PROJECT REPORT

A GESTURE-BASED TOOL FOR STERILE BROWSING OF RADIOLOGY IMAGES

Submitted by

TEAM ID: PNT2022TMID15252

HEMALATHA.J - (111519104045)

HIBAMARIAM.H - (111519104047)

KALAPOORANI.P - (111519104057)

KALPIKA.K - (111519104058)

R.M.D.ENGINEERING COLLEGE

(An Autonomous Institution)

R.S.M. Nagar, Kavaraipettai-601206

NOVEMBER 2022

TABLE OF CONTENT

INTRODUCTION

1.1 Overview

In this project we use gestures to browse images obtained during radiology. Gestures refer to non verbal forms of communication made using hands.

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 0 - then images is converted into rectangle, 1 - image is Resized into (200,200), 2 - image is rotated by -45°, 3 - image is blurred, 4 - image is Resized into (400,400), 5 - image is converted into grayscale etc.

1.2 Purpose

The main purpose of our project is to create application in which Human can interact with system without any keyboard or mouse to obtain a sterile based interaction only by using Camera. So we develop an application which only need user’s hand sign to give an input through the camera to the system.

LITERATURE SURVEY

This paper's primary goal was to enhance the sterile browsing of radiology images. To avoid difficulties 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. The gesture system relies on real-time robust tracking of the user's hand based in a motion fusion model.

The operation of the gesture interface was tested at the Washington Hospital Centre in Washington, DC. Two operations were observed in the hospital's neurosurgery department and insights regarding the suitability of a hand gesture system was obtained. To our knowledge, this is the first time that a hand gesture recognition system was successfully implemented in an "in vivo" neurosurgical biopsy. A sterile human - machine interface is of supreme importance because it is the means by which the surgeon controls medical information avoiding contamination of the patient, the OR and the surgeon.

2.1 Existing problems

Computing technology has become embedded in every aspect of our daily lives and man–machine interaction is becoming inevitable. It is widely believed that computer and display technology will keep on progressing further. A gateway which allows humans to communicate with machines and computers is known as the human–computer interface (HCI). Keyboard and mouse and touch-screen sensors are the traditional HCI approaches. However, these approaches are becoming a bottleneck for developing user friendly interfaces.

2.2 References

1. Kuno Y, Murashima T, Shimada N, Shirai Y. "Intelligent Wheelchair Remotely Controlledby Interactive Gestures." *Proceedings of 15th International Conference on Pattern Recognition* 2000;
2. Nishikawa A, Hosoi T, Koara K, Negoro D, Hikita A, Asano S, Kakutani H, Miyazaki F, Sekimoto M, Yasui M, Miyake Y, Takiguchi S, Monden M. "Face

MOUSe: A Novel Human-Machine Interface for Controlling the Position of a Laparoscope," *IEEE Trans. on Robotics and Automation* 2003;

3. Smith KR, Frank KJ, Bucholz RD. "The NeuroStation- a highly accurate, minimally invasive solution to frameless stereotatic neurosurgery," *Comput Med Imaging Graph* 1994;
4. Graetzel C, Fong TW, Grange S, Baur C. "A non-contact mouse for surgeon-computerinteraction," *Technol Health Care* 2004.

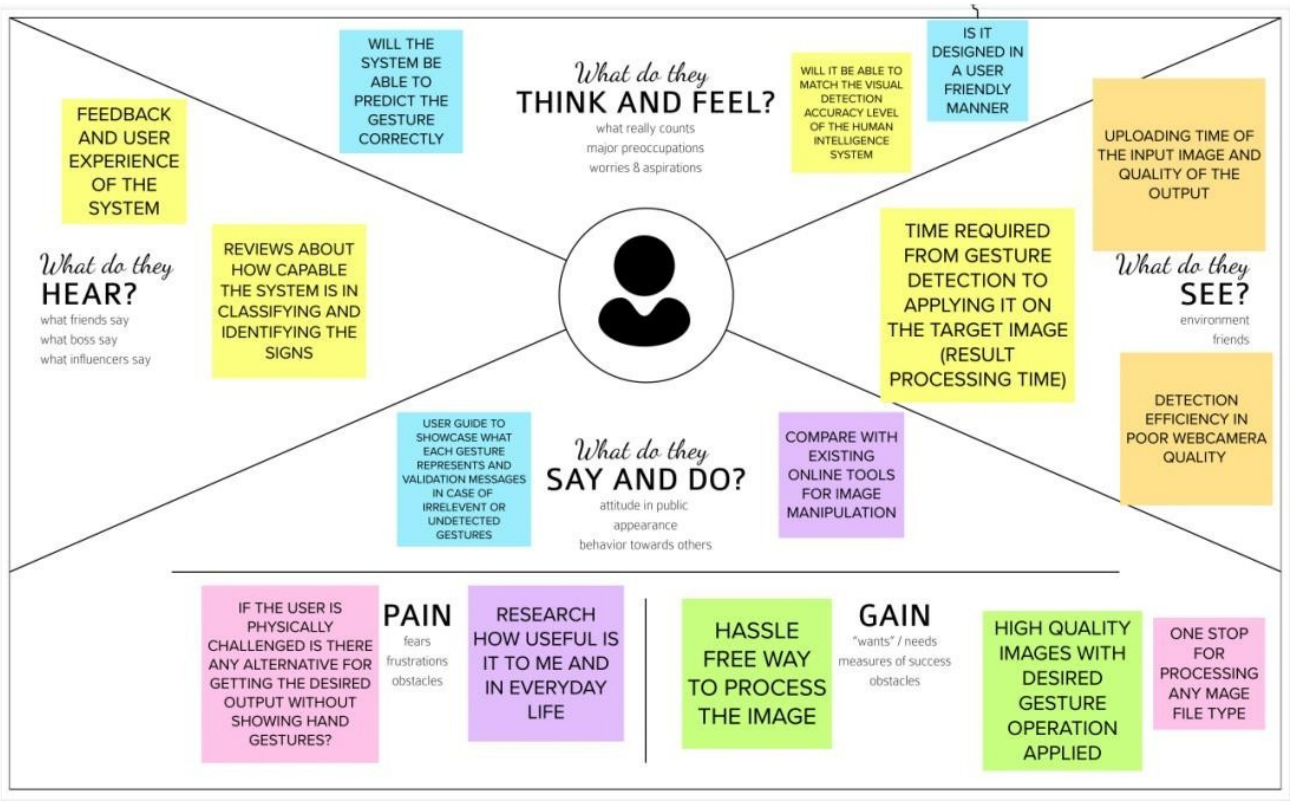
2.3 Problem Statement Definition

- The webcams must be able to recognize the motion gestures which are already trained to the machine.
-
- The user must be able to access the application from anywhere and whenever it's needed.
-
- The user must be able to upload images of various sizes and able to blur, resize and the images must be rotated respectively. This tool must be able to interact with humans and able to understand the symbols.
- The machine must be capable of performing several operations at the same time without any interruptions.

