A GESTURE- BASED TOOL FOR STERILE BROWSING OF RADIOLOGY IMAGES

(TEAM ID : PNT2022TMID29456)

PROJECT REPORT

Submitted by

PARTHIBAN S (510419104051)
POUVAN SHANKAR M (510419104052)
PRAGADEESHWARAN G (510419104053)
ASHOK KUMAR S (510419104016)
HARI SANKAR M (510419104029)

in partial fulfillment for the award of degree of
BACHELOR OF ENGINEERING
in
COMPUTER SCIENCE AND ENGINEERING

ARUNAI ENGINEERING COLLEGE, TIRUVANNAMALAI
ANNA UNIVERSITY: CHENNAI 600 025
NOVEMBER 2022

INDEX

CHAPTER 1

INTRODUCTION

Computer information technology is increasingly penetrating into the hospital domain. A major challenge involved in this process is to provide doctors with efficient, intuitive, accurate and safe means of interaction without affecting the quality of their work. Keyboards and pointing devices, such as a mouse, are today's principal method of human—computer interaction.
However, the use of computer keyboards and mice by doctors and nurses in intensive care units (ICUs) is a common method for spreading infections. In this project, we suggest the use of hand gestures as an alternative to existing interface techniques, offering the major advantage of sterility. Even though voice control also provides sterility, the noise level in the operating room (OR) deems it problematic.
In this project, we refer to gestures as a basic form of non-verbal communication made with the hands. Psychological studies showed that young children use gestures to communicate before they learn to talk. Manipulation, as a form of gesticulation, is often used when people speak to each other about some object. Naturalness of expression, non-encumbered interaction, intuitiveness and high sterility are all good reasons to replace the current interface technology (e.g., keyboard, mouse, and joystick) with more natural interfaces.

1.1 Project Overview

Humans are able to recognize body and sign language easily. This is possible due to the combination of vision and synaptic interactions that were formed along brain development . In order to replicate this skill in computers, some problems need to be solved: how to separate objects of interest in images and which image capture technology and classification technique are more appropriate, among others.
In this project Gesture based Desktop automation, First the model is trained pre trained on the images of different hand gestures, such as a showing numbers with fingers as 1, 2, 3, 4 . This model uses the integrated webcam to capture the video frame. The image of the gesture captured in the video frame is compared with the Pre-trained model and the gesture is identified. If the gesture predicts is 1 then images is blurred; 2 image is resized; 3 image is rotated etc.

1.2 Purpose

Purpose of the project is used to browse the radiology images using handgestures rather than using mouse, keyboard etc, to maintaining sterility inside the operation room.

CHAPTER 2

LITERATURE SURVEY

1. The recently developed depth sensors, e.g., the Kinect sensor, have provided new opportunities for human-computer interaction (HCI). Although great progress has been made by leveraging the Kinect sensor, e.g., in human body tracking, face recognition and human action recognition, robust hand gesture recognition remains an open problem. Compared to the entire human body, the hand is a smaller object with more complex articulations and more easily affected by segmentation errors. It is thus a very challenging problem to recognize hand gestures. This paper focuses on building a robust part-based hand gesture recognition system using Kinect sensor. To handle the noisy hand shapes obtained from the Kinect sensor, we propose a novel distance metric, Finger-Earth Mover's Distance (FEMD), to measure the dissimilarity between hand shapes. As it only matches the finger parts while not the whole hand, it can better distinguish the hand gestures of slight differences. The extensive experiments demonstrate that our hand gesture recognition system is accurate (a 93.2% mean accuracy on a challenging 10-gesture dataset), efficient (average 0.0750 s per frame), robust to hand articulations, distortions and orientation or scale changes, and can work in uncontrolled environments (cluttered backgrounds and lighting conditions). The superiority of our system is further demonstrated in two real-life HCI applications.

2. In this paper, we address natural human-robot interaction (HRI) in a smart assisted living (SAIL) system for the elderly and the disabled. Two common HRI problems are studied: hand gesture recognition and daily activity recognition. For hand gesture recognition, we implemented a neural network for gesture spotting and a hierarchical hidden Markov model for context-based recognition. For daily activity recognition, a multi sensor fusion scheme is developed to process motion data collected from the foot and the waist of a human subject. Experiments using a prototype wearable sensor system show the effectiveness and accuracy of our algorithms.

3. The use of doctor-computer interaction devices in the operation room (OR) requires new modalities that support medical imaging manipulation while allowing doctors' hands to remain sterile, supporting their focus of attention, and providing fast response times. This paper presents “Gestix,” a vision-based hand gesture capture and recognition system that interprets in real-time the user's gestures for navigation and manipulation of images in an electronic medical record (EMR) database. Navigation and other gestures are translated to commands based on their temporal trajectories, through video capture. “Gestix” was tested during a brain biopsy procedure. In the in vivo experiment, this interface prevented the surgeon's focus shift and change of location while achieving a rapid intuitive reaction and easy interaction. Data from two usability tests provide insights and implications regarding human-computer interaction based on nonverbal conversational modalities.

4. They have developed a system that uses computer vision to replace standard computer mouse functions with hand gestures. The system is designed to enable non-contact human-computer interaction (HCI), so that surgeons will be able to make more effective use of computers during surgery. In this paper, They begin by discussing the need for non-contact computer interfaces in the operating room. They then describe the design of our non-contact mouse system, focusing on the techniques used for hand detection, tracking, and gesture recognition. Finally, They present preliminary results from testing and planned future work.

5. A gesture interface is developed for users, such as doctors/surgeons, to browse medical images in a sterile medical environment. A vision-based gesture capture system interprets user's gestures in real-time to manipulate objects in an image visualization environment. A color distribution model of the gamut of colors of the users hand or glove is built at the start of each session resulting in an independent system. The gesture system relies on real-time robust tracking of the user's hand based on a color-motion fusion model, in which the relative weight applied to the motion and color cues are adaptively determined according to the state of the system. Dynamic navigation gestures are translated to commands based on their relative positions on the screen. A state machine switches between other gestures such as zoom and rotate, as well as a sleep state. Performance evaluation included gesture recognition accuracy, task learning, and rotation accuracy. Fast task learning rates were found with convergence after ten trials. A beta test of a system prototype was conducted during a live brain biopsy operation, where neurosurgeons were able to browse through MRI images of the patient's brain using the sterile hand gesture interface. The surgeons indicated the system was easy to use and fast with high overall satisfaction.

2.1 Existing problem

When the doctors inside the operation room use keyboard and mouse, there is a possibility of spreading infection. So, in this project we used hand gesture methodology to keep the doctor sterile.

2.2 References

1. Robust Part-Based Hand Gesture Recognition Using Kinect Sensor, August 2013, IEEE Transactions on Multimedia 15(5):1110-1120; DOI:10.1109/TMM.2013.2246148v
2. Wearable Sensor-Based Hand Gesture and Daily Activity Recognition for Robot- Assisted LivingIEEE Transactionson systems, Man , andCybernetics Part A: Systemsand humans, Vol .41, No. 3, May 2011.
3. A Gesture-based Tool for Sterile Browsing of Radiology Image , J Am Med Inform Assoc. 2008 May-Jun; 15(3): 321–323, doi: 10.1197/jamia.M241.

4. A Non-Contact Mouse for Surgeon-Computer Interaction, February 2004, Technology and health care: official journal of the European Society for Engineering and Medicine 12(3):245-57; DOI:10.3233/THC-2004-12304.

5.Real-Time Hand Gesture Interface for Browsing Medical Images, February 2007; International Journal of Intelligent Computing in Medical Sciences and Image Processing 2(1); DOI:10.1080/1931308X.2008.10644149.

2.3 Problem Statement Definition

Problem Statement (PS)
I am (Customer)
I'm trying to
But
Because
Which makes me feel
PS-1
Deaf and dumb person
To overcome my disabilities
Difficult to establish communication with others
To convey my message to a normal people is always a challenging
task for me
Sad and frustrated
PS-2
Driver and passenger
To interact with vehicles using hand gestures
Unexpected input recognition and behaviour
Drivers
don't need to take attention off from the road
Feel like the passenger and driver are safe

CHAPTER 3

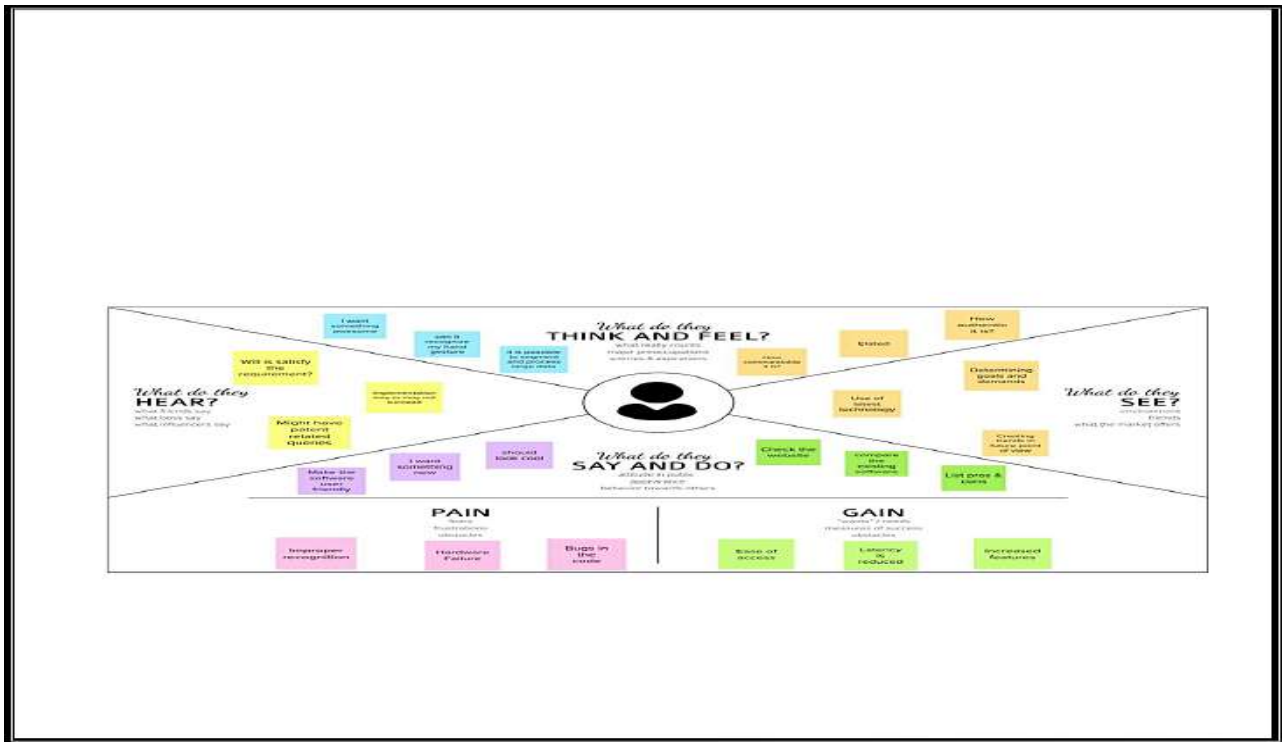
IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas

An empathy map is a simple, easy-to-digest visual that captures knowledge about a user's behavior and attitudes.

It is a useful tool to helps teams better understand their users.
Creating an effective solution requires understanding the true problem and the person who is experiencing it. The exercise of creating the map helps participants consider things from m the user's perspective along with his or her goals and challenges.

