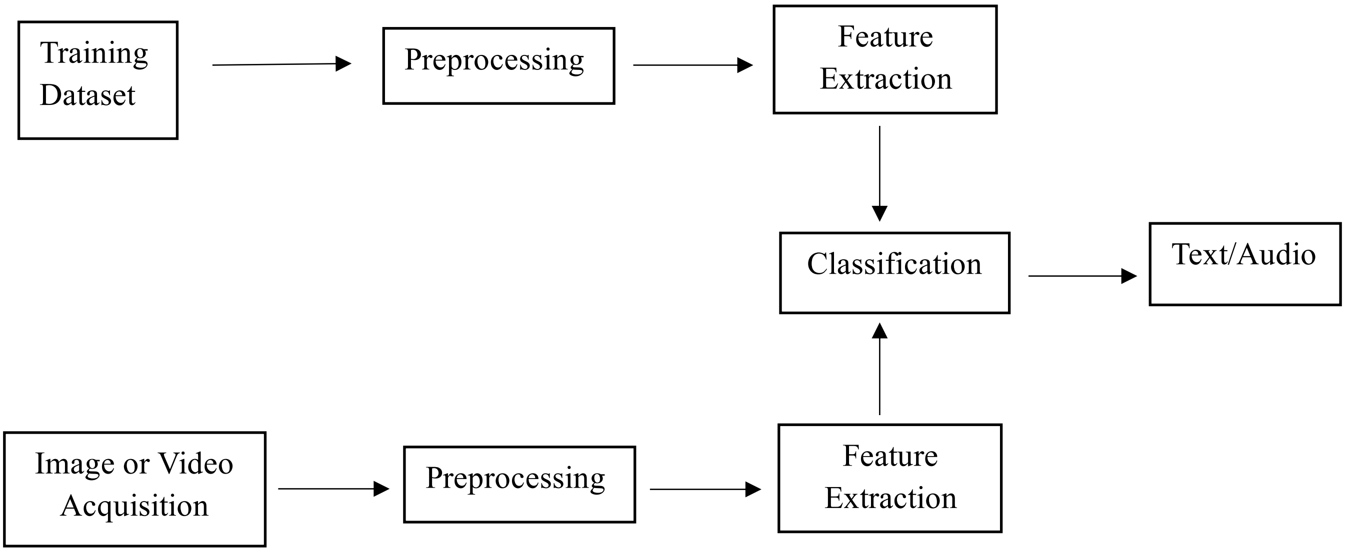


Fig. 3.7 E-R Diagram

3.8 Use Case Diagram

In the Unified Modeling Language (UML), a use case diagram can summarize the details of your system's users (also known as actors) and their interactions with the system. To build one, you'll use a set of specialized symbols and connectors. A use case diagram doesn't go into a lot of detail—for example, don't expect it to model the order in which steps are performed. Instead, a proper use case diagram depicts a high-level overview of the relationship between use cases, actors, and systems. Experts recommend that use case diagrams be used to supplement a more descriptive textual use case.

UML is the modeling toolkit that you can use to build your diagrams. Use cases are represented with a labeled oval shape. Stick figures represent actors in the process, and the actor's participation in the system is modeled with a line between the actor and use case. To depict the system boundary, draw a box around the use case itself.

The actors are on the outside of the system's border, whilst the use cases are on the inside. The behaviour of the system as viewed through the eyes of the actor is described in a use case. It explains the system's role as a series of events that result in a visible consequence for the actor. Use Case Diagrams: What Are They Good For? The objective of a use case diagram is to capture a system's dynamic nature.. However, this definition is too generic to describe the purpose, as other four diagrams (activity, sequence, collaboration, and State chart) also have the same purpose. We will look into some specific purpose, which will distinguish it from other four diagrams.