IDEATION & PROPOSED SOLUTION

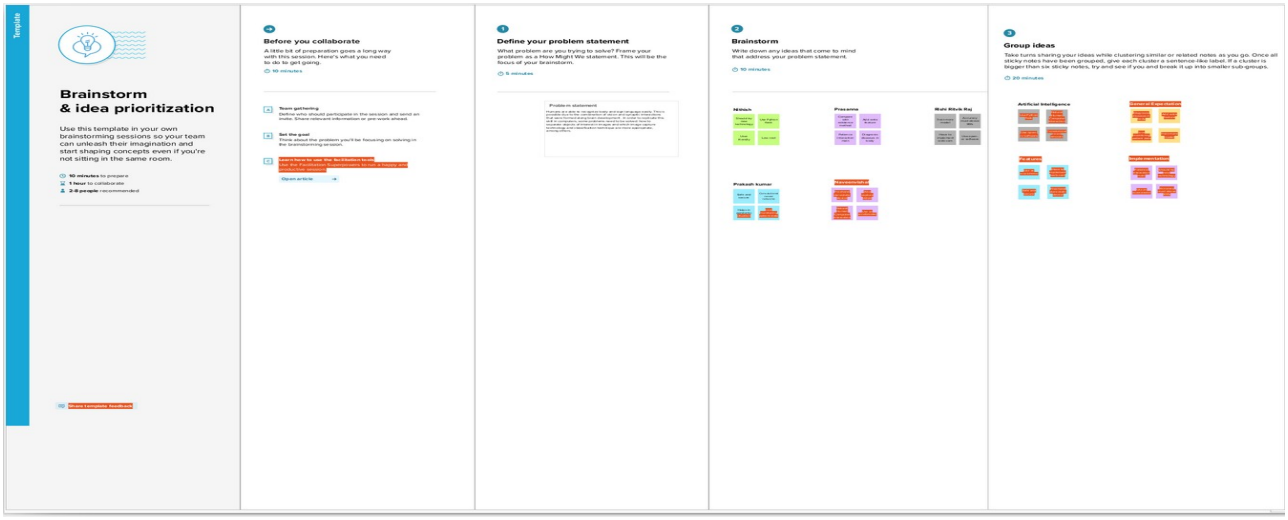
3.1 Empathy Map Canvas:

An empathy map is a simple, easy-to-digest visual that captures knowledge about a user's behaviour 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 the user's perspective along with his or her goals and challenges.



3.2 Ideation and Brainstorming:

Brainstorming provides a free and open environment that encourages everyone within a team to participate in the creative thinking process that leads to problem solving. Prioritizing volume over value, out-of-the-box ideas are welcome and built upon, and all participants are encouraged to collaborate, helping each other develop a rich amount of creative solutions.


Template
Brainstorm
& idea prioritization
Use this template in your own brainstorming sessions so your team can unleash their imagination and start shaping concepts even if you're not sitting in the same room.
10 minutes to prepare
1 hour to collaborate
2-8 people recommended
Before you collaborate
A little bit of preparation goes a long way with this session. Here's what you need to do to get going.
10 minutes
Team gathering
Define who should participate in the session and send an invite. Share relevant information or pre-work ahead.
Set the goal
Think about the problem you'll be focusing on solving in the brainstorming session.
Learn how to use the facilitation tools
Use the Facilitation Superpowers to run a happy and productive session.
Open article
1
Define your problem statement
What problem are you trying to solve? Frame your problem as a How Might We statement. This will be the focus of your brainstorm.
5 minutes
Problem statement
2
Brainstorm
Write down any ideas that come to mind that address your problem statement.
10 minutes
Nitesh
Prasanna
Rishi Ritvik Raj
Prakash kumar
3
Group ideas
Take turns sharing your ideas while clustering similar or related notes as you go. Once all sticky notes have been grouped, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you and break it up into smaller sub-groups.
20 minutes
Artificial Intelligence

–


4
Prioritize
Your team should all be on the same page about what's important moving forward. Place your ideas on this grid to determine which ideas are important and which are feasible.
20 minutes
Gaining more data for training
Accuracy inspite of small input data set
Proper output based on gesture
Working well with low quality images
Less computation time
Secure flask app
Additional image augmentation options
Customization of gestures
Importance
If each of these tasks could get done without any difficulty or cost, which would have the most positive impact?
Feasibility
Regardless of their importance, which tasks are more feasible than others? (Cost, time, effort, complexity, etc.)

3.3 Proposed solution

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	A gesture based tool for sterile browsing of radiology images is introduced to identify the hand gestures and reply accordingly.
2.	Idea / Solution description	Artificial Intelligence and Machine learning techniques are used to identify the hand gestures. The model is trained and the data is tested using image processing and then prediction will be done to perform the following action.
3.	Novelty / Uniqueness	Our model will detect and process the image then different operations are applied on the input.
4.	Social Impact / Customer Satisfaction	Our model is very helpful for the doctors to not touch the radiology images during the entire the operation since everything is done using hand gestures.
5.	Business Model (Revenue Model)	This model has more benefits because the advantage of our model is that it gives accurate solutions for the hospital staff.
6.	Scalability of the Solution	Our system is a real time hand tracking recognition technique which has a scalability of high range and it is very in medical field.

3.3 Problem Solution Fit

REQUIREMENT ANALYSIS

4.1 Functional Requirement

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	Model Training	CNN model trained on well known dataset
FR-2	Model Testing	Testing the model performance on new dataset
FR-3	Application Building	A user friendly application
FR-4	Hand detection	Filtering of hand from image capturing device.

FR-5	Model rendering	When the user uploads the gesture, the algorithm should start processing its task.
FR-6	Reporting	If any issues are faced by the customer or user, it will be directly notified to the developer.

4.2 Non-Functional Requirement

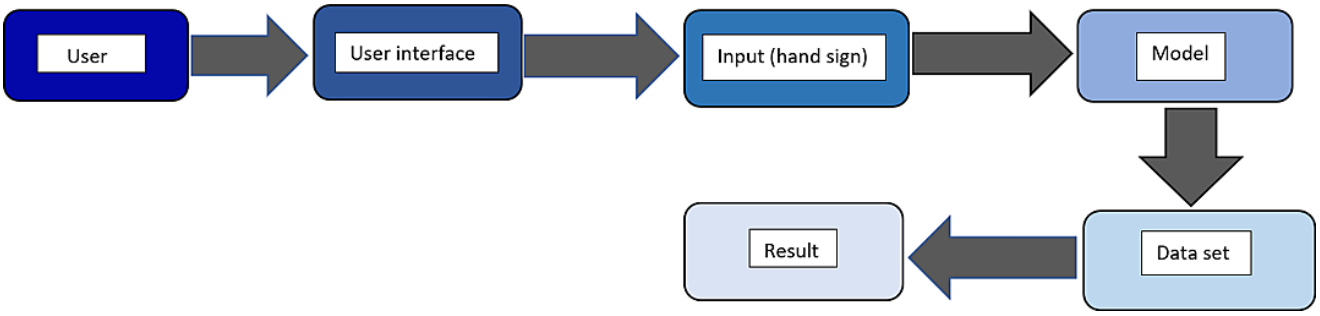
FR No.	Non-Functional Requirement	Description
NFR-1	Usability	It can be used for all users easily and also understandable for non- technical users to communicate with computer via hand sign.
NFR-2	Security	Accessible only in secure networks with administrative permissions, so there is less chanceof security breach.
NFR-3	Reliability	Its operatable under all condition and also, we cancommunicate with computer.
NFR-4	Performance	The performance of the software is high because the speed and accuracy are high. It also upgrades the lifestyle of human beings controlling things viahand signs.
NFR-5	Availability	When the gesture is available then only theapplication works. This application is onlyavailable in surgery rooms.
NFR-6	Scalability	In future we can develop the vehicles that wouldbeing controlled by hand gestures.