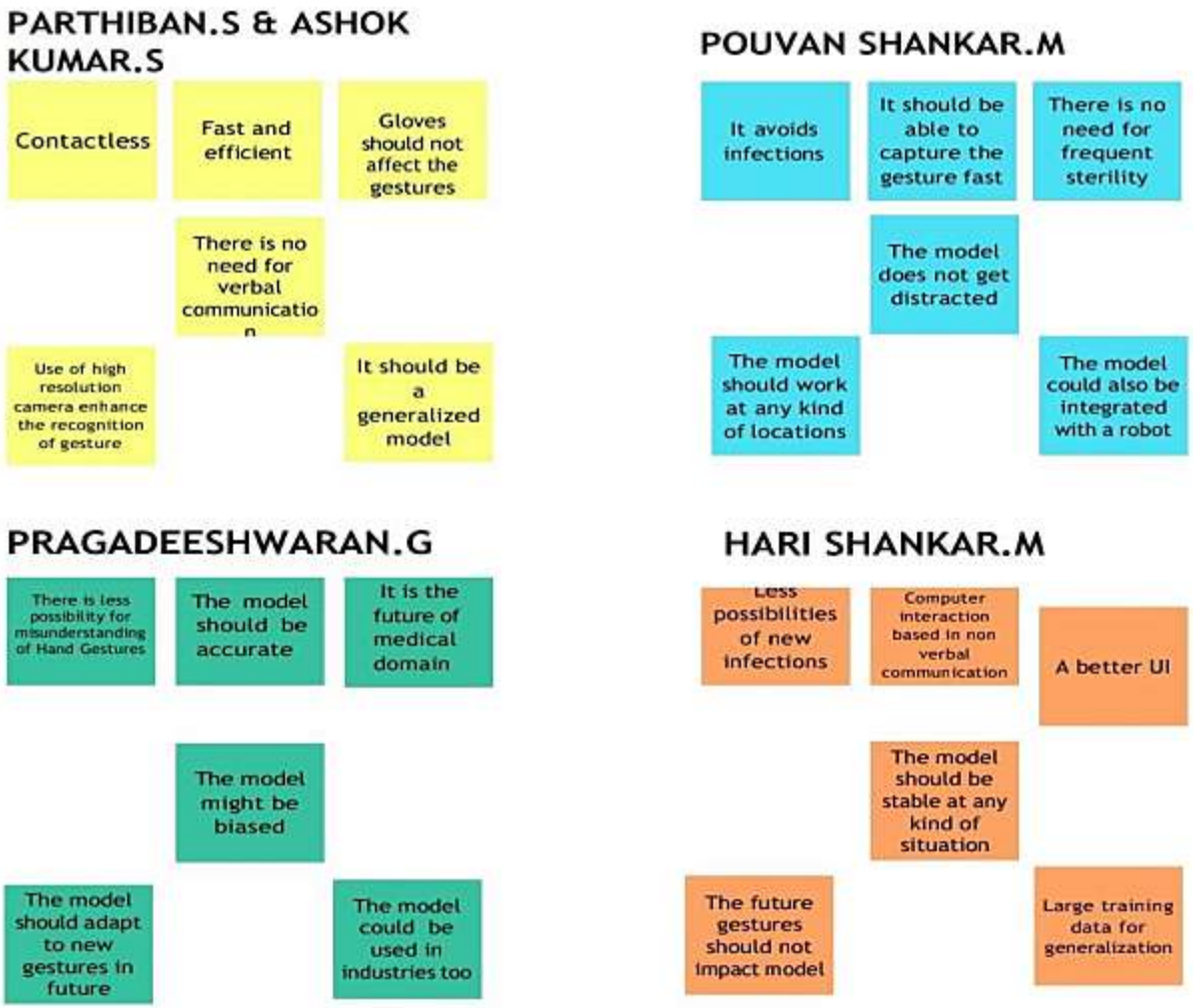
Empathy map



3.2 Ideation & Brainstorming

Step-1: Team Gathering, Collaboration and Select the Problem Statement

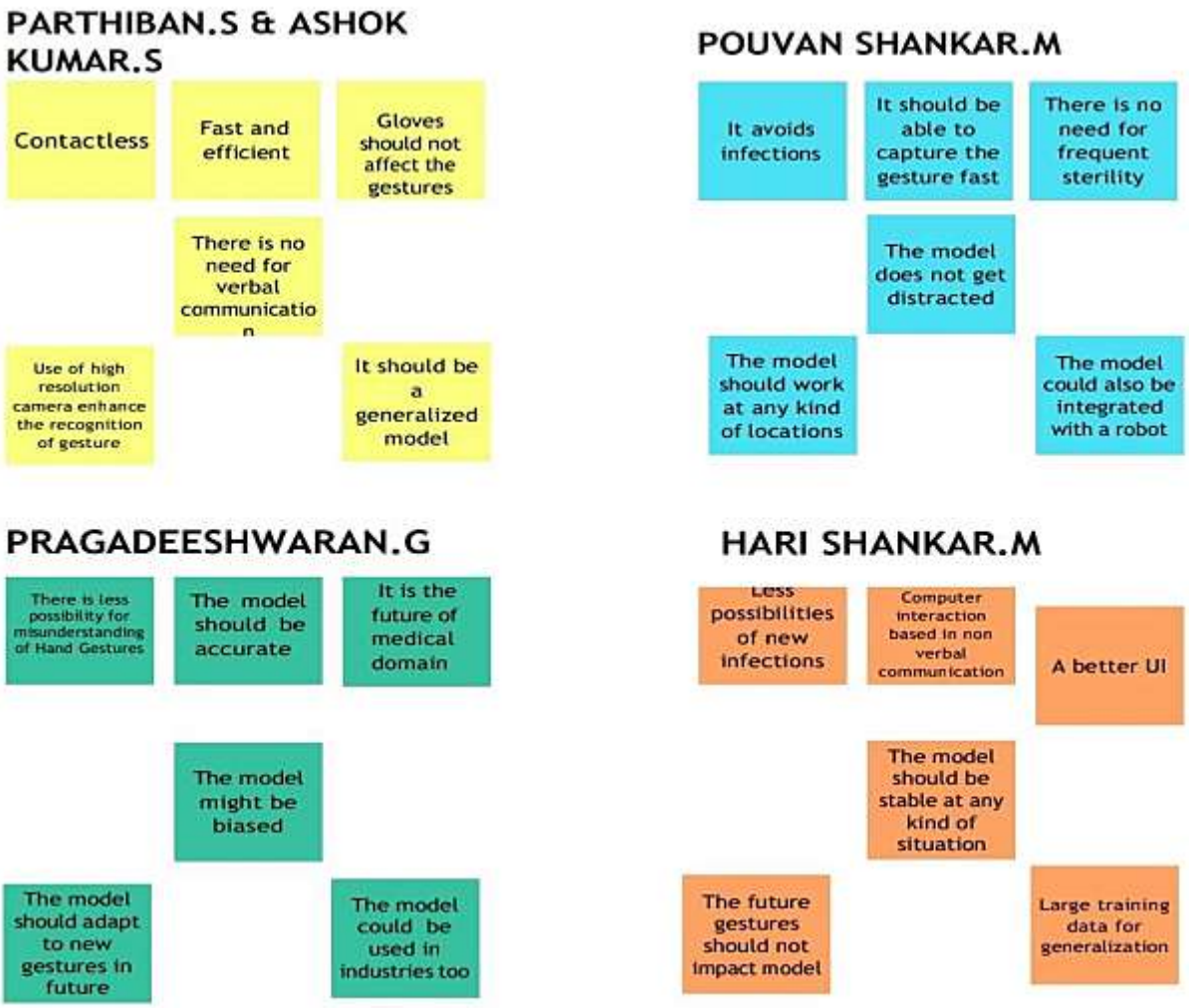
Step-2: Brainstorm, Idea Listing and Grouping



3.2 Ideation & Brainstorming

Step-1:Team Gathering,Collaboration and Select the Problem Statement

Step-2: Brainstorm, Idea Listing and Grouping



Step-3: Group ideas

3.3 Proposed Solution

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	“Hand Gesture Recognition Using Camera” is based on concept of Image processing. In recent year there is lot of research on gesture recognition using kinetic sensor on using HD camera but camera and kinetic sensors are more costly. This project is mainly focused on to reduce cost and improve robustness of the proposed system using simple web camera.
2.	Idea / Solution description	Most gesture recognition methods usually contain three major stages. The first stage is the object detection. The target of this stage is to detect hand objects in the digital images or videos. Many environment and image problems are needed to solve at this stage to ensure that the hand contours or regions can be extracted precisely to enhance the recognition accuracy. Common image problems contain unstable brightness, noise, poor resolution and contrast. The better environment and camera devices can effectively improve these problems. However, it is hard to control when the gesture recognition system is working in the real environment or is become a product. Hence,

		the image processing method is a better solution to solve these image problems to construct an adaptive and robust gesture recognition system. The second stage is object recognition. The detected hand objects are recognized to identify the gestures. At this stage, differentiated features and effective classifiers selection are a major issue in most researches. The third stage is to analyze sequential gestures to identify users' instructs or behaviors.
3.	Novelty / Uniqueness	In this project, we mainly focus on using pointing behavior for a natural interface, Handgesture recognition based human machineinterface isbeing developed vigorously in recent years. Due to the effect of lightningandcomplex background,mostvisual hand gesture recognitionsystemswork onlyunder restricted environment. To classify the dynamic hand gestures, we developed a simple and fast motion history image based method. In recent years, the gesture control techniquehas becomea new developmental trend for many humanbased electronicsproducts.This technique let people can control these products more naturally, intuitively and conveniently. In this paper, a fast gesture recognition scheme is proposed to be an interaction(HMI)systems.Our projectmainlypresentssome low complexity algorithms and gestures to reduce the gesture recognition complexity and bemore suitable for controlling real-time computer systems.
4.	Social Impact / Customer Satisfaction	Gesture technology comes as a boon to society, providing contact-less, safe, and inclusive

		experiences and they are easier representation, makes the presentation attractive, Quick expressing of message. A study shows that when the speaker uses gestures, the probability of the audience remembering the point is double than a normal speech.
5.	Business Model (Revenue Model)	Hand gesture recognition is a process of understanding and classifying meaningful movements by the human hands. Nowadays vehicles launched from the industry offers an increasing number of infotainment systems as well as comfort functions that can be controlled by the driver. Though they are feature rich which demands more attention of the driver and degrade the driving performance and thereby reducing the safety. The gestural interaction is a promising means to cover the full range of driver's operational needs while minimizing the visual workload thereby enhancing the drivers safety.
6.	Scalability of the Solution	Human hand is very smaller with very complex articulations comparing with the entire human body and therefore errors can be easily affected. Hand gesture recognition is of great importance for human computer interaction (HCI) because of its extensive applications in virtual reality and sign language recognition.

3.4 Problem Solution fit

CHAPTER 4

REQUIREMENT ANALYSIS

4.1 Functional Requirements

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	Identifying User Gestures	The user gestures are identified using the images of gestures captured by the camera
FR-2	Deployment in Cloud	The trained Deep Learning Model is deployed in cloud, which could be accessed anywhere around the world
FR-3	User Interface	The user interface, which helps in the Human Computer Interaction is designed
FR-4	Gestures related to the Application Domain	The model should be trained with the gestures related to the application domain.