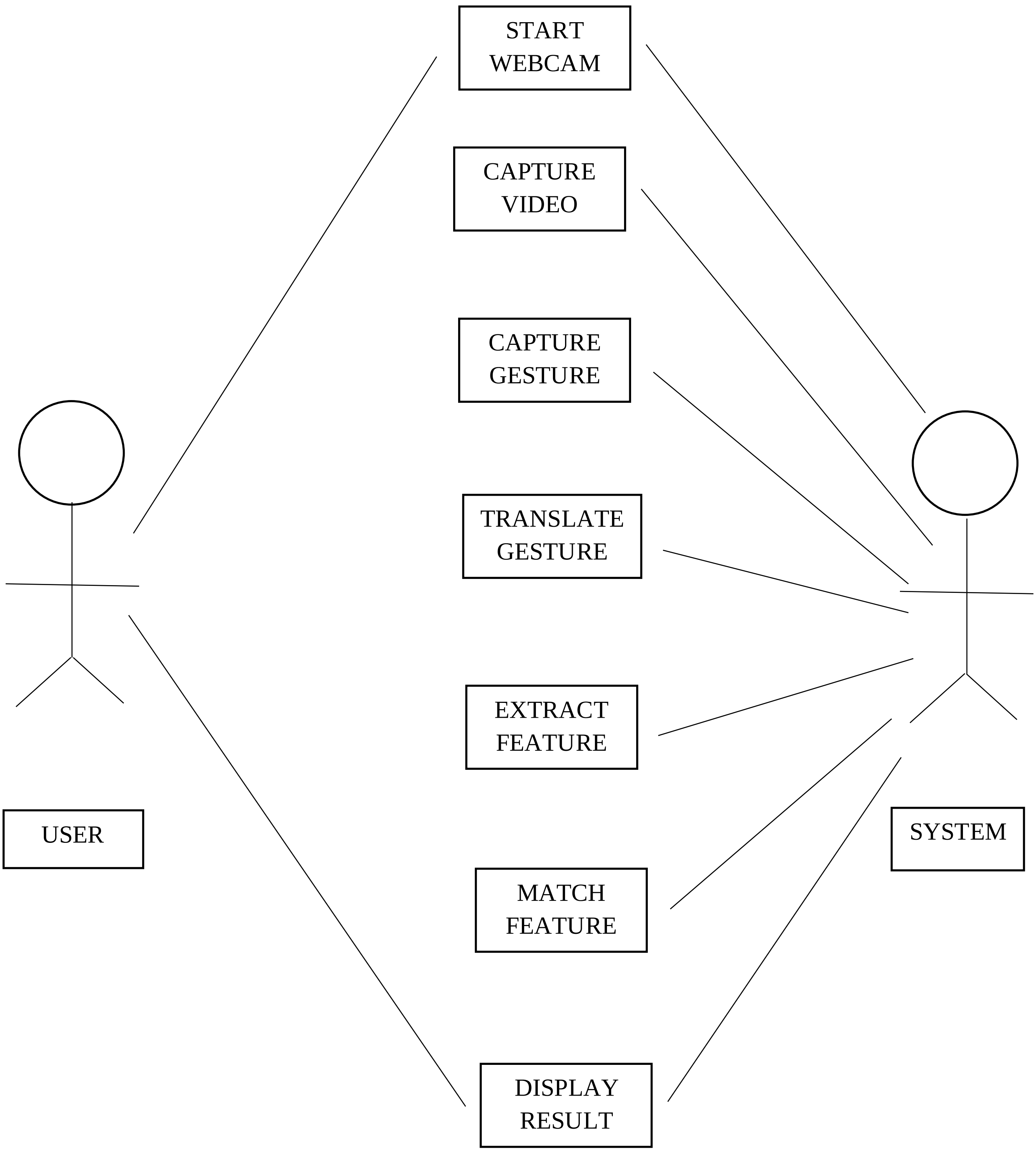


Fig. 3.8 Use Case Diagram

3.9 Sequence & Activity Diagram

A sequence diagram is a type of interaction diagram because it describes how and in what order group of objects works together. These diagrams are used by software developers and business professionals to understand requirements for a new system or to document an existing process. Sequence diagrams are sometimes known as event diagrams or event scenarios.

Sequence diagrams can be useful references for businesses and other organizations. Try drawing a sequence diagram to:

- Represent the details of a UML use case.
- Model the logic of a sophisticated procedure, function, or operation.
- See how objects and components interact with each other to complete a process.
- Plan and understand the detailed functionality of an existing or future scenario

The following scenarios are ideal for using a sequence diagram:

- Usage scenario: A usage scenario is a diagram of how your system could potentially be used. It's a great way to make sure that you have worked through the logic of every usage scenario for the system.
- Method logic: Just as you might use a UML sequence diagram to explore the logic of a use case, you can use it to explore the logic of any function, procedure, or complex process.
- Service logic: If you consider a service to be a high-level method used by different clients, a sequence diagram is an ideal way to map that out.

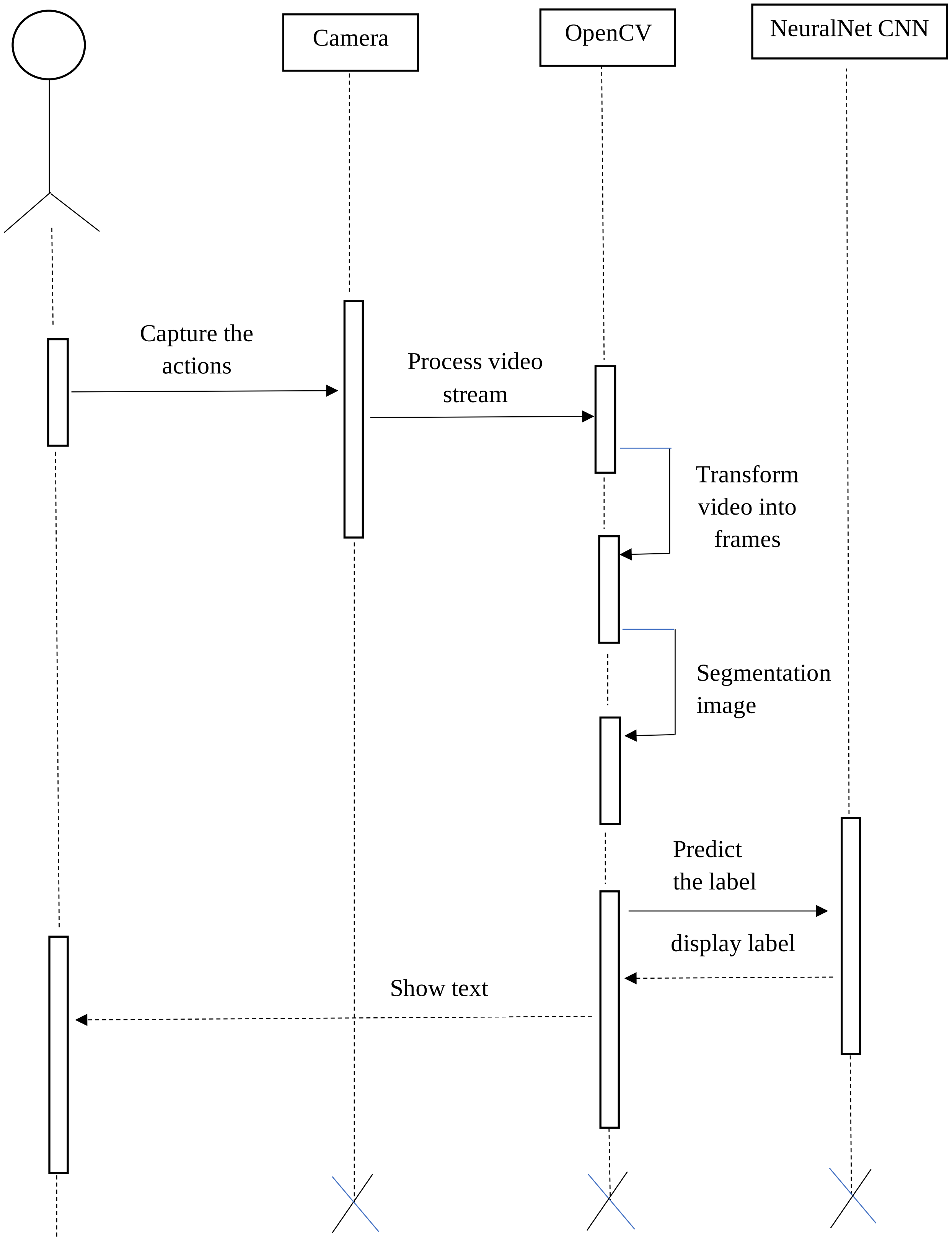


Fig. 3.9 Sequence & Activity Diagram

3.10 Collaboration & Class Diagram

A collaboration diagram, also known as a communication diagram, is an illustration of the relationships and interactions among software objects in the Unified Modeling Language (UML). Developers can use these diagrams to portray the dynamic behavior of a particular use case and define the role of each object.

To create a collaboration diagram, first identify the structural elements required to carry out the functionality of an interaction. Then build a model using the relationships between those elements. Several vendors offer software for creating and editing collaboration diagrams.