PROJECT DESIGN

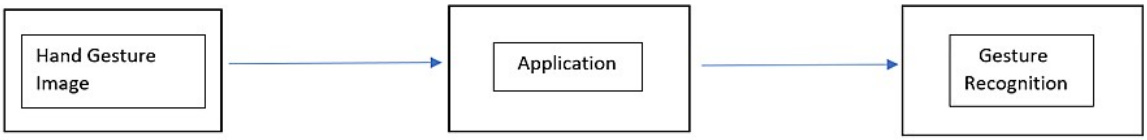
5.1 Data Flow Diagrams

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 requirementgraphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.

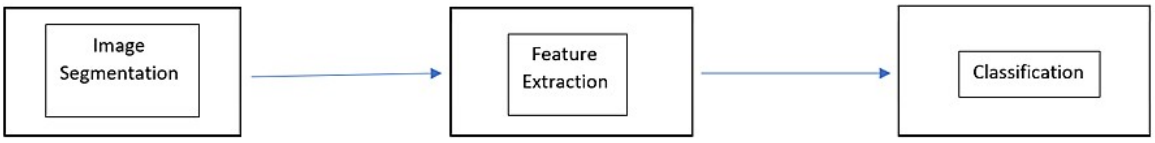
Simplified



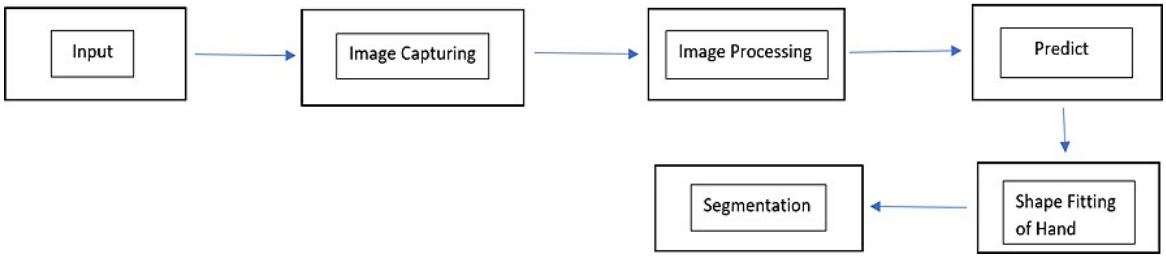
Data Flow Diagram-Level 0



Data Flow Diagram-Level 1



Data Flow Diagram-Level 2

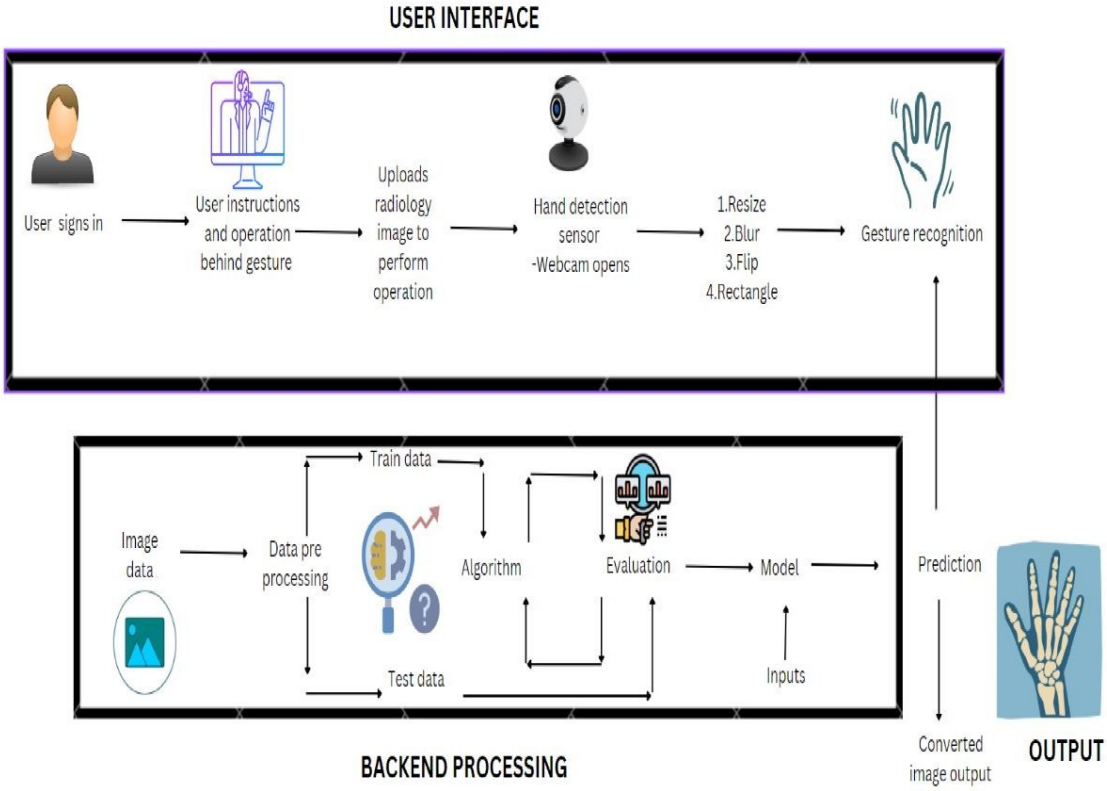


5.2 Solution and Technical Architecture

Solution Architecture:

- User interacts with the UI (User Interface) to upload the image as input.
- Depending on the different gesture inputs different operations are applied to the inputimage.
- The image can be resized, blur, flip and rectangle.
- Once model analyses the gesture, the prediction with operation applied on image isshowcased on the UI.
- Better execution in accurate results, sensitivity, system architecture design andflexibility of the software.

Technology Architecture Diagram:



5.3 User Stories

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority
Customer (Mobile user)	Download the database	USN-1	As a user, I can register for the application by entering my hospital ID and password.	I can access my account / dashboard	High
	Register	USN-2	As a user, I can register for the application by entering my hospital ID and password.	I can receive confirmation link.	High
	Login	USN-3	As a user, I will have to get confirmation from the radiology department to access the application.	I can register & access the dashboard with my hospital ID.	Low
	Upload the image	USN-4	As a user, I must upload the image to execute the command		High
Customer (Web user)	The functional requirements are same as mobile user	Same as mobile user	Same as mobile user	Same as mobile user	High when compared to mobile users

PROJECT PLANNING AND SCHEDULING

6.1 Sprint Planning & Estimation

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	**Data Collection**	USN-1	Download the Dataset	20	Low	Kalapoorani P
Sprint-1		USN-2	Image Pre- processing(Import the library, Image preprocessing, Configure ImageDataGenerator, Apply image generator functionality)	20	High	Kalpika K
Sprint-2	**Model Building**	USN-6	Import the model building libraries and initialize the model	10	Low	Hiba Mariam H
Sprint-2		USN-7	Adding CNN layers and Dense layers	10	High	Kalapoorani P
Sprint-2		USN-8	Configure the learning process	10	High	Hiba Mariam H