4.2. Non-functional Requirements

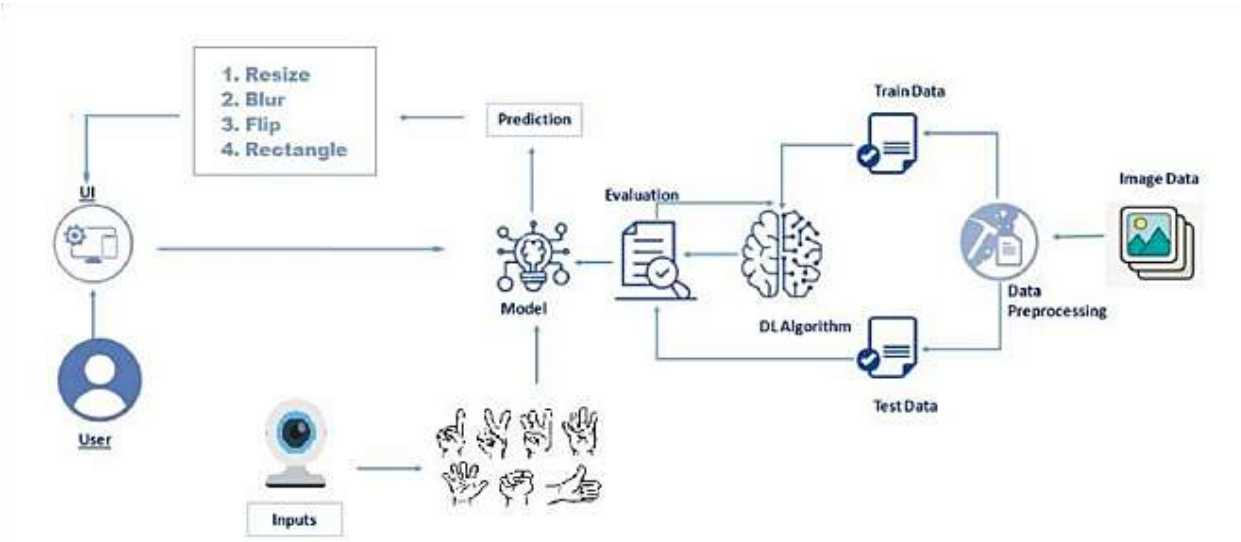
FR No.	Non-Functional Requirement	Description
NFR-1	Usability	The user interface which acts as an intermediate between the user and the DL Model which is deployed in the cloud

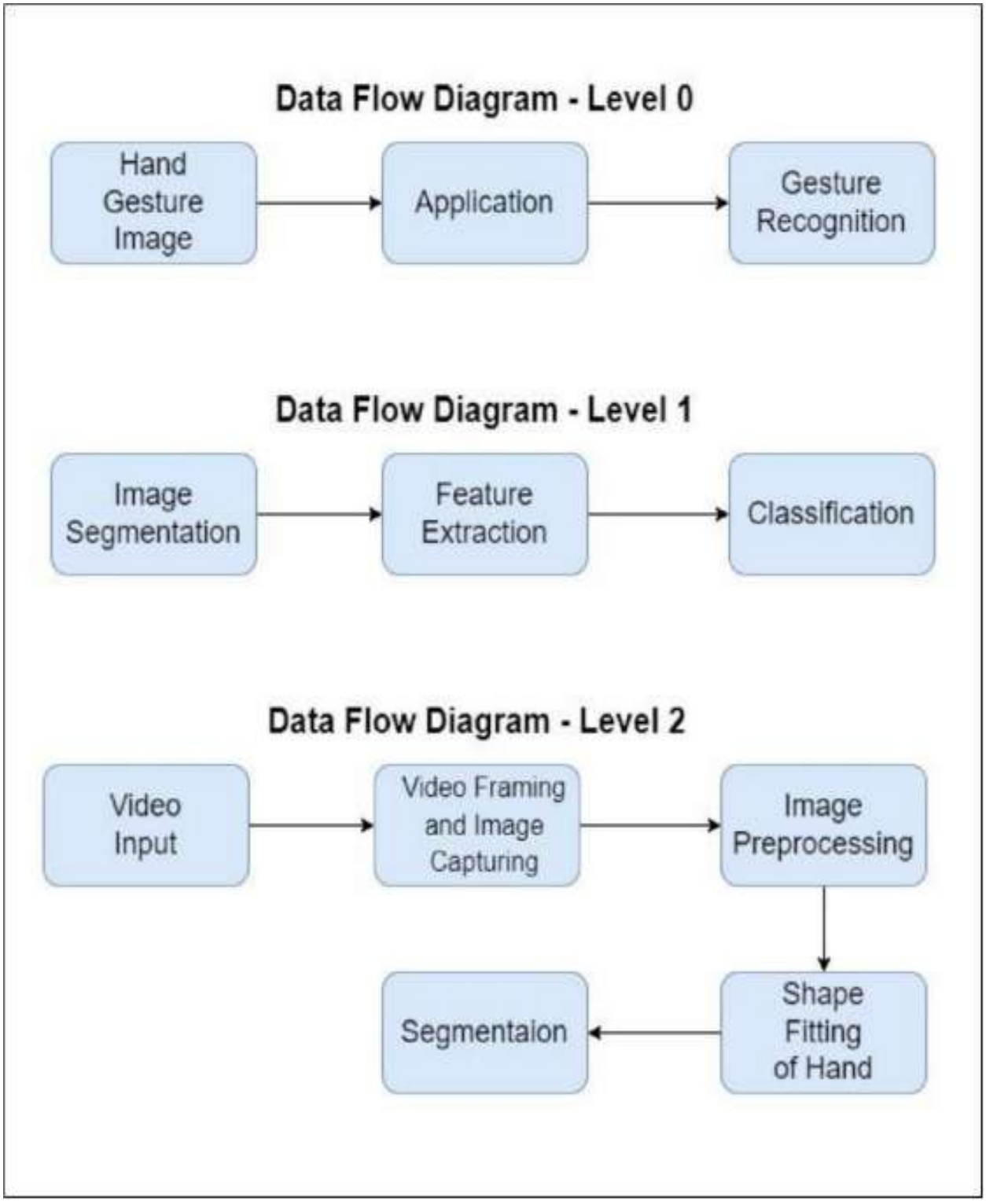
NFR-2	Security	The model deployed in the cloud should be accessible only by the approved users and it should be inaccessible by the attackers or the terrorists
NFR-3	Reliability	The tool or the system is 95% reliability for a year
NFR-4	Performance	The tool or the system should respond with the accurate response within 4-5 seconds
NFR-5	Availability	The model deployed in the cloud must be available to 99.8% of the people over a month during working hours
NFR-6	Scalability	The model deployed in the cloud must be accessible by over 10,00,000 people trying to access it using the user interface

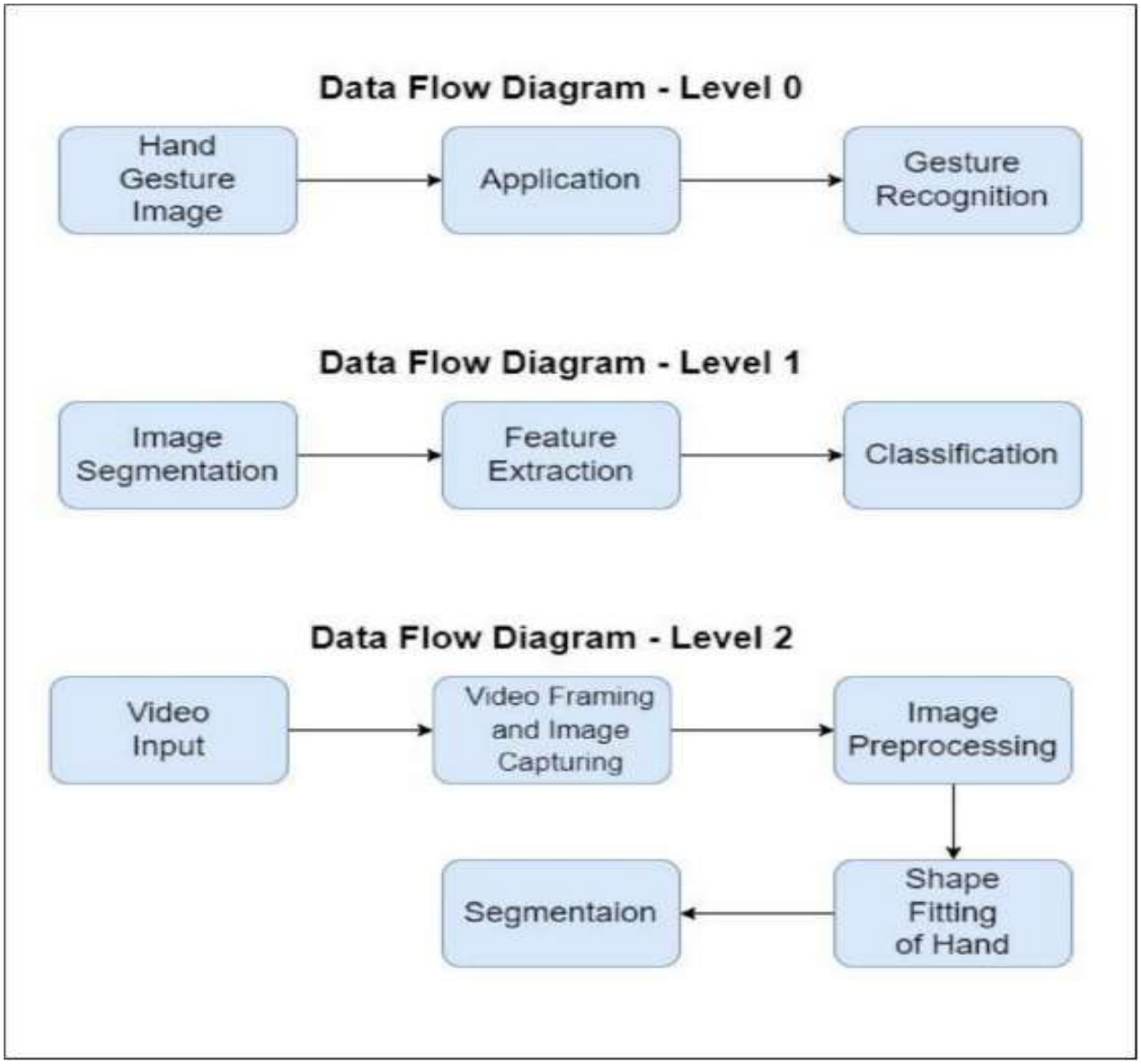
CHAPTER 5

PROJECT DESIGN

5.1 Data flow diagams




Data Flow Diagram - Level 0
Hand Gesture Image
Application
Gesture Recognition
Data Flow Diagram - Level 1
Image Segmentation
Feature Extraction
Classification
Data Flow Diagram - Level 2
Video Input
Video Framing and Image Capturing
Image Preprocessing
Shape Fitting of Hand
Segmentaion



User Stories:

Use the below templates to list all the user stories of the product

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release

Customer (Mobile user)	Launch Web App deployed in cloud	USN-1	As a user, I can launch the webapp where I can upload the images for recognition	I can upload the images for classification	High	Sprint-4
	Predictin	USN-2	As a user, I can get the predicted results from the model deployed in the cloud	I can resize the radiology image, blur the image, flip based on the hand gesture	High	Sprint-4
	Deploym ent of webapp in the cloud	USN-3	As a user, I need the webapp to be accessible all over the world	I can access the webapp deployed in the IBM cloud	Medium	Sprint-3
	Deploym ent of AI model in the cloud	USN-4	As a user, I need the AI model to be accessible all over the world	I can access the model deployed in the IBM cloud	Medium	Sprint-3
	Model Building	USN-5	As a user, I need an AI model which could classify or recognize the hand gestures	I can get the prediction from the AI model	Medium	Sprint-1
	User Interface Building	USN-6	As a user, I need a web app for human	I get User Interface for interaction	Medium	Sprint-2

			computer interaction	with the model		

Technical Architecture

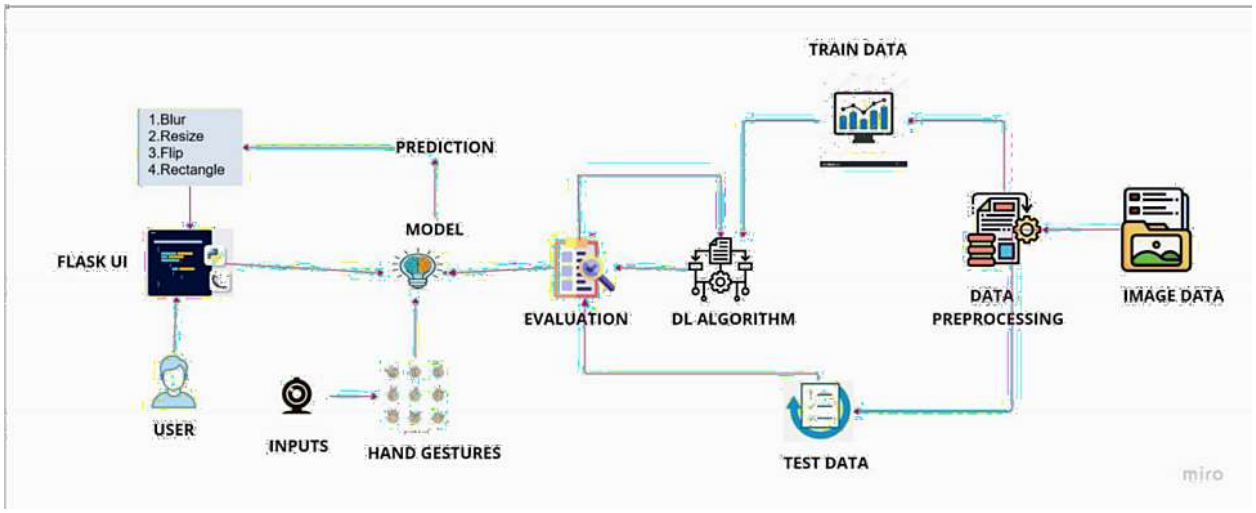


Table-1 : Components & Technologies:

S.No	Component	Description	Technology
1.	User Interface	UI (Web)	HTML, CSS, JavaScript.
2.	Application Logic-1 Image Pre-processing	Input image is pre-processed with the help of library files.	Python, TensorFlow
3.	Application Logic-2Building Model	Building CNN model to recognize the gesture.	Python, Keras

4.	Application Logic-3Creation of app	App is built to obtain gesture as input and to provide as output.	HTML, CSS, JavaScript
5.	Dataset	Hand gesture data set.	From IBM
6.	Cloud Database	User input image is stored in cloud.	IBM Cloud
7.	File Storage	File storage contains dataset and source code.	Device or Drive
8.	Machine Learning Model	CNN Model was used to recognize the pre-processed image by image capturing or by video segmenting.	CNN Model by Python, Keras

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	Application development, data pre-processing.	Visual studio code, Anaconda navigator, TensorFlow
2.	Security Implementations	It identifies the gesture only when the hand is in front of the camera.	OpenCV
3.	Scalable Architecture	It can be used in any environment and is able to identify the gesture	OpenCV

4.	Availability	It is used to reduce the possibility of spreading infections	AI
5.	Performance	Rapid response to the gesture.	CNN

User Stories:

User Type	Functional Requirement(Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer	Login	USN-1	Entering Webpage	Enter the application	High	Sprint 1
	Homepage	USN-2	Entering to the “Homepage” of theUI(Webpage)	Enter the homepage	High	Sprint 1
	About	USN-3	I can click on the “About” to details about theApplication	Get the details about the application	Low	Sprint 2
	Begin	USN-4	As a user I can upload my radiology imagefrom the computer.	Choose any imagefrom my device	High	Sprint 2
	Predict	USN-5	As a user I can turn on the camera using predict button	Turn on the camera for prediction	High	Sprint 3

		USN-6	Predicting the images using Hand Gesture	Can resize, blur, and flip my image using my hand gesture	High	Sprint 3
		USN-7	I can give a gesture of raised fist and it recognizes	Can get my fixed resized image	High	Sprint 4
		USN-8	I can show my index finger	Can get a rectangular image	High	Sprint 4
		USN-9	I can show my index finger, middle finger and ring finger at once	Can get my image blurred	High	Sprint 4

CHAPTER 6

PROJECT PLANNING & SCHEDULING

Product Backlog, Sprint Schedule, and Estimation :

Use the below template to create product backlog and sprint schedule

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Priority	Team Members
Sprint-1	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	High	Parthiban , Pouvan shankar
		USN-2	As a user, I will receive confirmation email once I have registered for the application	High	Pragadeeshwaran , Ashok kumar
		USN-3	As a user, I can register for the application through Facebook	Low	Parthiban , Hari shankar
		USN-4	As a user, I can register for the application through Gmail	Medium	Pouvan shankar , Ashok kumar

Sprint 2	Login	USN-5	As a user, I can log into the application by entering email & password	High	Pragadeeshwaran , Parthiban
	Dashboard	USN-6	As a user, I can view my profile and update my details	Medium	Parthiban , Hari shankar
		USN-7	As a user, I can view all images uploaded High	High	Pouvan Shankar, Ashok kumar
		USN-8	As a user, I can change my password	High	Parthiban , Pragadeeshwaran
Sprint-3	Image Capturing	USN-9	As a user, I can capture images of hand gestures made by me	High	Hari shankar , Ashok kumar
	Image Processing	USN-10	In the application, the captured images are processed to identify the hand gesture	High	Pragadeeshwaran , Hari shankar
	Data Storage	USN-11	In the application radiology images uploaded by the user	High	Pragadeeshwaran , Ashok kumar

			are stored using a database		
Sprint-4	Sterile Browsing	USN-12	Depending on the different gesture inputs different operations are performed on the input image	High	Pouvan Shankar , Parthiban
	Displaying the operations performed	USN-13	Once the model analyses the gesture, the prediction with operation applied on the image is showcased on the user interface	High	Ashok kumar, Parthiban , Hari shankar

Project Tracker, Velocity & Burndown Chart:

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	20	6 Days	26 Oct 2022	30 Oct 2022	20	30 Oct 2022
Sprint-2	20	6 Days	01 Nov 2022	06 Nov 2022	20	06 Nov 2022
Sprint-3	20	6 Days	08 Nov 2022	13 Nov 2022	20	13 Nov 2022
Sprint-4	20	6 Days	15 Nov 2022	20 Nov 2022	20	20 Nov 2022

Velocity:

Imagine we have a 10-day sprint duration, and the velocity of the team is 20 (points per sprint). Let's calculate the team's average velocity (AV) per iteration unit (story points per day)

$$AV = \frac{sprint\ duration}{velocity} = \frac{20}{10} = 2$$

Burn down Chart:

A burn down chart is a graphical representation of work left to do versus time. It is often used in agile software development methodologies suchas Scrum. However, burn down charts can be applied to any project containing measurable progress over time.

CHAPTER 7

CODING & SOLUTIONING

home.html

```
<html>
<script>

</script>
<body>
  <div class="main">
    <div class="navbar">
        <div class="icon">
          <p class="logo"><h1>Hand Gestures System</h1></p>
        </div>

           <div class="menu">
            <ul>
             <li><a class="active" href="{{ url_for('home')}}">HOME</a></li>
             <li><a class="active" href="{{ url_for('intro')}}">INTRODUTION</a></li>
             <li><a class="active" href="{{ url_for('image1')}}">LAUNCH</a></li>
            </ul>
           </div>
        </div>
    <div class="content">
      <p class="par">
        <h2> GESTURE RECOGNITION OF RADIOLOGY IMAGES,<br>
        THROUGH STERILE BROWSING.</h2></p>

     </div>
  </div>

<style>

*{
   margin: 0;
   padding: 0;
}

.main{
```

```
    width: 100%;
    height: 100vh;



background-image: linear-gradient(to top, rgba(0,0,0,0.5)50%,rgba(0,0,0,0.5)50%),
url("https://miro.medium.com/max/1100/0*cLUSgHqt5kXYCXFZ.gif");
    background-position: center;
    background-size: cover;
}

.navbar{
    width: 1500px;
    height: 130px;
    margin: top;
}

.icon{
    font-size: 20px;
    padding-left: 2%;
    padding-top: 1%;
    float: left;
    height: 50px;
    color: #ff7200;
    align-items: center;

}

.logo{
    color: #ff7200;
    padding-right: 10px;
    width: 85%;
    margin: auto;
    padding: 35px o;
    display: flex;
    align-items: center;
    justify-content: space-between;
}
```

```
.menu{
    width: 400px;
    float: right;
    height: 70px;
}

ul{
    float: right;



padding-left: 20px;
    display: flex;
    justify-content: center;
    align-items: center;

}

ul li{
    list-style: none;
    margin-left: 72px;
    margin-right: 15px;
    margin-top: 27px;
    font-size: 17px;
}

ul li a{
    text-decoration: none;
    color: #fff;
    font-family: Arial;
    font-weight: bold;
    transition: 0,4s ease-in-out;
}

ul li a:hover{
    color: #ff7200;
}

.content{
    width: 1200px;
```

```
    height: auto;
    margin: auto;
    color: #fff;
    position: relative;
}

.content .par{
    padding-left: 20px;
    padding-bottom: 100px;
    font-family: Arial;
    letter-spacing: 3.2px;
    line-height: 80px;
}

.content h1{
    font-family: 'Times New Roman';
    font-size: 50px;
    padding-left: 20px;
    margin-top: 9px;
    letter-spacing: 2px;
}

.content .cn{
    width: 160px;
    height: 40px;
    background: #ff7200;
    border: none;
    margin-bottom: 10px;
    margin-left: 20px;
    font-size: 18px;
    border-radius: 10px;
    cursor: pointer;
    transition: .4s ease;
}

.content .cn a{
    text-decoration: none;
    color: #000;
    transition: .3s ease;
}
```

```
.cn:hover{
   background-color: #fff;
}

.content span{
   color: #ff7200;
   font-size: 60px;
}

@import url("https://fonts.googleapis.com/css?family=Luckiest+Guy");

</style>



</html>
```