Notations of a collaboration diagram

A collaboration diagram resembles a flowchart that portrays the roles, functionality and behavior of individual objects as well as the overall operation of the system in real time. The four major components of a collaboration diagram include the following:

1. **Objects.** These are shown as rectangles with naming labels inside. The naming label follows the convention of object name: class name. If an object has a property or state that specifically influences the collaboration, this should also be noted.
2. **Actors.** These are instances that invoke the interaction in the diagram. Each actor has a name and a role, with one actor initiating the entire use case.
3. **Links.** These connect objects with actors and are depicted using a solid line between two elements. Each link is an instance where messages can be sent.
4. **Messages between objects**. These are shown as a labeled arrow placed near a link. These messages are communications between objects that convey information about the activity and can include the sequence number.

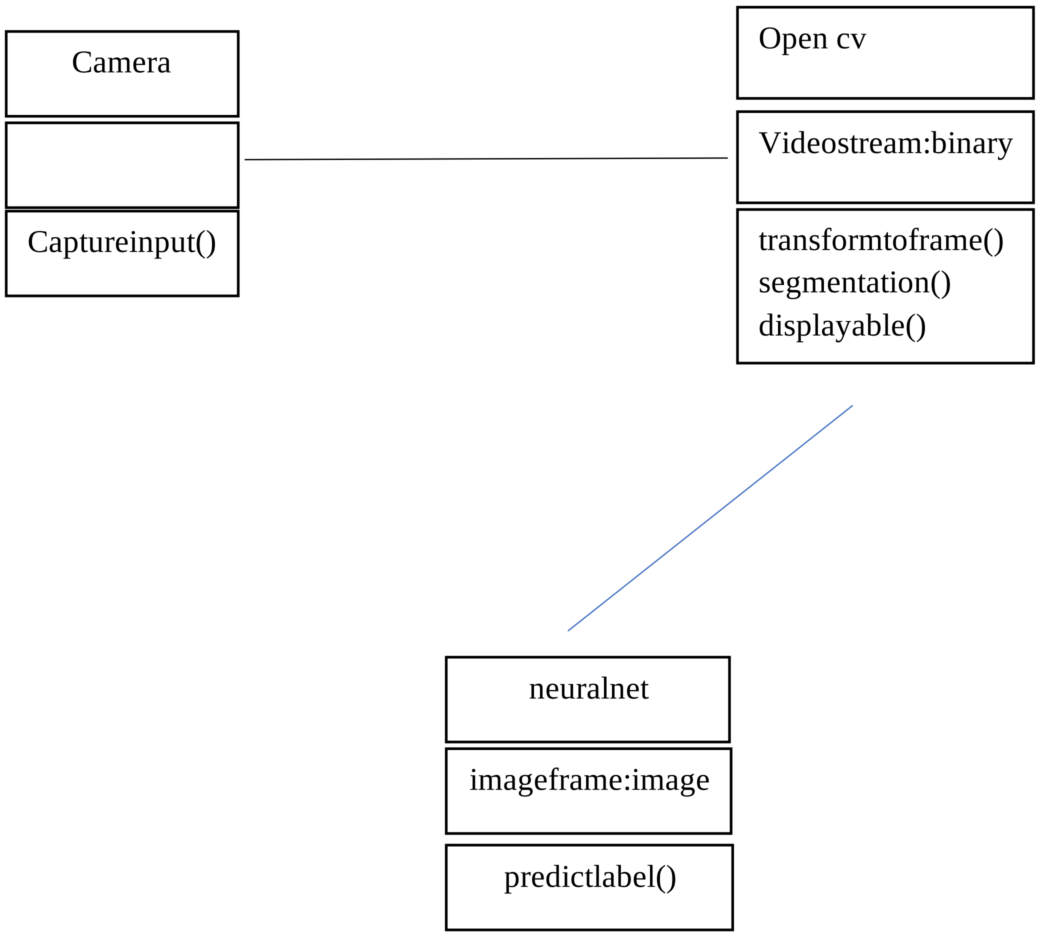


Fig. 3.10 Collaboration & Class Diagram

CHAPTER-IV

RESULT

4. Snapshots

The interface aims to provide users with a seamless and engaging experience in converting hand gestures to speech and text. The Home screen serves as the central hub, allowing users to perform gestures, view the converted output, and access additional information through the About Us section. The Exit option ensures a smooth exit process, maintaining a user-friendly interface flow.