Sprint-2		USN-9	Train test and save the model	10	High	Hemalatha J
Sprint-3	Website Building	USN-3	Create HTML pages	10	Mediu m	Kalpika K
Sprint-3		USN-4	Build Python code	20	High	Hemalatha J
Sprint-3		USN-5	Run the application	10	High	Kalapoorani P
Sprint-4	Train The Model on IBM	USN-11	Register for IBM Cloud	20	High	Hiba Mariam H
Sprint-4		USN-12	Train the Model and Test the Model and its Overall Performance	20	High	Kalapoorani P

6.2 Sprint Delivery Schedule

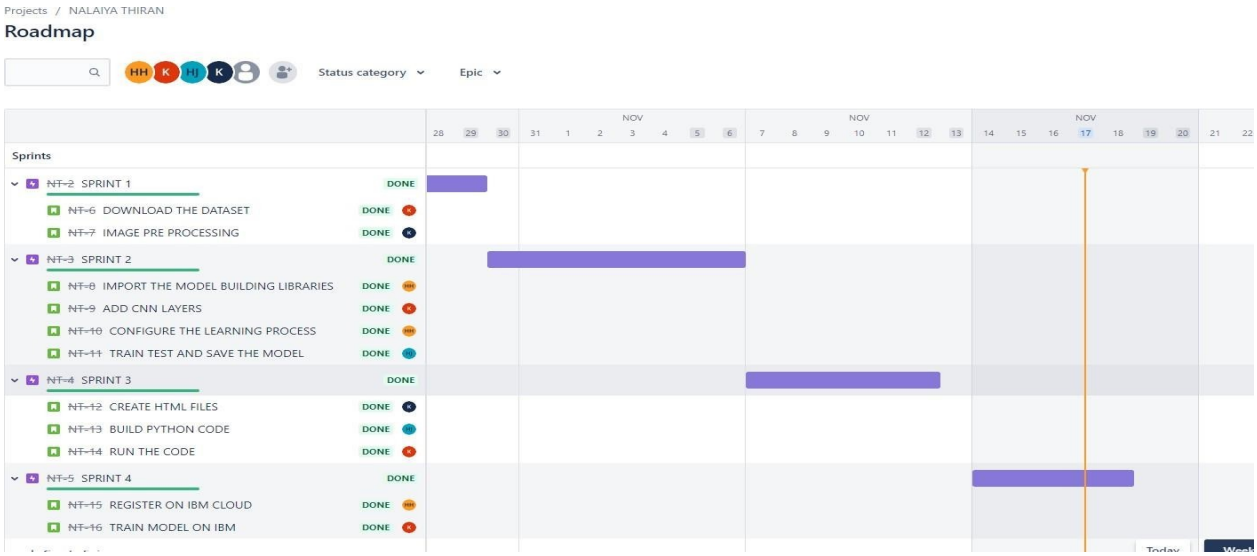
Sprint	**Total Story Points**	**Duration**	**Sprint Start Date**	**Sprint End Date (Planned)**	**Story Points Completed (as on Planned End Date)**	**Sprint Release Date(Actual)**
Sprint-1	40	6 Days	24 Oct 2022	29 Oct 2022	40	29 Oct 2022
Sprint-2	40	6 Days	31 Oct 2022	05 Nov 2022	40	05 Nov 2022
Sprint-3	40	6 Days	07 Nov 2022	12 Nov 2022	40	12 Nov 2022
Sprint-4	40	6 Days	14 Nov 2022	18 Nov 2022	40	19 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) periteration unit (story points per day)

$$AV = \frac{\text{SPRINT DURATION}}{\text{VELOCITY}} = \frac{12}{6} = 2$$

Roadmap:



CODING AND SOLUTION

7.1 Features

Model Building:

Convolution Neural Network (CNN)
Deep learning is a very significant subset of machine learning because of its high performance across various domains. Convolution Neural Network (CNN), is a powerful image processing deep learning type often using in computer vision that comprises an image and video recognition along with a recommender system.

CNN uses a multilayer system consists of the input layer, output layer, and a hidden layer that comprises multiple convolutional layers, pooling layers, fully connected layers.

Model Training:

Training a neural network typically consists of two phases:

A forward phase, where the input is passed completely through the network.

A backward phase, where gradients are backpropagated (backprop) and weights are updated.

We'll follow this pattern to train our CNN. There are also two major implementation-specific ideas we'll use:

During the forward phase, each layer will cache any data (like inputs, intermediate values, etc) it'll need for the backward phase. This means that any backward phase must be preceded by a corresponding forward phase.

During the backward phase, each layer will receive a gradient and also return a gradient. It will receive the gradient of loss with respect to its outputs ($\partial L / \partial out$) and return the gradient of loss with respect to its inputs ($\partial L / \partial in$).

These two ideas will help keep our training implementation clean and organized.

TESTING

8.1 Test Cases

Test case ID	Feature Type	Component	Test Scenario	Pre-Requisite	Steps To Execute	Test Data	Expected Result	Actual Result	Status	Comments	TC for Automation(Y/N)	BUG ID
Test Case - 01	Functional	web Page	Verify user is able to see the page popup when user they enter to web page		1.Enter URL and click go		webpage should display	Working as expected	Pass			
Test Case - 02	UI	Home Page	Verify the UI elements in Hompage		1.Enter URL and click go 2.webpage displayed		Application should show below UI elements: a.Hompage b.Introduction c.Launch	Working as expected	Pass			
Test Case - 03	Functional	Home page	Verify user is able to See the deatils about the webpage		1.Enter URL and click go 2.Webpage Displayed 3.Displays necessary details		User should navigate to homepage	Working as expected	Pass			
Test Case - 04	Functional	Introducti on	Verify user is able to details about uses of the gesture based tool and its importance		1.Enter URL and click go 2.Webpage Displayed 3.Displays necessary details 4.click introduction to go Displays about the uses of the gesture based tool		user should navigate to introduction	Working as expected	Pass			
Test Case - 05	Functional	Launch	Verify user is able to navigate to launch		1.Enter URL and click go 2.Webpage Displayed 3.Displays necessary details 4.click introduction to go Displays about the uses of the gesture based tool 5.Click Launch to navigates to launch page		user should navigate to Launch	Working as expected	Pass			
Test Case - 06	Functional	Launch	Verify user is able to upload image and predicts using the hand gesture		1.Enter URL and click go 2.Webpage Displayed 3.Displays necessary details 4.click introduction to go Displays about the uses of the gesture based tool 5.Click Launch to navigates to launch page 6.uploads the image and predicts using the hand gesture	Upload image : img1.jpg	Application should Display the images in different types of images like blurred , rotated image	Working as expected	Pass			

8.2 User Acceptance Testing

PURPOSE OF THE DOCUMENT

The purpose of this document is to briefly explain the test coverage and open issuesof the Handwritten Digit Recognition project at the time of the release to User Acceptance Testing (UAT).