intro.html

```
<html>
<script>
     </script>

<body>
  <h1 style="color:red">INTRODUCTION</h1><br>
<h1 style="color:white">
  <p style="align-content: center;text-align: justify; margin-left: 0;margin-right: 0;">
  # Gesture recognition is a technology that uses sensors to read and interpret hand movements as commands. It is a subdiscipline
  of computer vision. Gestures can originate from any bodily motion or state, but commonly originate from the face or hand.<br><br>
   #  Current focuses in the field include emotion recognition from face and hand gesture recognition. Users can use simple gestures to control or
  interact with devices without physically touching them. Many approaches have been made using cameras and computer vision algorithms
  to interpret sign language.<br><br>  # Gesture recognition can be seen as a way for computers to begin to understand human body language, thus
```

```
  building a better bridge between machines and humans than older text user interfaces or even GUIs (graphical user interfaces),which
  still limit the majority of input to keyboard and mouse and interact naturally without any mechanical devices.</p>
  </h1>
<!--Brian Tracy-->

<div class="header">


<div style="width:30%;float:left;font-size:2vw;text-align:left;color: #ff7200;font-weight: bold; padding-top:1.5%;padding-left:2%;"> GESTURE RECOGNITION </div>
<div class="menu">
       <ul>
        <li><a class="active" href="{{ url_for('home')}}">HOME</a></li>
        <li><a class="active" href="{{ url_for('intro')}}">INTRODUTION</a></li>
        <li><a class="active" href="{{ url_for('image1')}}">LAUNCH</a></li>
       </ul>
     </div>
</div>
</body>

  <style>
.header {
  position: relative;
                              top:0;
                              margin:0px;
                              z-index: 1;
                              left: 0px;
                              right: 0px;
                              position: fixed;
                              background-color: none ;
                              color: white;
                              overflow: hidden;
                              padding-left:20px;
                              font-family: 'Josefin Sans';
                              font-size: 2vw;
                              width: 100%;
                              height:8%;
                              text-align: center;
```

```
}

.menu{
    width: 400px;
    float: right;
    height: 70px;
}
ul{
    float: right;
    padding-left: 20px;
    display: flex;
    justify-content: center;
    align-items: center;

}

ul li{
    list-style: none;
    margin-left: 20px;
    margin-right: 90px;
    margin-top: 27px;
    font-size: 17px;
}

ul li a{
    text-decoration: none;
    color: #fff;
    font-family: Arial;
    font-weight: bold;
    transition: 0,4s ease-in-out;
}

ul li a:hover{
    color: #ff7200;
}



.topnav-right a {
  font-weight: arial;
```

```
  font-weight: bold;
  color: #fff;
  padding: 14px 16px;
  text-decoration: none;
}


.topnav-right a:hover {
 color: #ff7200;
}



.topnav-right {
  float: right;
  padding-right:100px;
}

body {

  background-color: black;
  background-repeat: no-repeat;
  background-size:cover;
  background-size: cover;
  background-position: 0px 0px;
  }
  .button {
  background-color: #091425;
  border: none;
  color: white;
  padding: 15px 32px;
  text-align: center;
  text-decoration: none;
  display: inline-block;
  font-size: 12px;
  border-radius: 16px;
}
```

```
.button:hover {
  box-shadow: 0 12px 16px 0 rgba(0,0,0,0.24), 0 17px 50px 0 rgba(0,0,0,0.19);
}
form {border: 3px solid #f1f1f1; margin-left:400px;margin-right:400px;}

input[type=text], input[type=password] {
  width: 100%;
  padding: 12px 20px;
  display: inline-block;
  margin-bottom:18px;
  border: 1px solid #ccc;
  box-sizing: border-box;
}

button {
  background-color: #091425;
  color: white;
  padding: 14px 20px;
  margin-bottom:10px;
  border: none;
  cursor: pointer;
  width: 17%;
  border-radius:4px;
  font-family:Montserrat;
}



button:hover {
  opacity: 0.8;
}

.cancelbtn {
  width: auto;
  padding: 10px 18px;
  background-color: #f44336;
}

.imgcontainer {
```

```
  text-align: center;
  margin: 24px 0 12px 0;
}

img.avatar {
  width: 30%;
  border-radius: 50%;
}
.container {
  padding: 16px;
}
span.psw {
  float: right;
  padding-top: 16px;
}


/* Change styles for span and cancel button on extra small screens */
@media screen and (max-width: 300px) {
  span.psw {
     display: block;
     float: none;
  }
  .cancelbtn {
     width: 100%;
  }
}

.home{
          margin:80px;
          width: 84%;
  height: 500px;
  padding-top:10px;
  padding-left: 30px;
}
.login{
          margin:80px;
          box-sizing: content-box;
  width: 84%;
  height: 420px;
```

```
  padding: 30px;
  border: 10px solid blue;
}


.left,.right{
 box-sizing: content-box;
 height: 400px;
 margin:20px;
 border: 10px solid blue;
}

.mySlides {display: none;}
img {vertical-align: middle;}

/* Slideshow container */
.slideshow-container {
  max-width: 1000px;
  position: relative;
  margin: auto;
}

/* Caption text */
.text {
  color: #f2f2f2;
  font-size: 15px;
  padding: 8px 12px;
  position: absolute;
  bottom: 8px;
  width: 100%;



 text-align: center;
}
/* The dots/bullets/indicators */
.dot {
  height: 15px;
  width: 15px;
  margin: 0 2px;
```

```
  background-color: #bbb;
  border-radius: 50%;
  display: inline-block;
  transition: background-color 0.6s ease;
}
.active {
  background-color: none;
}

/* Fading animation */
.fade {
  -webkit-animation-name: fade;
  -webkit-animation-duration: 1.5s;
  animation-name: fade;
  animation-duration: 1.5s;
}
@-webkit-keyframes fade {
  from {opacity: .4}
  to {opacity: 1}
}

@keyframes fade {
  from {opacity: .4}
  to {opacity: 1}
}

/* On smaller screens, decrease text size */
@media only screen and (max-width: 300px) {
  .text {font-size: 11px}
}
@import url("https://fonts.googleapis.com/css?family=Montserrat&display=swap");
* {
  padding: 0;
  margin: 0;
}
body {
  height: 100vh;
  display: flex;
  flex-direction: column;
  justify-content: center;
```

```
  align-items: center;
}
h1 {
  font-family: "Montserrat Medium";
  max-width: 90ch;
  text-align: center;
  transform: scale(0.94);
  animation: scale 3s forwards cubic-bezier(0.5, 1, 0.89, 1);
}
@keyframes scale {
  100% {
    transform: scale(1);
  }
}
span {
  display: inline-block;
  opacity: 0;
  filter: blur(4px);
}
span:nth-child(1) {
  animation: fade-in 1s 0.1s forwards cubic-bezier(0.11, 0, 0.5, 0);
}
span:nth-child(2) {
  animation: fade-in 0.8s 0.2s forwards cubic-bezier(0.11, 0, 0.5, 0);
}
span:nth-child(3) {
  animation: fade-in 0.8s 0.3s forwards cubic-bezier(0.11, 0, 0.5, 0);
}
span:nth-child(4) {
  animation: fade-in 0.8s 0.4s forwards cubic-bezier(0.11, 0, 0.5, 0);
}
span:nth-child(5) {
  animation: fade-in 0.8s 0.5s forwards cubic-bezier(0.11, 0, 0.5, 0);
}



span:nth-child(6) {
  animation: fade-in 0.8s 0.6s forwards cubic-bezier(0.11, 0, 0.5, 0);
}
```

```
span:nth-child(7) {
  animation: fade-in 0.8s 0.7s forwards cubic-bezier(0.11, 0, 0.5, 0);
}
span:nth-child(8) {
  animation: fade-in 0.8s 0.8s forwards cubic-bezier(0.11, 0, 0.5, 0);
}
span:nth-child(9) {
  animation: fade-in 0.8s 0.9s forwards cubic-bezier(0.11, 0, 0.5, 0);
}
span:nth-child(10) {
  animation: fade-in 0.8s 1s forwards cubic-bezier(0.11, 0, 0.5, 0);
}
span:nth-child(11) {
  animation: fade-in 0.8s 1.1s forwards cubic-bezier(0.11, 0, 0.5, 0);
}
span:nth-child(12) {
  animation: fade-in 0.8s 1.2s forwards cubic-bezier(0.11, 0, 0.5, 0);
}

span:nth-child(13) {
  animation: fade-in 0.8s 1.3s forwards cubic-bezier(0.11, 0, 0.5, 0);
}



span:nth-child(14) {
  animation: fade-in 0.8s 1.4s forwards cubic-bezier(0.11, 0, 0.5, 0);
}
span:nth-child(15) {
  animation: fade-in 0.8s 1.5s forwards cubic-bezier(0.11, 0, 0.5, 0);
}
span:nth-child(16) {
  animation: fade-in 0.8s 1.6s forwards cubic-bezier(0.11, 0, 0.5, 0);
}
span:nth-child(17) {
  animation: fade-in 0.8s 1.7s forwards cubic-bezier(0.11, 0, 0.5, 0);
}
span:nth-child(18) {
  animation: fade-in 0.8s 1.8s forwards cubic-bezier(0.11, 0, 0.5, 0);
}
```

```
span:nth-child(19) {
  animation: fade-in 0.8s 1.9s forwards cubic-bezier(0.11, 0, 0.5, 0);
}
span:nth-child(20) {
  animation: fade-in 0.8s 2.0s forwards cubic-bezier(0.11, 0, 0.5, 0);
}
span:nth-child(21) {
  animation: fade-in 0.8s 2.1s forwards cubic-bezier(0.11, 0, 0.5, 0);
}




span:nth-child(22) {
  animation: fade-in 0.8s 2.2s forwards cubic-bezier(0.11, 0, 0.5, 0);
}
span:nth-child(23) {
  animation: fade-in 0.8s 2.3s forwards cubic-bezier(0.11, 0, 0.5, 0);
}span:nth-child(24) {
  animation: fade-in 0.8s 2.4s forwards cubic-bezier(0.11, 0, 0.5, 0);
}span:nth-child(25) {
  animation: fade-in 0.8s 2.5s forwards cubic-bezier(0.11, 0, 0.5, 0);
}span:nth-child(26) {
  animation: fade-in 0.8s 2.6s forwards cubic-bezier(0.11, 0, 0.5, 0);
}span:nth-child(27) {
  animation: fade-in 0.8s 2.7s forwards cubic-bezier(0.11, 0, 0.5, 0);
}span:nth-child(28) {
  animation: fade-in 0.8s 2.8s forwards cubic-bezier(0.11, 0, 0.5, 0);
}
@keyframes fade-in {
  100% {
    opacity: 1;
    filter: blur(0);
  }
}

</style>
</html>
```

launch.html

```
<html lang="en">
     <head>
  <meta charset="utf-8">
     <meta http-equiv="X-UA-Compatible" content="IE=edge">
     <meta name="viewport" content="width=device-width, initial-scale=0.6">
     <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.3.1/jquery.min.js"></script>
     <link href="https://fonts.googleapis.com/icon?family=Material+Icons" rel="stylesheet">
     <meta charset="UTF-8">
     <title>Predict</title>
     <link href="https://cdn.bootcss.com/bootstrap/4.0.0/css/bootstrap.min.css" rel="stylesheet">
     <script src="https://cdn.bootcss.com/popper.js/1.12.9/umd/popper.min.js"></script>
     <script src="https://cdn.bootcss.com/jquery/3.3.1/jquery.min.js"></script>
     <script src="https://cdn.bootcss.com/bootstrap/4.0.0/js/bootstrap.min.js"></script>
     <link href="{{ url_for('static', filename='css/main.css') }}" rel="stylesheet">
  <style>

.menu{
    width: 400px;
    float: right;
    height: 70px;
}
ul{
    float: right;
    padding-left: 30px;



 padding-bottom: 40px;
    padding-right: 20px;
    display: flex;
    justify-content: center;
    align-items: center;

}
ul li{
    list-style: none;
    margin-left: 80px;
```

```
    margin-right: 10px;
    margin-top: 0px;
    padding-left: 5px;
    font-size: 20px;
}
ul li a{
    text-decoration: none;
    color: #ff7200;
    font-family: Arial;
    font-weight: bold;
    transition: 0,4s ease-in-out;
}
ul li a:hover{
    color: white;
}
.bar
  {
  margin: 0px;

 padding:20px;
  background-color:black;
  opacity:0.6;
  color:black;
  font-family:'Roboto',sans-serif;
  font-style: italic;
  border-radius:20px;
  font-size:25px;
  }
a
  {
  color:black;
  float:right;
  text-decoration:none;
  font-style:normal;
  padding-right:30px;
  }
div1{
    text-align: center;
    width: 650spx;
    height: 800px;
```

```
    padding: 190px;
    margin: 10px;
    position: absolute;
  }
body
  {


background-image: url("https://img3.goodfon.com/wallpaper/nbig/a/af/ruki-znaki-steny.jpg");
      background-size: cover;
  }
.header {         position: relative;
        top:0;
        margin:0px;
        z-index: 1;
        left: 0px;
        right: 0px;
        position: fixed;
        background-color: none;
        color: white;
        box-shadow: none;
        overflow: hidden;
        padding-left:20px;
        font-family: 'Josefin Sans';
        font-size: 2vw;
        width: 100%;
        height:8%;
        text-align: center;
      }
.topnav {
    overflow: hidden;
    background-color:#ff7200;
  }
  .topnav-right a {



    float: left;
    color: black;
    text-align: center;
```

```
  padding: 14px 16px;
  text-decoration: none;
  font-size: 18px;
}
.topnav-right a.active {
  background-color: white;
}

.topnav-right a:hover {
  background-color: rgb(181, 228, 236);
  color: rgb(6, 27, 36);
}
.topnav-right {
  float: right;
  padding-right:100px;
}
.button {
background-color: #091425;
border: none;
color: black;
padding: 15px 32px;
text-align: center;
text-decoration: none;
display: inline-block;



font-size: 12px;
 border-radius: 16px;
 }
 .button:hover {
  box-shadow: 0 12px 16px 0 rgba(0,0,0,0.24), 0 17px 50px 0 rgba(0,0,0,0.19);
 }
 form {
  border: 2px solid black;
  margin-left:400px;
  margin-right:400px;}
 input[type=text], input[type=password] {
  width: 100%;
  padding: 12px 20px;
```

```
  display: inline-block;
  margin-bottom:18px;
  border: 1px solid #ccc;
  box-sizing: border-box;
}
button {
  background-color: none;
  color: black;
  padding: 14px 20px;
  margin-bottom:10px;
  border: none;
  cursor: pointer;
  width: 17%;


  border-radius:4px;
  font-family:Montserrat;
}
button:hover {
  opacity: 0.8;
}
.cancelbtn {
  width: auto;
  padding: 10px 18px;
  background-color: #f44336;
}
.imgcontainer {
  text-align: center;
  margin: 24px 0 12px 0;
}
img.avatar {
  width: 30%;
  border-radius: 50%;
}
.container {
  padding: 16px;
}
span.psw {
  float: right;
  padding-top: 16px;
```

```
  }


/* Change styles for span and cancel button on extra small screens */
  @media screen and (max-width: 300px) {
    span.psw {
       display: block;
       float: none;
    }
    .cancelbtn {
       width: 100%;
    }
  }
  .home{
    margin:80px;
    width: 84%;
    height: 500px;
    padding-top:10px;
    padding-left: 30px;
  }
  .login{
    margin:80px;
    box-sizing: content-box;
    width: 84%;
    height: 420px;
    padding: 30px;
    border: 10px solid rgb(12, 91, 94);
  }



  .left,.right{
   box-sizing: content-box;
   height: 400px;
   margin:20px;
   border: 10px solid rgb(12, 91, 94);
  }
  .mySlides {display: none;}
  img {vertical-align: middle;}
  /* Slideshow container */
```