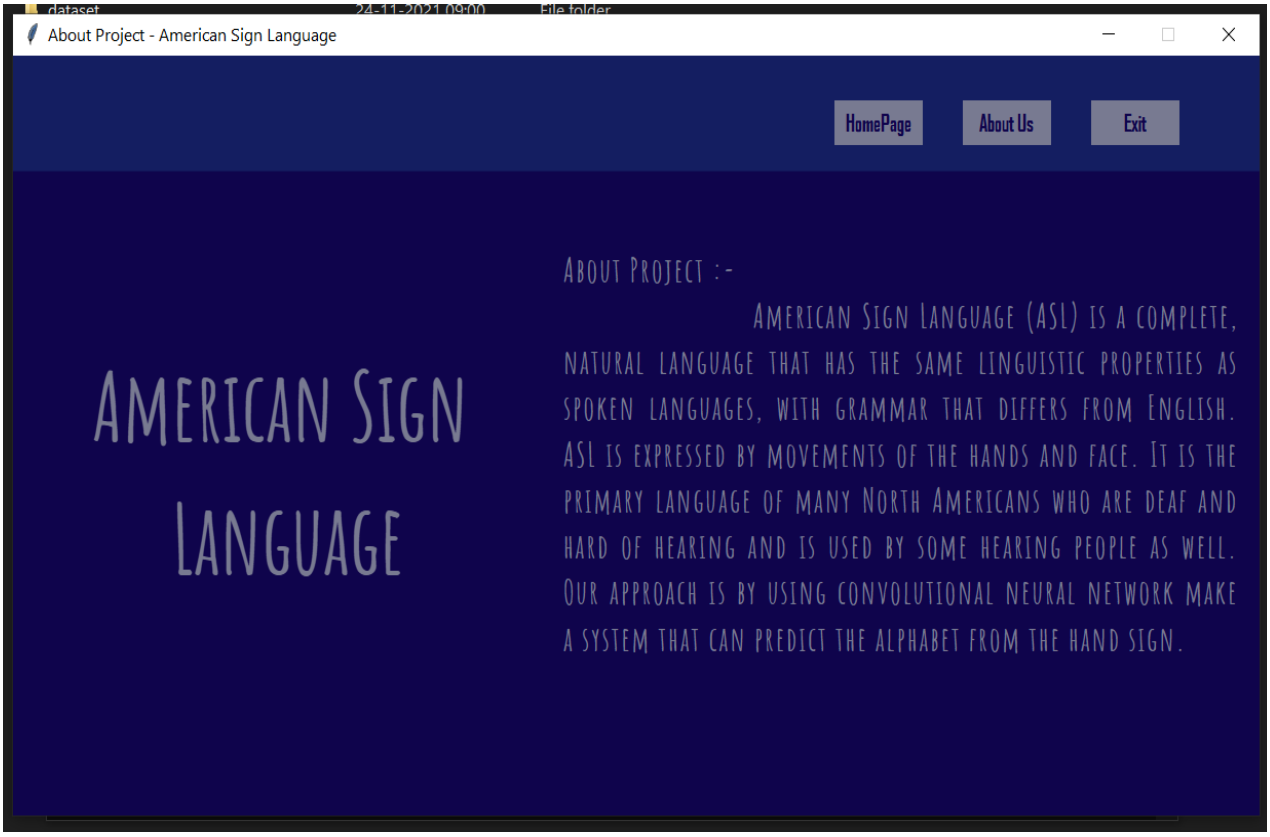


Fig. 4.1 Application Interface (1)

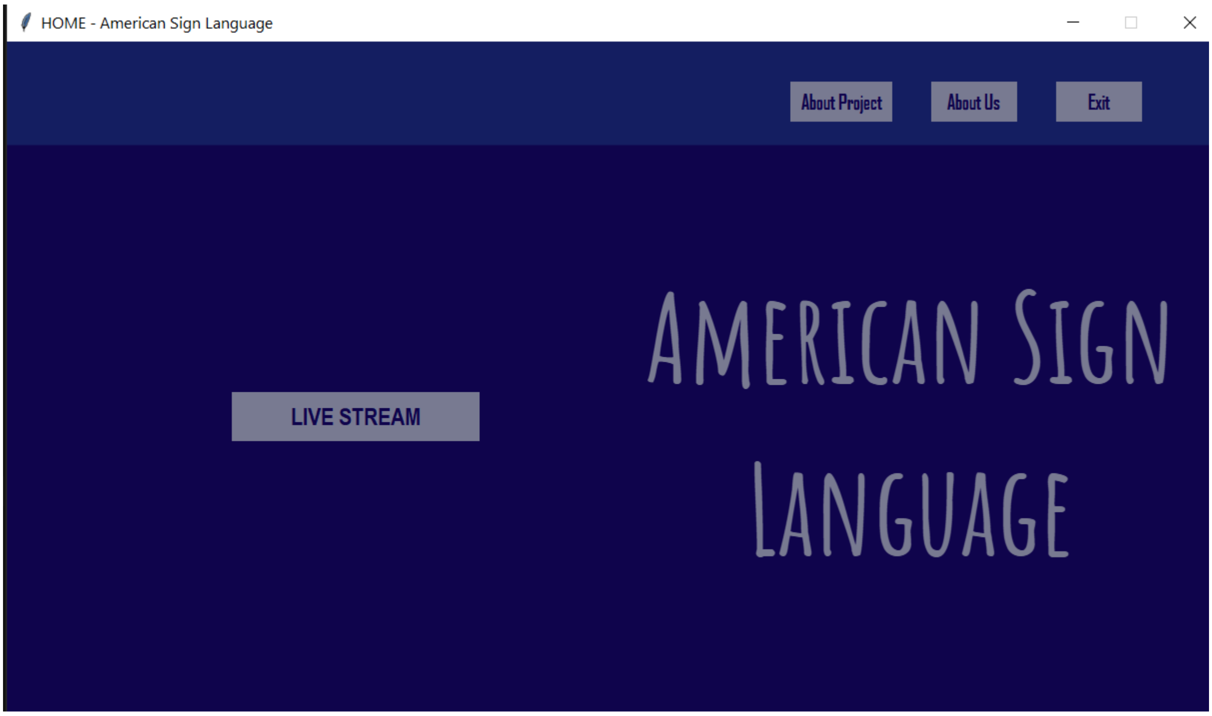


Fig.4.2 Application Interface (2)

Home Screen:

The Home screen serves as the main landing page of the interface. It provides users with an overview of the available functionalities and options. The screen typically includes the following elements:

- **Gesture Recognition Area:** This is the designated area where users can perform hand gestures. It may be represented by a video feed or a graphical representation of the user's hand.
- **Speech and Text Output Display:** This section displays the converted speech and text output generated from the recognized hand gestures. It may consist of a text box or a dynamic speech synthesis output.
- **Instructional Prompts:** The Home screen may include instructional prompts or guidelines to assist users in performing the correct hand gestures. These prompts can be displayed as text instructions or visual cues.

About Us:

The About Us section provides information about the application, its purpose, and the development team. A brief overview of the hand gesture to speech and text converter application, highlighting its goals and functionality.

A list of key features and capabilities of the interface, such as real-time gesture recognition, accurate speech synthesis, and text conversion.

Exit Option:

The Exit option allows users to exit the interface and close the application. It provides a convenient way to end the session and return to the device's home screen or another application.

- **Exit Button:** A clearly labeled button or option that users can click or select to exit the interface.
- **Confirmation Prompt:** In some cases, a confirmation prompt may appear to ensure that the user intends to exit the application. This helps prevent accidental exits.
- **Graceful Closure:** The application should handle the exit process gracefully, ensuring that any ongoing processes or resources are properly terminated or saved before closing.

Live Stream Display:

The interface prominently displays a live video stream area where users can see their hand movements in real-time. This display provides immediate visual feedback, allowing users to adjust their gestures and ensure accurate recognition.