DEFECT ANALYSIS

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Total
By Design	1	0	1	0	2
Duplicate	0	0	0	0	0
External	0	0	2	0	2
Fixed	4	1	0	1	6
Not Reproduced	0	0	0	1	1
Skipped	0	0	0	1	1
Won't Fix	1	0	1	0	2
Total	6	1	4	3	14

TEST CASE ANALYSIS

Section	Total Cases	Not Tested	Fail	Pass
Client Application	10	0	3	7
Security	2	0	1	1
Performance	3	0	1	2
Exception Reporting	2	0	0	2

RESULT

9.1 Performance Metrics

Application Performance

Locust Test Report

During: 11/14/2022, 10:54:06 AM - 11/14/2022, 10:56:49 AM

Target Host: http://127.0.0.1:5000

Script: locust.py

Request Statistics

Method	Name	# Requests	# Fails	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	RPS	Failures/s
GET	/	151	0	7	3	18	6975	0.9	0.0
GET	/image1	156	0	6	3	25	7090	1.0	0.0
GET	/intro	159	0	6	3	18	8317	1.0	0.0
GET	/predict	42	4	15431	2982	95299	6335	0.3	0.0
	Aggregated	**508**	**4**	**1281**	**3**	**95299**	**7377**	**3.1**	**0.0**

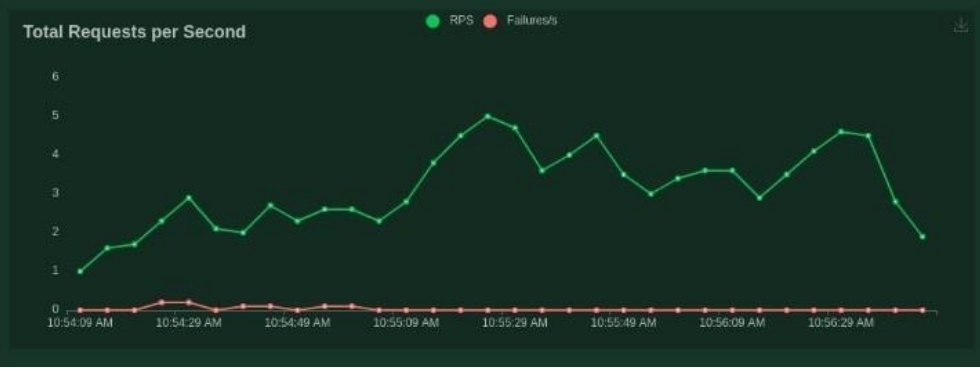
Response Time Statistics

Method	Name	50%ile (ms)	60%ile (ms)	70%ile (ms)	80%ile (ms)	90%ile (ms)	95%ile (ms)	99%ile (ms)	100%ile (ms)
GET	/	6	7	9	10	11	12	14	18
GET	/image1	5	6	7	8	10	11	16	26
GET	/intro	6	6	7	9	11	11	18	18
GET	/predict	7500	11000	17000	21000	40000	49000	95000	95000
	Aggregated	**6**	**7**	**9**	**10**	**13**	**6300**	**31000**	**95000**

Failures Statistics

Method	Name	Error	Occurrences
GET	/predict	500 Server Error: INTERNAL SERVER ERROR for url: http://127.0.0.1:5000/predict	4

Charts




Final ratio
Ratio per User class
100.0% AppUser
25.0% home
25.0% intro
25.0% image1
25.0% predict
Total ratio
100.0% AppUser
25.0% home
25.0% intro
25.0% image1
25.0% predict

Model Performance:

Confusion Matrix

Confusion Matrix

In [10]:
```
from sklearn.metrics import classification_report
```

In [11]:
```
path = "Dataset/test/"
test_images_path = []
labels = []
for directory in os.listdir(path):
    for img_name in os.listdir(path+directory ):
        test_images_path.append(path+directory+"/"+img_name)
        labels.append(int(directory))
```

In [19]:
```
predicted_list = []
for img_path in test_images_path:
    img = image.load_img(img_path,grayscale=True,target_size=(64,64))
    x = image.img_to_array(img)
    x = np.expand_dims(x,axis=0)
    pred = np.argmax(model.predict(x),axis=1)
    predicted_list.append(pred)
```

In [17]:
```
matrix = tf.math.confusion_matrix(labels,predicted_list)
print("The Confusion Matrix")
cm=pd.DataFrame(matrix,index=['gesture-0','gesture-1','gesture-2','gesture-3','gesture-4','gesture-5'],
                columns=['gesture-0','gesture-1','gesture-2','gesture-3','gesture-4','gesture-5'])
print(cm)
```

```
The Confusion Matrix
           gesture-0  gesture-1  gesture-2  gesture-3  gesture-4  gesture-5
gesture-0          5          0          0          0          0          0
gesture-1          0          5          0          0          0          0
gesture-2          0          1          4          0          0          0
gesture-3          0          0          0          5          0          0
gesture-4          0          0          0          2          3          0
gesture-5          0          0          0          0          0          5
```

Classification report:

Classification Report

In [66]:
```
print(classification_report(labels, predicted_list))
```

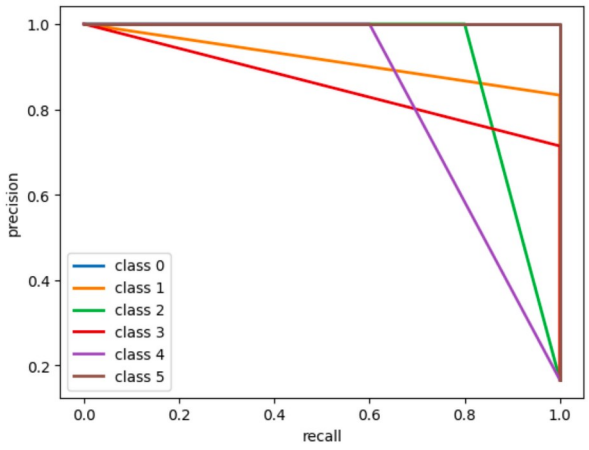
```
              precision    recall  f1-score   support

           0       1.00      1.00      1.00         5
           1       0.83      1.00      0.91         5
           2       1.00      0.80      0.89         5
           3       0.71      1.00      0.83         5
           4       1.00      0.60      0.75         5
           5       1.00      1.00      1.00         5

    accuracy                           0.90        30
   macro avg       0.92      0.90      0.90        30
weighted avg       0.92      0.90      0.90        30
```

Precision Recall Graph:

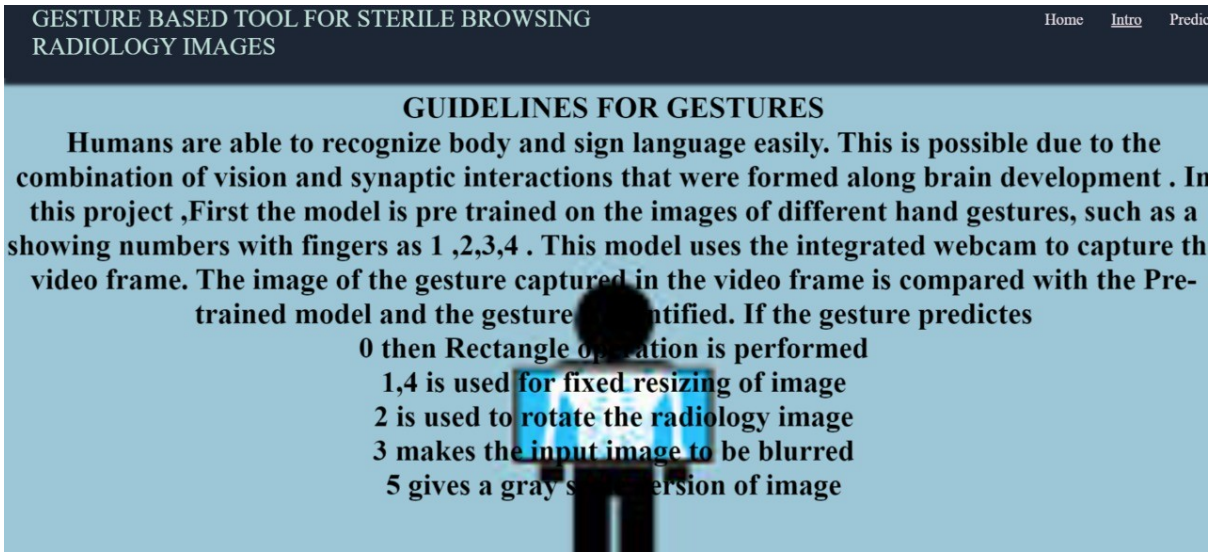
Precision Recall Graph

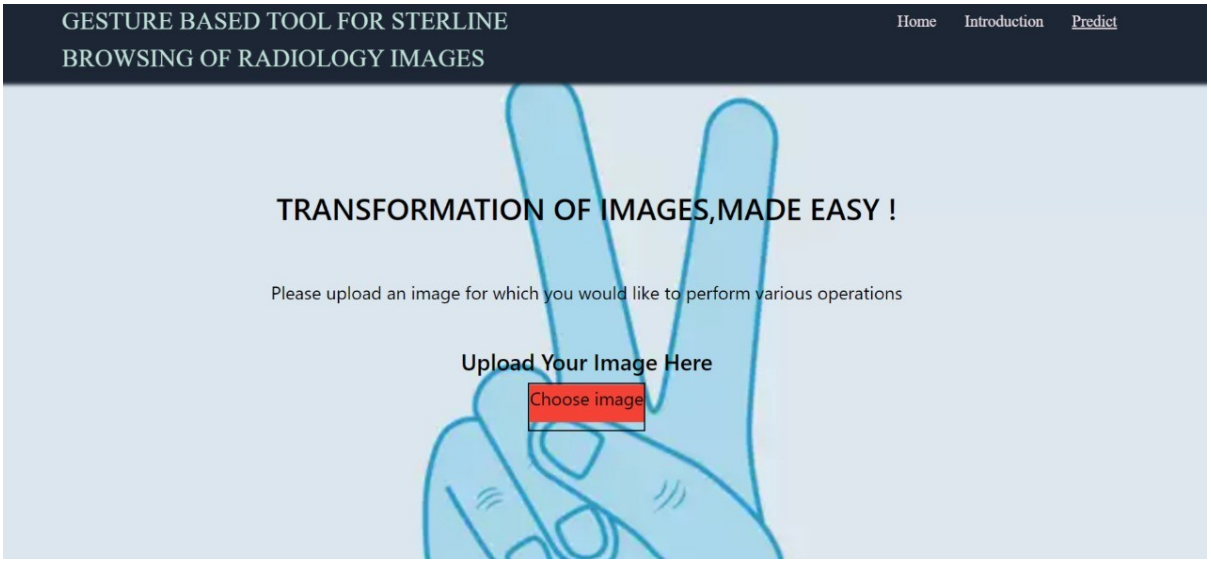
In [123]:
```
n_classes = 6
precision = dict()
recall = dict()
for i in range(n_classes):
    precision[i], recall[i], _ = precision_recall_curve(y_test[:, :,i],
                                                        predicted_prob[:, :,i])
    plt.plot(recall[i], precision[i], lw=2, label='class {}'.format(i))
plt.xlabel("recall")
plt.ylabel("precision")
plt.legend()
```

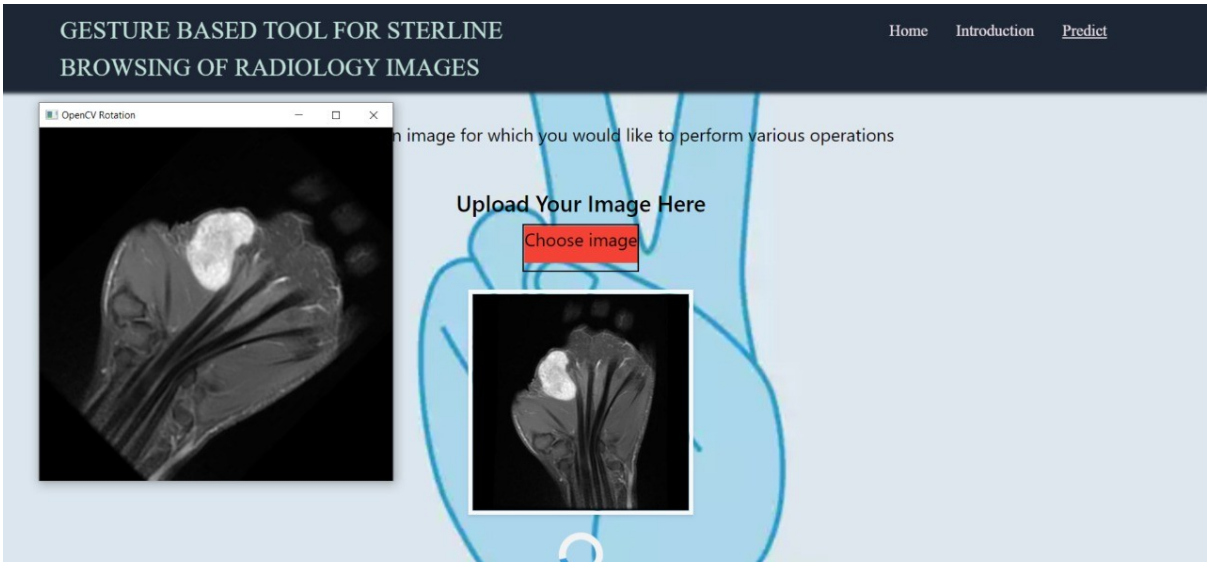
Out[123]: <matplotlib.legend.Legend at 0x7fc50e3566d0>



Output:


GESTURE BASED TOOL FOR STERILE BROWSING
RADIOLOGY IMAGES
Home
Intro
GUIDELINES FOR GESTURES
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
this project ,First the model is 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 th
video frame. The image of the gesture captured in the video frame is compared with the Pre-
trained model and the gesture
0 then Rectangle
1,4 is used for fixed resizing of image
2 is used to rotate the radiology image
3 makes the input image to be blurred
5 gives a gray


GESTURE BASED TOOL FOR STERLINE
BROWSING OF RADIOLOGY IMAGES
Home
Introduction
Predict
TRANSFORMATION OF IMAGES,MADE EASY !
Please upload an image for which you would like to perform various operations
Upload Your Image Here
Choose image


GESTURE BASED TOOL FOR STERLINE
BROWSING OF RADIOLOGY IMAGES
Home
Introduction
Predict
OpenCV Rotation
image for which you would like to perform various operations
Upload Your Image Here
Choose image

ADVANTAGES & DISADVANTAGES

Advantages:

- Major advantage of this tool is that it helps to maintain the sterility of theenvironment.
- It is also easy to use and is quicker than the existing methods to browseimages.
- It can also be performed even if the surgeon is a bit far away from thesystem, this helps to save time.
- The tool does not need the person using it to have an apparatus or anydevices 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 expensivedevices to capture images and process it.
- Such systems are difficult to develop because of the complexity and the costof implementation.
- As each gesture is assigned a specific control command, this system is not platform independent since certain control commands vary as the operatingsystem varies.