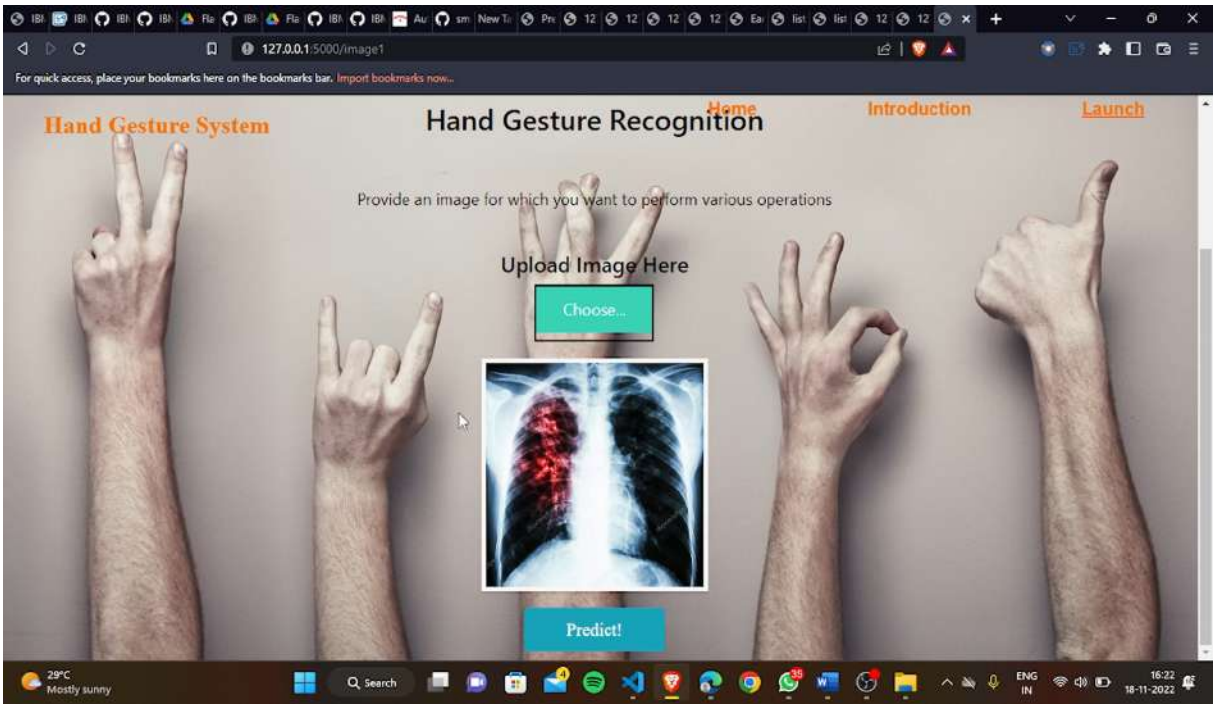
```
.slideshow-container {
  max-width: 1000px;
  position: relative;
  margin: auto;
}
/* Caption text */
.text {
  color: #f2f2f2;
  font-size: 15px;
  padding: 8px 12px;
  position: absolute;
  bottom: 8px;
  width: 100%;
  text-align: center;
}
/* The dots/bullets/indicators */
.dot {
  height: 15px;


 width: 15px;
  margin: 0 2px;
  background-color: #bbb;
  border-radius: 50%;
  display: inline-block;
  transition: background-color 0.6s ease;
}
.active {
  background-color: none;
}
/* Fading animation */
.fade {
  -webkit-animation-name: fade;
  -webkit-animation-duration: 1.5s;
  animation-name: fade;
  animation-duration: 1.5s;
}
@-webkit-keyframes fade {
  from {opacity: .4}
  to {opacity: 1}
```

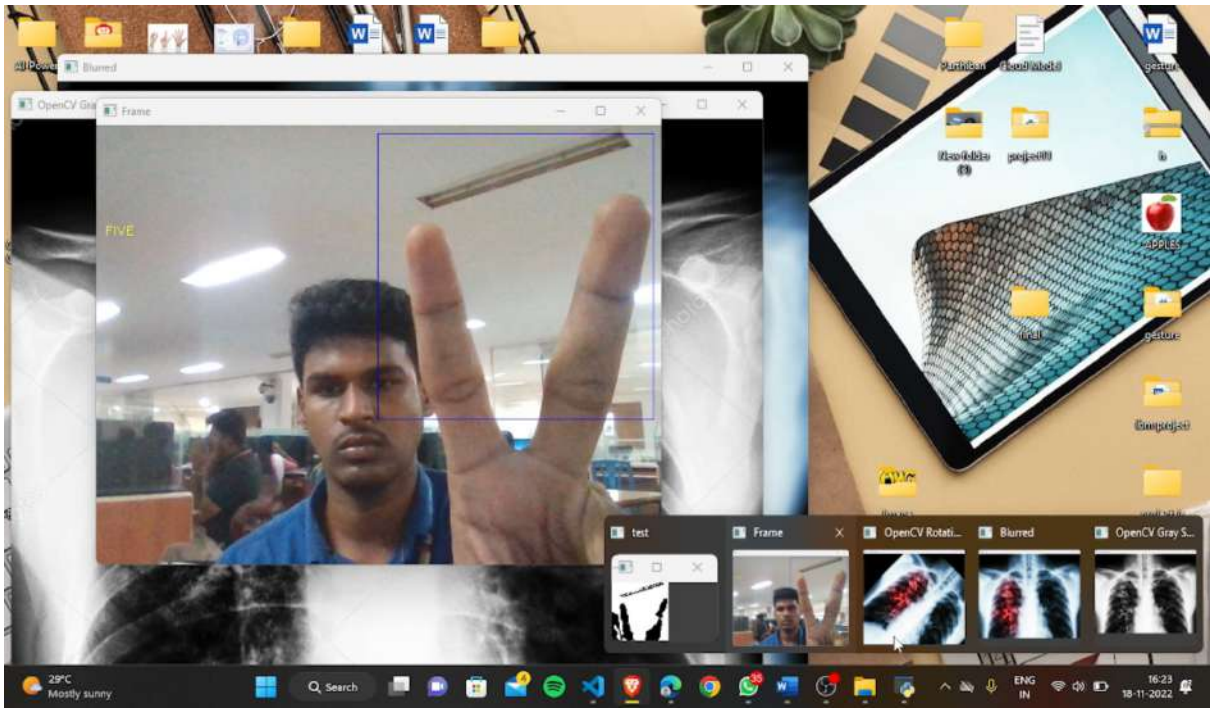
```
  }
  @keyframes fade {
    from {opacity: .4}
    to {opacity: 1}
  }




/* On smaller screens, decrease text size */
  @media only screen and (max-width: 300px) {
    .text {font-size: 11px}
  }
  </style>
  </head>
  <body>
  <div class="header">
  <div style="width:50%;float:left;font-size:2vw;text-align:left;font-weight: bold;
  color:#ff7200; padding-top:1%;padding-left:2%;">Hand Gesture System</div>
   <div class="menu">
     <ul>
      <li><a class="active" href="{{ url_for('home')}}">Home</a></li>
      <li><a class="active" href="{{ url_for('intro')}}">Introduction</a></li>
      <li><a class="active" href="{{ url_for('image1')}}"><u>Launch</u></a></li>
    </ul>
  </div>
  </div>
  <br>
    <div1 style=""><h1><font color="Black" size="6" font-family="Roboto">Hand Gesture
Recognition</h1><br>
    <p><i><font color="Black" size="4" fonr-family="sans-serif"></i>Provide an image for
which you want to perform various operations</p>
    <br>
          <div>
            <h4>Upload Image Here</h4>
        <form action = "http://localhost:5000/" id="upload-file" method="post"
enctype="multipart/form-data">
          <label for="imageUpload" class="upload-label">
            Choose...
          </label>
```

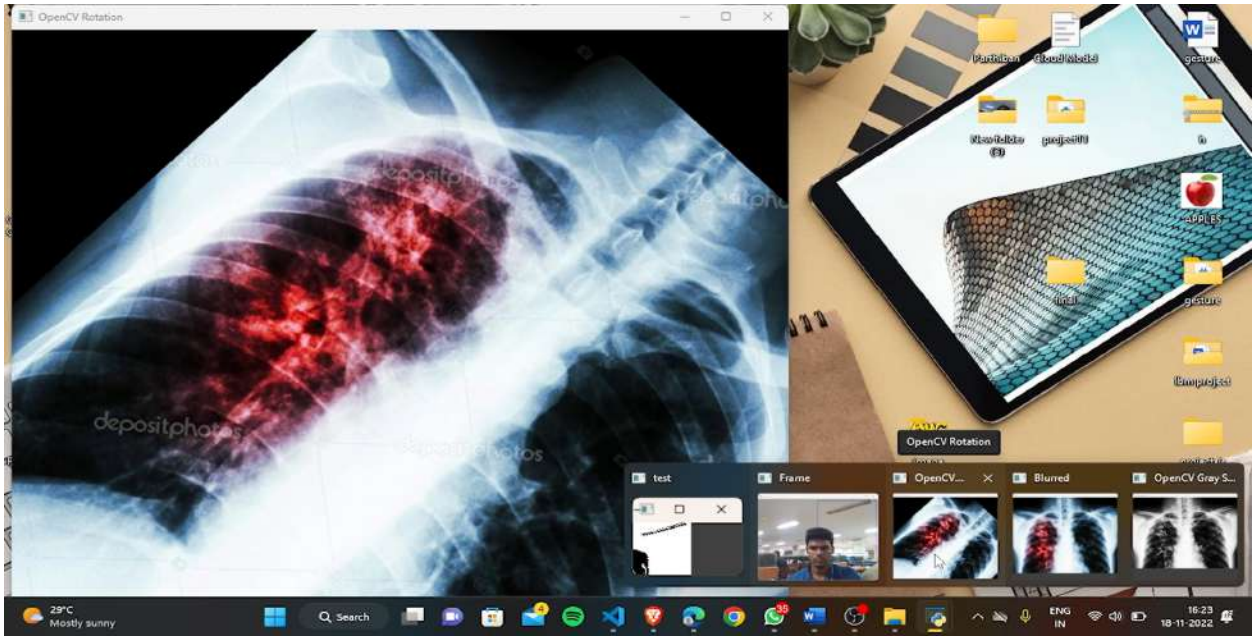
```
      <input type="file" name="image" id="imageUpload" accept=".png, .jpg, .jpeg,.pdf">
    </form>
        <center>
    <div class="image-section" style="display:none;">
      <div class="img-preview">
        <div id="imagePreview">
        </div>
      </div>
      <div>
        <button type="button" class="btn btn-info btn-lg " id="btn-predict">Predict!</button>
      </div>
    </div>
    <div class="loader" style="display:none;"></div>
        </center>
  </div>
  </div1>
    <footer>
  <script src="{{ url_for('static', filename='js/main.js') }}" type="text/javascript"></script>
</footer>
</html>
```