The following figure will show the recognizing of letters -

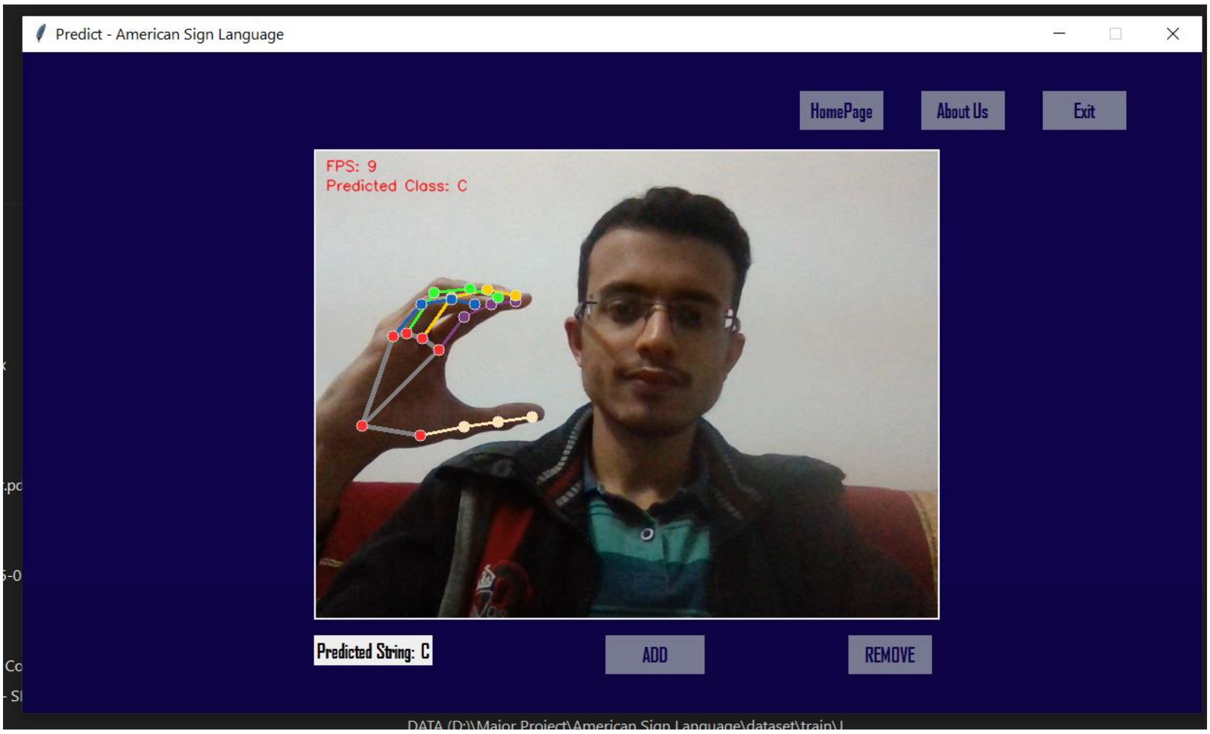
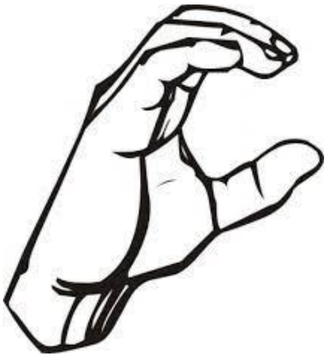


Fig.4.3 Classifying letter "C"



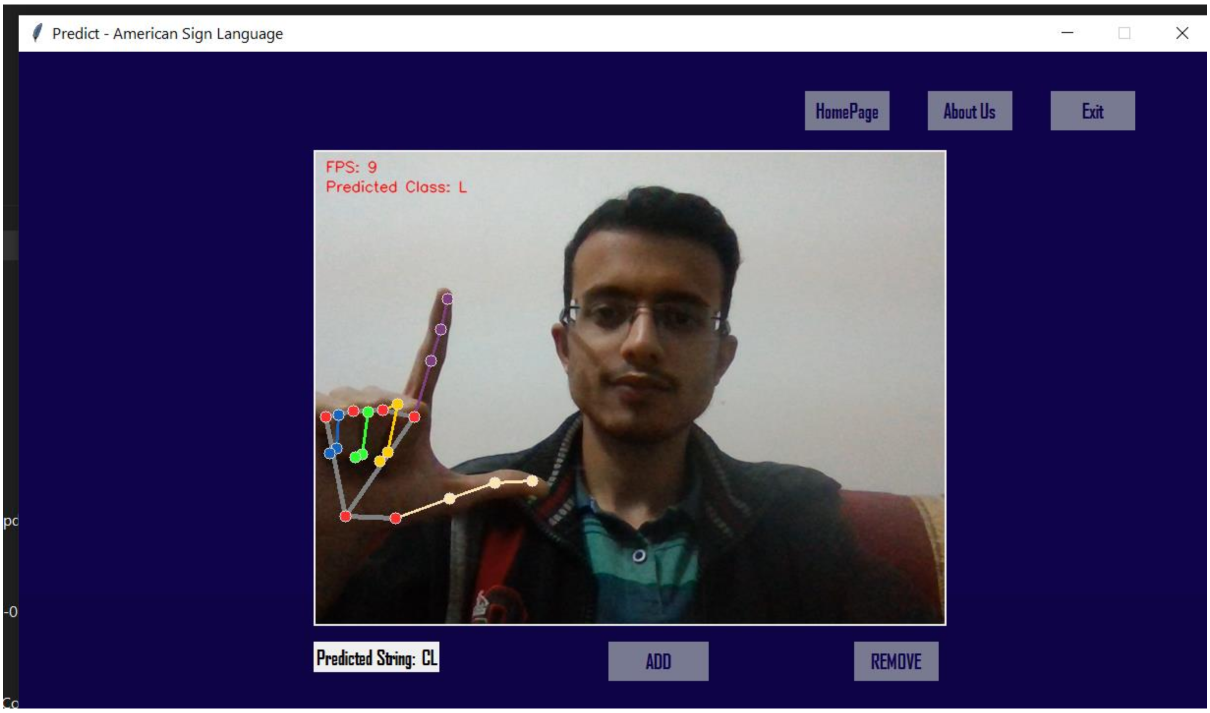


Fig.4.4 Classifying letter “L”