CONCLUSION

So, the main objective of our Project is to create an application in which the user uses their hands to give input to the system to browse Images. This is performed in order to achieve Sterile Browsing. This enables doctors and surgeons to maintain the sterility as they would not have to touch anymouse 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 regardlessof 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. The application is built in Industrial method by following Agile methodology to understand the basic concepts of Agile methodology, and programming concepts of Python, Tensorflow. And also train and implement the model on IBM Cloud etc. Further, this technology can be extended to other industries like it can be used by presenters, byteachers for show images in the classroom, etc

FUTURE SCOPE

1. In future, certain disadvantages can be removed, and achieve full Sterile Browsing.
2.Might be implemented not only for sterile browsing but also for other applications.
3.Can be implemented on AR/VR 3D Tracking Technologies.
4.The tool can be made quicker by increasing the recognition speed
5.More number of gestures can be added thereby increasing this tool'sfunctionality and useability for different purposes.
6.Tracking of both hands can be added to increase the set of commands
7.Voice commands can also be added to further increase the functionality

APPENDIX

Source Code:

7.1 MODEL BUILDING:

Training Model:

```
# Installing Required packages
pip install tensorflow
pip install numpy
pip install flask

# Impporting Required Packages
import numpy as np
import tensorflow
from tensorflow.keras.models import Sequential
from tensorflow.keras import layers
from tensorflow.keras.layers import Dense,Flatten
from tensorflow.keras.layers import Conv2D,MaxPooling2D
from keras.preprocessing.image import ImageDataGenerator

# Image Processing
train_datagen=ImageDataGenerator(rescale=1./255,shear_range=0.2,zoom
_range=0.2,horizontal_flip=True)
test_datagen=ImageDataGenerator(rescale=1./255)
# In[3]:
x_train = train_datagen.flow_from_directory(r'dataset/train',
                                            target_size=(64, 64),
                                            batch_size=5,
                                            color_mode='grayscale',
class_mode='categorical')
x_test = test_datagen.flow_from_directory(r'dataset/test',
                                          target_size=(64, 64),
                                          batch_size=5,
                                          color_mode='grayscale',
                                          class_mode='categorical')

# Model Building
model=Sequential()
# Adding Layers
model.add(Conv2D(32,(3,3),input_shape=(64,64,1),activation='relu'))
model.add(MaxPooling2D(pool_size=(2,2)))
```

```
model.add(Conv2D(32,(3,3),activation='relu'))
model.add(MaxPooling2D(pool_size=(2,2)))
model.add(Flatten())


# Adding Dense Layers
model.add(Dense(units=128,activation='relu'))
model.add(Dense(units=6,activation='softmax'))

model.summary()
model.compile(optimizer='adam',loss='categorical_crossentropy',metri
cs=['accuracy'])

# Model Training
model.fit_generator(x_train,
                    steps_per_epoch = len(x_train) ,
                    epochs = 20,
                    validation_data = x_test,
                    validation_steps = len(x_test) )
# Save the model

model.save('gesture.h5')
```

Testing Model:

```
model_json = model.to_json()
with open('model-bw.json',"w") as json_file:
    json_file.write(model_json)

model = load_model('gesture.h5')

img =
image.load_img('Dataset/test/2/3.jpg',grayscale=True,target_size=(64,64))

x = image.img_to_array(img)

x = np.expand_dims(x,axis=0)
pred = np.argmax(model.predict(x),axis=1)

index = ['0','1','2','3','4','5']

res = str(index[pred[0]])
```

Application Building:

Home.html

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


<style>
.header {  position: relative;
                top:0;
                margin:0px;
                z-index: 1;
                left: 0px;
                right: 0px;
                position: fixed;
                background-color:#1d2634 ;
                color: rgb(181, 228, 236);
                box-shadow: 0px 8px 4px #1d2634;
                overflow: hidden;
                padding-left:5px;
                font-family: 'Times New Roman';
                font-size: 2vw;
                width: 100%;
                height:15%;
                text-align: center;
           }
    /* rgb(10, 102, 109)  */
           .topnav {
  overflow: hidden;
  background-color: #84d39e;
}

.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: #1d2634;
  color: rgb(238, 226, 234);
}
.topnav-right a:hover {
```

```
  background-color: #263043;
  color: white;
}

.topnav-right {
  float: right;
  padding-right:85px;
  /* padding-top: 550px; */
}

body {

  background-color: #263043 ;
  background-repeat: no-repeat;
  background-size:cover;
  background-position: 0px 0px;
  }
  .button {
  background-color: #091425;
  border: none;
  color: rgb(181, 228, 236);
  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: rgb(181, 228, 236);
  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: rgb(181, 228, 236);
}

.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(13, 53, 68);
}
.left,.right{
 box-sizing: content-box;
 height: 400px;
 margin:20px;
 border: 10px solid rgb(13, 53, 68);
}

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

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

/* Caption text */
.text {
  color: #9ac0c0;
  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 {
  color: rgb(145, 216, 221);
}

/* 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=Luckiest+Guy");
/* BODY */
body {
  position: absolute;
  top: 0;
  left: 0;
  right: 0;
  bottom: 0;
  width: 100%;
  height: 100%;
```

```
  overflow: hidden;
  font-family: "Arial", cursive;
  -webkit-font-smoothing: antialiased;
}

::selection {
  background: transparent;
}
/* CLOUDS */
body:before {
  content: "";
  position: absolute;
  top: 0;
  left: 0;
  right: 0;
  width: 0;
  height: 0;
  margin: auto;
  border-radius: 100%;
  background: transparent;
  display: block;
  box-shadow: 0 0 150px 100px rgba(255, 255, 255, 0.6),
    200px 0 200px 150px rgba(255, 255, 255, 0.6),
    -250px 0 300px 150px rgba(255, 255, 255, 0.6),
    550px 0 300px 200px rgba(255, 255, 255, 0.6),
    -550px 0 300px 200px rgba(255, 255, 255, 0.6);
}
/* JUMP */
h1 {
  cursor: default;
  position: absolute;
  top: 0;
  left: 0;
  right: 0;
  bottom: 0;
  width: 100%;
  height: 100px;
  margin: 70px;
  display: block;
  text-align: center;
}

h1 span {
  position: relative;
  top: 0px;
  display: inline-block;
  font-size: 25px;
  color: white;
```

```
  /* text-shadow: 0 1px 0 rgb(151, 201, 197), 0 2px 0 rgb(151, 201,
197), 0 3px 0 rgb(151, 201, 197), 0 4px 0 rgb(151, 201, 197),
    0 5px 0 rgb(151, 201, 197), 0 6px 0 transparent, 0 7px 0
transparent, 0 8px 0 transparent,
    0 9px 0 transparent, 0 10px 10px rgba(58, 159, 167, 0.4); */
}

h1 span:nth-child(2) {
  -webkit-animation-delay: 0.1s;
}

h1 span:nth-child(3) {
  -webkit-animation-delay: 0.2s;
}

h1 span:nth-child(4) {
  -webkit-animation-delay: 0.3s;
}

h1 span:nth-child(5) {
  -webkit-animation-delay: 0.4s;
}

h1 span:nth-child(6) {
  -webkit-animation-delay: 0.5s;
}

h1 span:nth-child(7) {
  -webkit-animation-delay: 0.6s;
}

h1 span:nth-child(8) {
  -webkit-animation-delay: 0.2s;
}

h1 span:nth-child(9) {
  -webkit-animation-delay: 0.3s;
}

h1 span:nth-child(10) {
  -webkit-animation-delay: 0.4s;
}

h1 span:nth-child(11) {
  -webkit-animation-delay: 0.5s;
}

h1 span:nth-child(12) {
```

```
  -webkit-animation-delay: 0.6s;
}

h1 span:nth-child(13) {
  -webkit-animation-delay: 0.7s;
}

h1 span:nth-child(14) {
  -webkit-animation-delay: 0.8s;
}

/* ANIMATION */
@-webkit-keyframes bounce {
  100% {
    top: -20px;
    text-shadow: 0 1px 0 #ccc, 0 2px 0 #ccc, 0 3px 0 #ccc, 0 4px 0
#ccc,
      0 5px 0 rgb(217, 209, 209), 0 6px 0 #ccc, 0 7px 0 #ccc, 0 8px
0 #ccc, 0 9px 0 #ccc,
      0 50px 25px rgba(0, 0, 0, 0.2);
  }
}