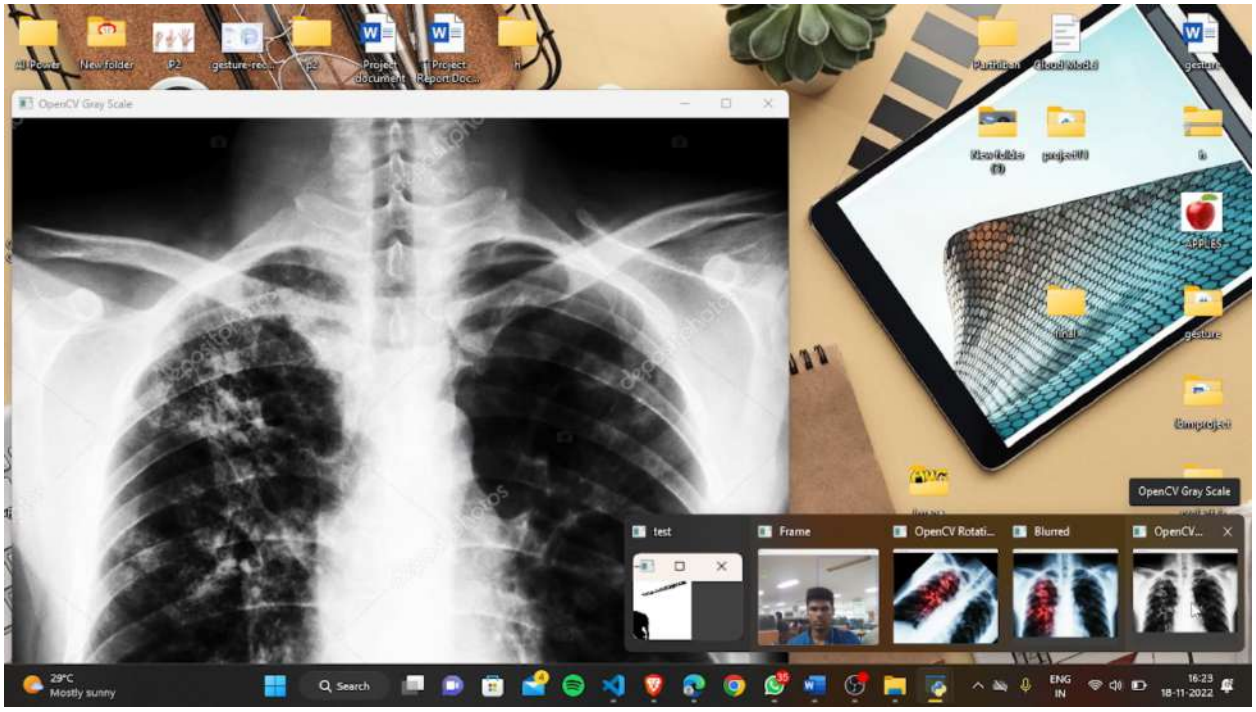
CHAPTER 8

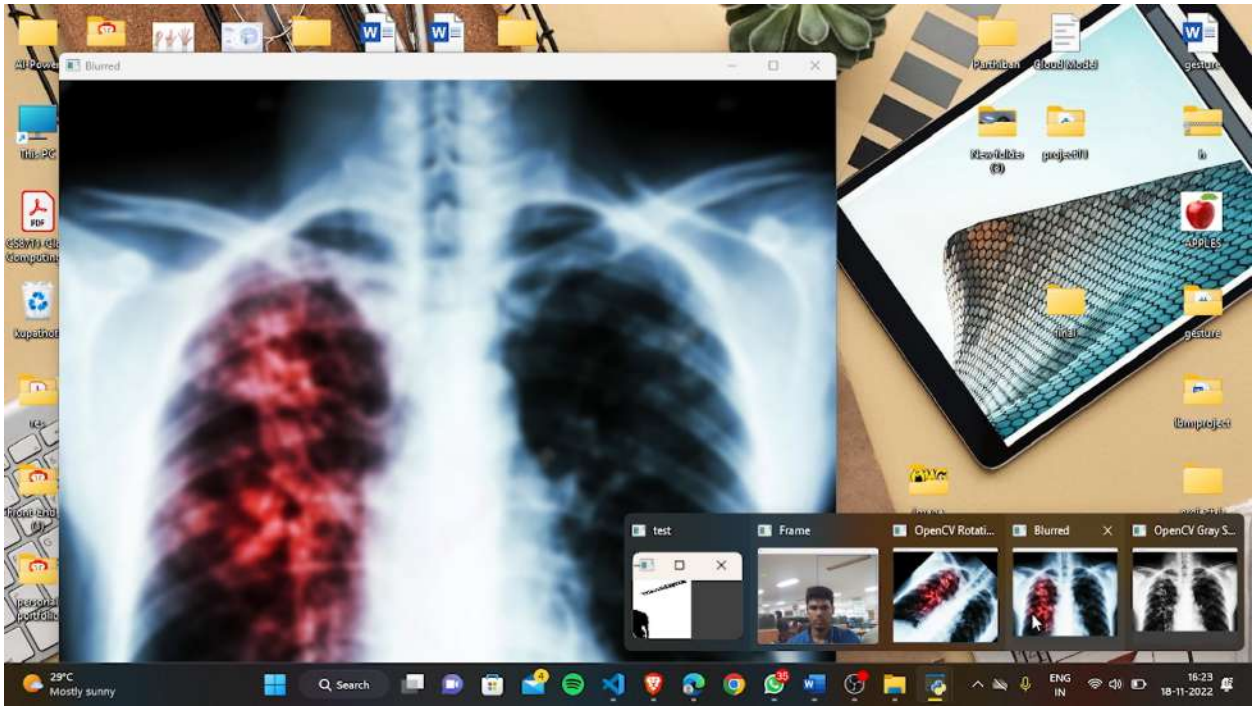
TESTING






OpenCV Rotation
test
Frame
OpenCV...
Blurred
OpenCV Gray S...
OpenCV Rotation
29°C
Mostly sunny
Search
ENG
IN
16:23
18-11-2022


OpenCV Gray Scale
test
Frame
OpenCV Rotati...
Blurred
OpenCV...
OpenCV Gray Scale
29°C
Mostly sunny
Search
ENG
IN
16:23
18-11-2022

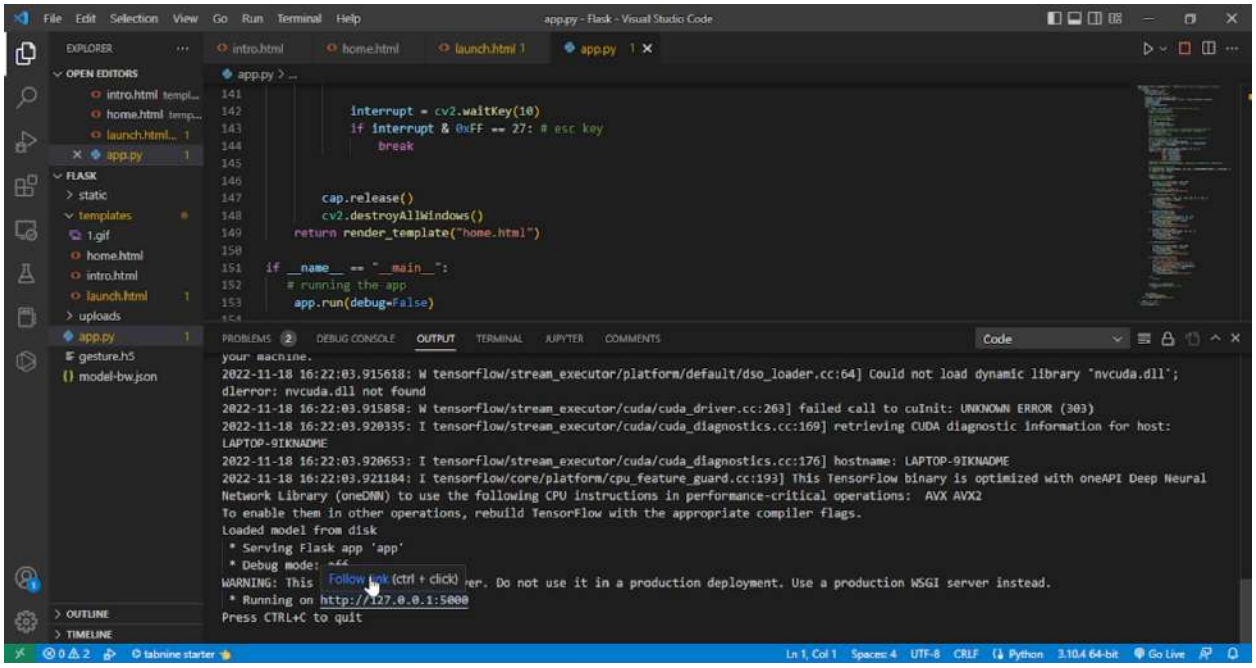

Blurred
test
Frame
OpenCV Rotati...
Blurred
OpenCV Gray S...
29°C
Mostly sunny
Search
ENG
IN
16:23
18-11-2022