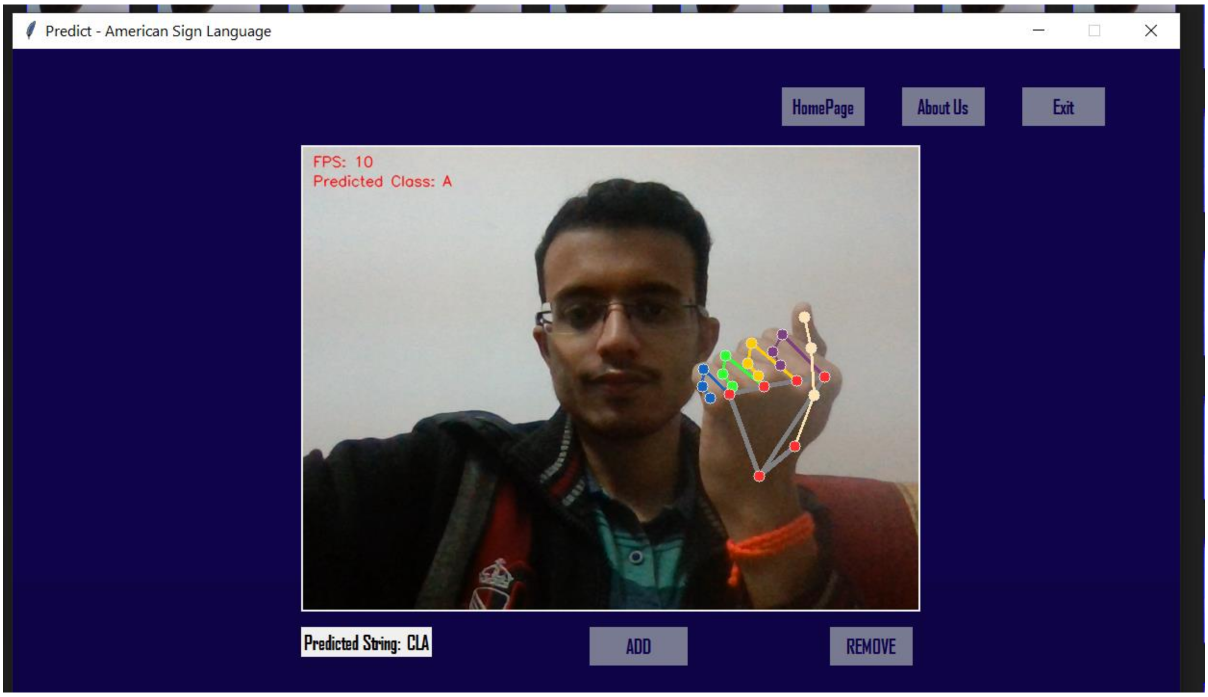


Fig.4.5 Classifying letter "A"

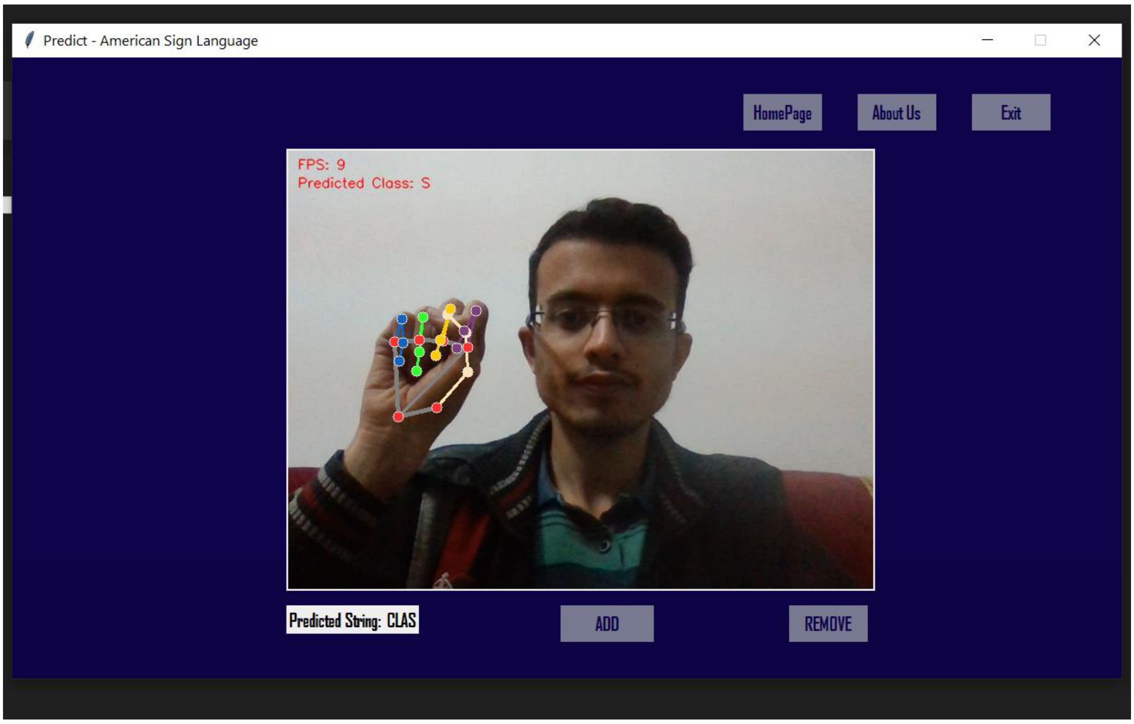


Fig.4.6 Classifying letter “S”

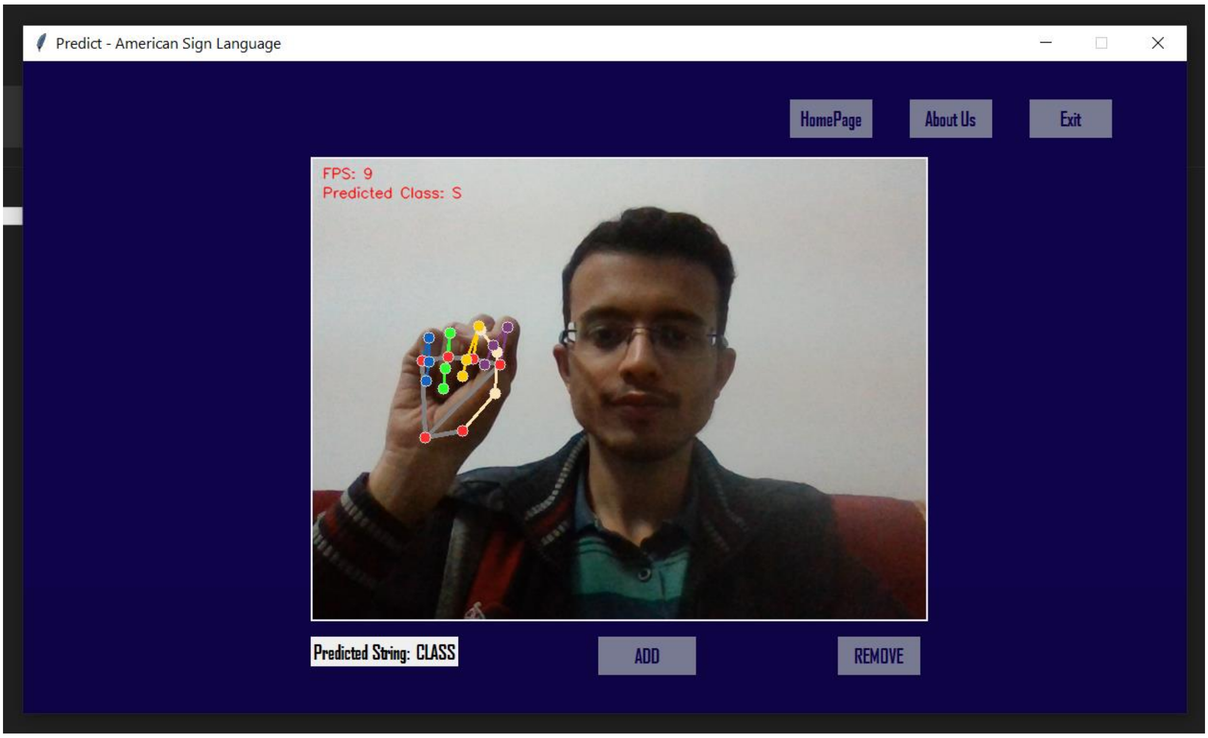


Fig.4.7 Classifying letter "S"

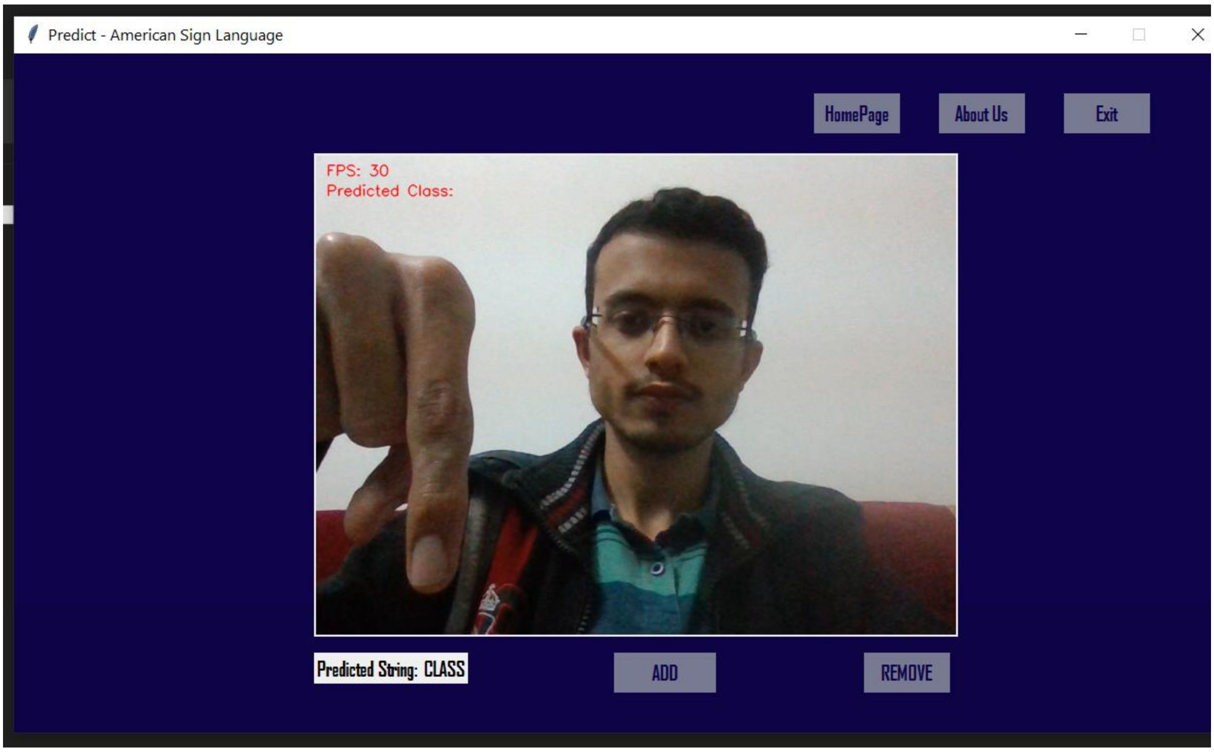


Fig.4.8 Display the text “CLASS”

4.2 Challenges Faced :

Gesture Recognition Accuracy:

Accurately detecting and recognizing hand gestures in real-time can be challenging. Factors such as variations in lighting conditions, background clutter, and occlusions (e.g., hands partially hidden or obstructed) can impact the accuracy of gesture recognition. Robust algorithms and techniques, such as computer vision and machine learning, are employed to improve recognition accuracy.

Gesture Variability and Complexity:

Hand gestures can vary significantly across individuals, cultures, and contexts. Different people may have distinct ways of expressing the same gesture. Additionally, gestures can be complex and involve intricate movements, making their recognition and interpretation more challenging. Developing a gesture recognition system that can handle a wide range of gestures, including subtle and complex movements, is crucial for accurate conversion.

Noise and Interference:

External noise sources, such as background conversations, environmental sounds, or sensor inaccuracies, can introduce interference and affect the recognition process. Noise reduction techniques, advanced filtering algorithms, and sensor calibration are employed to minimize the impact of noise and improve the robustness of gesture recognition.

Real-Time Processing:

To provide an interactive user experience, hand gestures need to be processed, converted into speech and text, and displayed in real-time. Achieving real-time processing requires efficient algorithms and optimized computational resources to handle the computational load.

Ambiguity in Gesture Interpretation:

Some hand gestures may have multiple interpretations or meanings depending on the context. Resolving ambiguities and accurately mapping gestures to their intended meanings pose a challenge. Context-awareness, user context analysis, and incorporating additional cues (e.g., facial expressions, body posture) can help disambiguate gestures and improve interpretation accuracy.

User Interface Design:

Designing an intuitive and user-friendly interface is crucial for effective communication and user engagement. The interface should clearly convey the gesture recognition process, provide visual feedback of recognized gestures, and offer intuitive controls for interaction. The design should be accessible, accommodate different user preferences.

CHAPTER -V

CONCLUSION

5. Conclusion

The practical adaption of the interface solution for visually impaired and blind people is limited by simplicity and usability in practical scenarios. As an easy and practical way to achieve human-computer- interaction, in this solution hand gesture to speech and text conversion has been used to facilitate the reduction of hardware components.