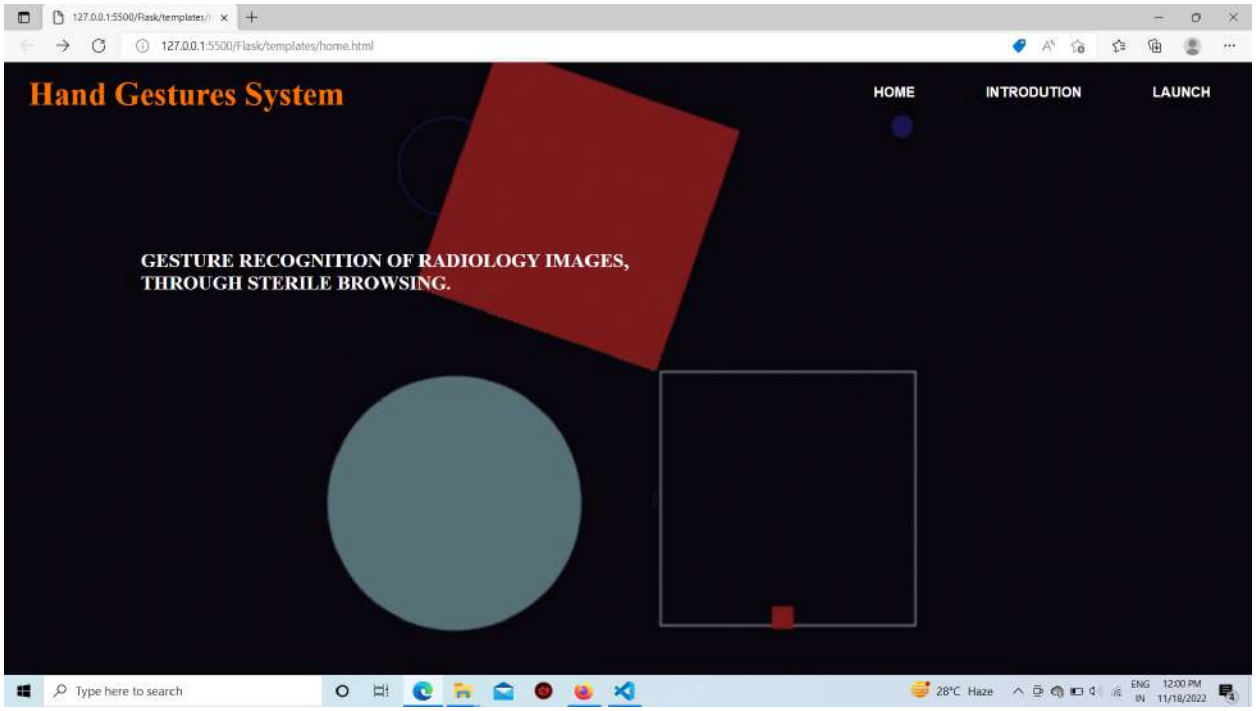
CHAPTER 9

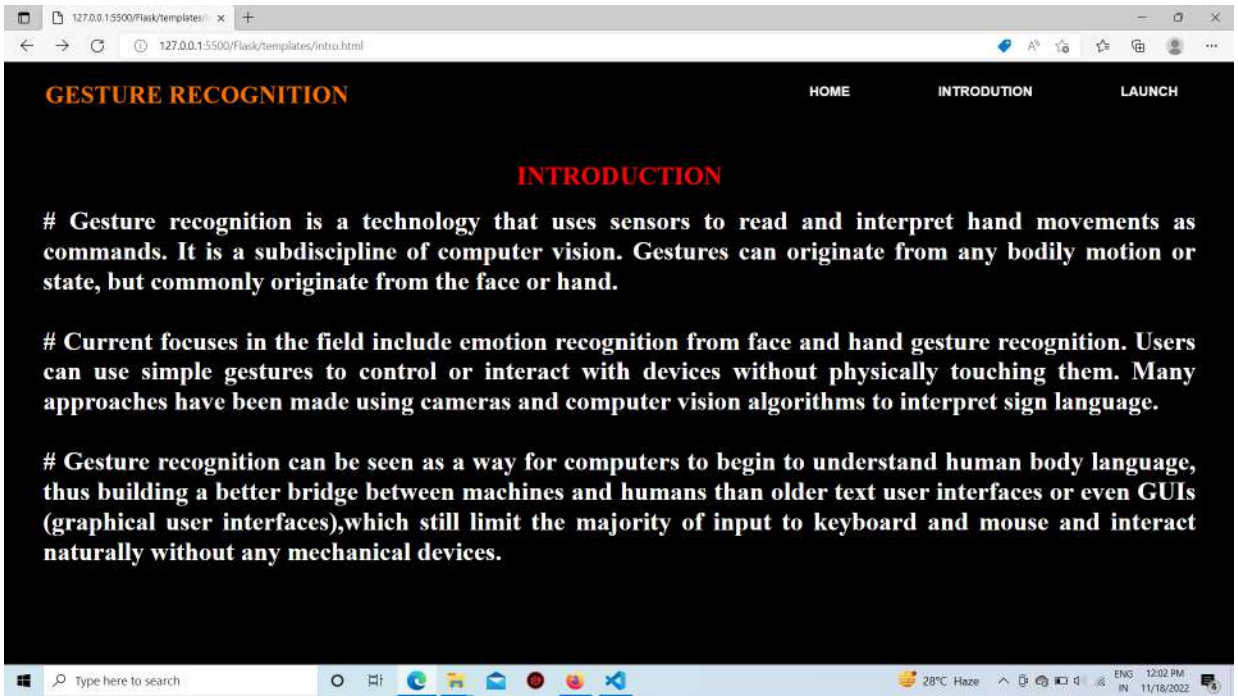
RESULTS

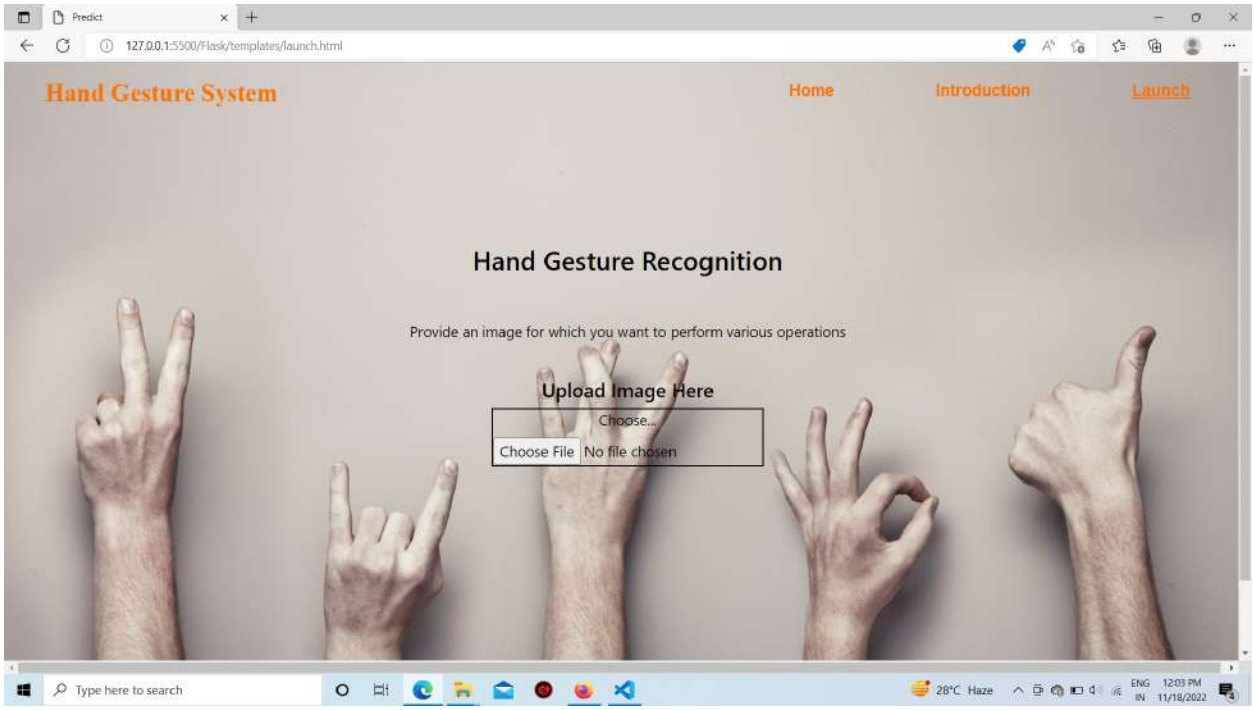
In this project, we found that we can maintain the sterility of an operation theater, etc by using hand based gesture tools to browse the images obtained.

Final findings (Output) of the project along with screenshots as follows.




127.0.0.1:5500/Flask/templates/home.html
Hand Gestures System
HOME
INTRODUTION
LAUNCH
GESTURE RECOGNITION OF RADIOLOGY IMAGES,
THROUGH STERILE BROWSING.
Type here to search
28°C Haze
ENG
IN
12:00 PM
11/18/2022


127.0.0.1:5500/Flask/templates/intro.html
GESTURE RECOGNITION
HOME
INTRODUTION
LAUNCH
INTRODUCTION
Gesture recognition is a technology that uses sensors to read and interpret hand movements as commands. It is a subdiscipline of computer vision. Gestures can originate from any bodily motion or state, but commonly originate from the face or hand.
Current focuses in the field include emotion recognition from face and hand gesture recognition. Users can use simple gestures to control or interact with devices without physically touching them. Many approaches have been made using cameras and computer vision algorithms to interpret sign language.
Gesture recognition can be seen as a way for computers to begin to understand human body language, thus building a better bridge between machines and humans than older text user interfaces or even GUIs (graphical user interfaces),which still limit the majority of input to keyboard and mouse and interact naturally without any mechanical devices.
Type here to search
28°C Haze
ENG
IN
12:02 PM
11/18/2022


Predict
127.0.0.1:5500/Flask/templates/launch.html
Hand Gesture System
Home
Introduction
Launch
Hand Gesture Recognition
Provide an image for which you want to perform various operations
Upload Image Here
Choose...
Choose File
No file chosen
Type here to search
28°C Haze
ENG
IN
12:03 PM
11/18/2022

CHAPTER 10

ADVANTAGES & DISADVANTAGES

Advantages:

- Major advantage of this tool is that it helps to maintain the sterility of the environment.
- It is also easy to use and is quicker than the existing methods to browse images.
- It can also be performed even if the surgeon is a bit far away from the system, this helps t o save time.
- The tool does not need the person using it to have an apparatus or any devices on them to use it.
- They can simply move their hands to browse through the images.

Disadvantages:

- The tool can be quite expensive as it requires cameras and other expensive devices to cap ture images and process it.

CHAPTER 11

CONCLUSION

In this project we developed a tool which recognizes the hand gestures and enables doctors to browse through radiology images using these gestures. This enables doctors and surgeons to maintain the sterility as they would not have to touch any mouse or keyboard to go through the images. This tool is also easy to use and is quicker than the regular method of using mouse/keyboard. It can be used regardless of the users location since they don't have to be in contact with any device. It also does not require the user to have any device on them to use it.Further this technology can be extended to other industries like it can be used by presenters, by teachers for show images in the classroom, etc.

CHAPTER 12

FUTURE SCOPE

- The tool can be made quicker by increasing the recognition speed.
- More number of gestures can be added thereby increasing this tool's functionality and use ability for different purposes.
- Tracking of both hands can be added to increase the set of commands.
- Voice commands can also be added to further increase the functionality.

CHAPTER 13

APPENDIX

Source Code

app.py

```
from flask import Flask,render_template,request
# Flask-It is our framework which we are going to use to run/serve our application.
#request-for accessing file which was uploaded by the user on our application.
import operator
import cv2 # opencv library
import matplotlib.pyplot as plt
import matplotlib.image as mpimg
import numpy as np

from tensorflow.keras.models import load_model#to load our trained model
import os
from werkzeug.utils import secure_filename

app = Flask(__name__,template_folder="templates") # initializing a flask app
# Loading the model
model=load_model('gesture.h5')
print("Loaded model from disk")


@app.route('/')# route to display the home page
def home():
    return render_template('home.html')#rendering the home page


@app.route('/intro') # routes to the intro page
def intro():
    return render_template('intro.html')#rendering the intro page

@app.route('/image1',methods=['GET','POST'])# routes to the index html
def image1():
    return render_template("launch.html")
```

```
@app.route('/predict',methods=['GET', 'POST'])# route to show the predictions in a web UI
def launch():



    if request.method == 'POST':
     print("inside image")
     f = request.files['image']

     basepath = os.path.dirname(__file__)
     file_path = os.path.join(basepath, 'uploads', secure_filename(f.filename))
     f.save(file_path)
     print(file_path)
     cap = cv2.VideoCapture(0)
     while True:
        _, frame = cap.read() #capturing the video frame values
        # Simulating mirror image
        frame = cv2.flip(frame, 1)
        # Got this from collect-data.py
        # Coordinates of the ROI
        x1 = int(0.5*frame.shape[1])
        y1 = 10
        x2 = frame.shape[1]-10
        y2 = int(0.5*frame.shape[1])
        # Drawing the ROI
        # The increment/decrement by 1 is to compensate for the bounding box
        cv2.rectangle(frame, (x1-1, y1-1), (x2+1, y2+1), (255,0,0) ,1)
        # Extracting the ROI
        roi = frame[y1:y2, x1:x2]
        # Resizing the ROI so it can be fed to the model for prediction
        roi = cv2.resize(roi, (64, 64))
        roi = cv2.cvtColor(roi, cv2.COLOR_BGR2GRAY)
        _, test_image = cv2.threshold(roi, 120, 255, cv2.THRESH_BINARY)
        cv2.imshow("test", test_image)
        # Batch of 1
        result = model.predict(test_image.reshape(1, 64, 64, 1))
        prediction = {'ZERO': result[0][0],
                      'ONE': result[0][1],
                      'TWO': result[0][2],
```

```
                'THREE': result[0][3],
                'FOUR': result[0][4],
                'FIVE': result[0][5]}
        # Sorting based on top prediction
        prediction = sorted(prediction.items(), key=operator.itemgetter(1), reverse=True)
        # Displaying the predictions
        cv2.putText(frame, prediction[0][0], (10, 120), cv2.FONT_HERSHEY_PLAIN, 1,
(0,255,255), 1)
        cv2.imshow("Frame", frame)
        #loading an image
        image1=cv2.imread(file_path)




          if prediction[0][0]=='ONE':
           resized = cv2.resize(image1, (200, 200))
           cv2.imshow("Fixed Resizing", resized)
           key=cv2.waitKey(3000)
           if (key & 0xFF) == ord("1"):
              cv2.destroyWindow("Fixed Resizing")
        elif prediction[0][0]=='ZERO':
           cv2.rectangle(image1, (480, 170), (650, 420), (0, 0, 255), 2)
           cv2.imshow("Rectangle", image1)
           cv2.waitKey(0)
           key=cv2.waitKey(3000)
           if (key & 0xFF) == ord("0"):
              cv2.destroyWindow("Rectangle")
        elif prediction[0][0]=='TWO':
           (h, w, d) = image1.shape
           center = (w // 2, h // 2)
           M = cv2.getRotationMatrix2D(center, -45, 1.0)
           rotated = cv2.warpAffine(image1, M, (w, h))
           cv2.imshow("OpenCV Rotation", rotated)
           key=cv2.waitKey(3000)
           if (key & 0xFF) == ord("2"):
              cv2.destroyWindow("OpenCV Rotation")
        elif prediction[0][0]=='THREE':
```

```
            blurred = cv2.GaussianBlur(image1, (21, 21), 0)
            cv2.imshow("Blurred", blurred)
            key=cv2.waitKey(3000)
            if (key & 0xFF) == ord("3"):
                cv2.destroyWindow("Blurred")

        elif prediction[0][0]=='FOUR':
            resized = cv2.resize(image1, (400, 400))
            cv2.imshow("Fixed Resizing", resized)
            key=cv2.waitKey(3000)
            if (key & 0xFF) == ord("4"):
                cv2.destroyWindow("Fixed Resizing")

        elif prediction[0][0]=='FIVE':
            '''(h, w, d) = image1.shape
            center = (w // 2, h // 2)
            M = cv2.getRotationMatrix2D(center, 45, 1.0)
            rotated = cv2.warpAffine(image1, M, (w, h))'''
            gray = cv2.cvtColor(image1, cv2.COLOR_RGB2GRAY)
            cv2.imshow("OpenCV Gray Scale", gray)
            key=cv2.waitKey(3000)




 if (key & 0xFF) == ord("5"):
                cv2.destroyWindow("OpenCV Gray Scale")

        else:
            continue
        interrupt = cv2.waitKey(10)
        if interrupt & 0xFF == 27: # esc key
            break
    cap.release()
    cv2.destroyAllWindows()
  return render_template("home.html")
if __name__ == "__main__":
  # running the app
```

```
app.run(debug=False)
```

GitHub Link :
https://github.com/IBM-EPBL/IBM-Project-52197-1660991160

Project Demo Link:
https://drive.google.com/drive/my-drive