On the whole, the solution aims to provide aid to those in need thus ensuring social relevance. The people can easily communicate with each other. The user-friendly nature of the system ensure that people can use it without any difficulty and complexity. The application is cost efficient and eliminates the usage of expensive technology.

The project gives us the many advantages of usage area of sign language. After this system, it is an opportunity to use this type of system in any places such as schools, doctor offices, colleges, universities, airports, social services agencies, community service agencies and courts, briefly almost everywhere.

One of the most important demonstrations of the ability for communication to help sign language users communicate with each other occurred. Sign languages can be used everywhere when it is needed and it would reach various local areas. The future works are about developing mobile application of such system that enables everyone be able to speak with deaf people.

CHAPTER-VI

FUTURE SCOPE

6. Future Scope

This is an effective hand gesture recognition system to address the problem of extracting frames from a video and processing it. In the future scope, various hand gestures can be recognized and applied as input to the computer. The hand gestures representing numbers can also be converted into commands to perform related tasks in real time. Enhancing the recognition capability for various lightning conditions, which is encountered as a challenge in this project can be worked upon in future.

Enhanced Gesture Recognition Accuracy:

Continued research and advancements in computer vision algorithms and machine learning techniques will likely improve the accuracy and reliability of gesture recognition. Fine-tuning models, incorporating deep learning approaches, and leveraging large-scale gesture databases can lead to more robust and precise recognition, even in complex or dynamic hand gestures.

Multimodal Integration:

The future of hand gesture to speech and text converters lies in integrating them with other modalities, such as voice recognition and eye-tracking technology. Combining multiple input sources can provide a more comprehensive and natural means of communication, allowing users to seamlessly switch between gestures, voice commands, and eye movements to convey their messages effectively.

Real-Time Translation and Multilingual Support:

Expanding the capabilities of hand gesture to speech and text converters to include real-time translation and multilingual support is a promising area of development. Users will be able to communicate through gestures in their native language while the system instantly translates their gestures into speech and text output in different languages, facilitating communication across diverse linguistic backgrounds.

Adaptability and Personalization:

Future hand gesture to speech and text converters may offer enhanced adaptability and personalization features. These systems could learn and adapt to individual users' gesture styles, preferences, and patterns over time, improving recognition accuracy and providing a more personalized communication experience.

Integration with Internet of Things (IoT) Devices:

The integration of hand gesture to speech and text converters with IoT devices can create a seamless and intuitive user experience. Users will be able to control and interact with various smart devices, such as home automation systems, wearable devices, and smart appliances, through simple hand gestures, eliminating the need for physical buttons or complex user interfaces.

Advanced Gesture-Based Interaction:

Advancements in gesture recognition technology can lead to the development of more sophisticated and expressive gesture sets. This can enable users to perform intricate gestures that convey nuanced meanings or control complex actions, further expanding the possibilities for gesture-based interaction in various domains.

Reference

[1] J.P. Bonet. “Reducci_on de las letras y arte para ense~nar a hablar a los mudos”, Coleccion Cl_asicos Pepe. C.E.P.E., 1992.

[2] William C. Stokoe. Sign Language Structure [microform] / William C. Stokoe. Distributed by ERIC Clearinghouse, [Washington, D.C.], 1978.

[3] William C. Stokoe, Dorothy C Casterline, and Carl G Croneberg. “A Dictionary of American Sign Language on Linguistic Principles” Linstok Press, [Silver Spring, Md.], New Edition, 1976.

[4]Guo D, Zhou W, Wang M, Li H. Sign language recognition based on adaptive HMMS with data augmentation. Proceedings - International Conference on Image Processing, ICIP. 2016; 2016-Augus: p. 2876-2880

[5] Jin CM, Omar Z, Jaward MH. A mobile application of American sign language translation via image processing algorithms. Proceedings - 2016 IEEE Region 10 Symposium, TENSYMP 2016. 2016;: p. 104-109.

[6] Shahriar S, Siddiquee A, Islam T, Ghosh A, Chakraborty R, Khan AI, et al. Real-Time American Sign Language Recognition Using Skin Segmentation and Image Category Classification with Convolutional Neural Network and Deep Learning. IEEE Region 10 Annual International Conference, Proceedings/TENCON. 2019; 2018-Octob(October): p. 1168-1171.

[7] Hore S, Chatterjee S, Santhi V, Dey N, Ashour AS, Balas VE, et al. Optimized Neural Networks. 2017;: p. 139-151

PUBLICATIONS

NOTIFIER FOR ABNORMAL PULSE RATE

Sudhanshu Shukla
B.Tech (Scholar)
Department of Computer Science & Engineering,
Shri Shankaracharya Institute of Professional Management and Technology, Raipur,
Chhattisgarh, India
sudhanshu.shukla@ssipmt.com

Preeti Tuli
Assistant Professor
Department of Computer Science & Engineering,
Shri Shankaracharya Institute of Professional Management and Technology, Raipur,
Chattisgarh, India
p.tuli@ssipmt.com

Anjali Jha
B.Tech (Scholar)
Department of Computer Science & Engineering,
Shri Shankaracharya Institute of Professional Management and Technology, Raipur,
Chattisgarh, India
anjali.jha@ssipmt.com

Simran Banarase
B.Tech (Scholar)
Department of Computer Science & Engineering,
Shri Shankaracharya Institute of Professional Management and Technology, Raipur,
Chhattisgarh, India
simran.banarase@ssipmt.com

ABSTRACT-

According to a survey by WHO, 8% of death all over the world causes due to respiratory issues. It is causing due to genetics, smoking, pollution, and burning biomass fuel. As we know many individuals have lost their lives during the COVID-19 period, especially at the point when doctors cannot physically meet and treat the patients until the situation is more critical. Keeping all this point in mind we have developed a system using IoT to monitor individual health by tracking the heart rate of one. In this way, we are providing the solution to the one having respiratory issues and helping them to get more appropriate treatment over time.

KEYWORDS – Internet of Things, Wi-Fi Module, Arduino Mega, Application Programming Interface, Healthcare

I. INTRODUCTION

An abnormal pulse rate impairs blood circulation in the body. As a result, less oxygen reaches different parts of the body, affecting internal organs damage [1]. As we know, an abnormal pulse rate is harmless and causes uncomfortable symptoms such as chest pain, dizziness, fainting, weakness, and shortness of breath..during lockdown period (i.e. coronavirus period) Physicians cannot see and treat patients in person until the situation is critical [2]. The COVID-19 pandemic has created many problems for everyone. In particular, patients whose heart