</style>

<body>
<h1 style="color: rgb(193, 207, 207);">
  <table style="width:100% margin-left: 0px;">
    <tr>
      <!-- <th><img
src="https://media.tenor.com/images/30169e4a670daf12443df7d2dd140176
/tenor.gif" alt="NOT AVAILABLE" width="300" height="300"></th> -->
      <th> <iframe style="margin-top: 50px ;margin-left:0px"
width="1250" height="350"
src="https://www.youtube.com/embed/qkaozt9H6hM?
t=6&start=3&loop=1&autoplay=1&mute=1&controls=0">
      </iframe><br></th>
      <th></th>
    </tr>
  </table>

    <span>TEAM ID - PNT2022TMID15252</span>
  <!-- </br>
    <span>MEMBERS</span> -->
  </br>
    <span>KALAPOORANI P</span>
  </br>
    <span>KALPIKA K</span>
```

```
  </br>
    <span>HIBA MARIAM H</span>
  </br>
  <span>HEMALATHA J</span>

</h1>
<div class="header">
<div style="width:50%;float:left;font-size:2vw;text-
align:left;color:#c4dfd7; padding-top:1%;padding-left:5%;"><b>
GESTURE BASED TOOL FOR STERILE BROWSING OF RADIOLOGY
IMAGES</b></div>
  <div class="topnav-right"style="padding-top:0.5%;color:white">
    <a  class="active" href="{{ url_for('home')}}"><u>Home</u></a>
    <a  class="active" href="{{ url_for('intro')}}">Introduction</a>
    <a  class="active" href="{{ url_for('image1')}}">Prediction</a>
  </div>
</div>
</body>
</html>
```

intro.html

```
<html>
<script>


</script>


<style>
.header {  position: relative;
                top:0;
                margin:0px;
                z-index: 1;
                left: 0px;
                right: 0px;
                position: fixed;
                background-color: #1d2634  ;
                color: white;
                box-shadow: 0px 8px 4px #1d2634;
                overflow: hidden;
                padding-left:5px;
                font-family: 'Times New Roman';
                font-size: 2vw;
                width: 100%;
                height:15%;
                text-align: center;
          }
```

```
          .topnav {
  overflow: hidden;
  background-color: #FCAD98;
}

.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: #1d2634;
  color: rgb(238, 226, 234);
}

.topnav-right a:hover {
  background-color:  #263043;
  color: white;
}

.topnav-right {
  float: right;
  padding-right:165x;
}

body {

  background-color: #263043 ;
  background-repeat: no-repeat;
  background-size:cover;
  background-image: url("https://encrypted-tbn0.gstatic.com/images?
q=tbn:ANd9GcRGbQtbFCyxnw4WTUPZuN3xFcLVmMox9IaMTA&usqp=CAU");
    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: #FCAD98;
}

/* 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);
  color:black;
  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>

<body><br><br>
  <h1>GUIDELINES FOR GESTURES</h1>
<h1>
  <span> 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 this
project ,First the model is 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 predictes</span> <br>
 <span>0 then Rectangle operation is performed</span><br>
```

```
 <span>1,4 is  used for fixed resizing of image</span><br>
 <span>2 is used to rotate the radiology image</span><br>
 <span>3 makes the input image to be blurred</span><br>
 <span>5 gives a gray scale version of image</span><br>
  <!-- 1 stands for fixed resizing,2 rotates the image,3 makes the
image blurred,5 causes gray scale</span>  -->
  <!-- <span> recognition system </span> <span> provides us </span >
<span> an innovative,</span> <span>natural,</span> <span>  user
friendly </span> <span> way of interaction </span > <span> with the
computer</span> -->
  <!-- <span> which is more </span> <span> familiar to the </span>
<span>human beings. </span > <span> In our project, </span> <span>
the hand region </span> <span> is extracted from </span> <span> the
background  </span > <span> by using</span>
<span> Region of intrest. </span> <span> Then, </span> <span> we
will be </span > <span>predicting the labels </span> <span> based on
the </span> <span> CNN trained model weights  </span> <span> of hand
gestures </span > <span> using that predicted labels</span>
<span> we apply if conditions </span> <span> to control some of the
actions </span> <span>like </span > <span>reshaping , blur, flip of
the given image.</span> -->

</h1>
<!--Brian Tracy-->

<div class="header">
<div style="width:50%;float:left;font-size:2vw;text-
align:left;color:#c1e2d9; padding-top:1%;padding-left:2%;">GESTURE
BASED TOOL FOR STERILE BROWSING RADIOLOGY IMAGES</div>
  <div class="topnav-right"style="padding-top:0.5%;">

    <a class="active" href="{{ url_for('home')}}">Home</a>
    <a class="active" href="{{ url_for('intro')}}"><u>Intro</u></a>
    <a class="active" href="{{ url_for('image1')}}">Predict</a>
  </div>
</div>
</body>

</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.j
s"></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>
  .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:20px;
  }
  a:hover{
  background-color:black;
  color:black;
  font-size:30px;
  padding-left:10px;
  }

  div1{
    text-align: center;
    width: 650spx;
    height: 800px;
    padding: 190px;
    margin: 10px;
    position: absolute;
  }



  body
  {
      background-image:
url("https://www.businessinsider.in/thumb/msid-66990434,width-
700,height-525,imgsize-70168/the-v-sign-represents-peace-.jpg");
      background-size: cover;
  }

  .header {     position: relative;
        top:0;
        margin:0px;
        z-index: 1;
        left: 0px;
        right: 0px;
        position: fixed;
        background-color:#1d2634;
        color:rgb(181, 228, 236)k;
        box-shadow: 0px 8px 4px  #1d2634;
        overflow: hidden;
        padding-left:5px;
        font-family: 'Times New Roman';
        font-size: 2vw;
        width: 100%;
        height:15%;
        text-align: center;
```

```
      }
      .topnav {
    overflow: hidden;
    background-color: #056959;
  }

  .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: #1d2634;
    color: rgb(238, 226, 234);
  }

  .topnav-right a:hover {
    background-color: #263043;
    color: white;
  }



  .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: #091425;
    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;

  }
  .new
  {
    background-color: #f44336;
    height:10%;
  }
  .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: white;
}

/* 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;color:#c1e2d9; padding-top:1%;padding-left:5%;">GESTURE
BASED TOOL FOR STERLINE BROWSING OF RADIOLOGY IMAGES</div>
    <div class="topnav-right"style="padding-top:0.5%;">

      <a class="active" href="{{ url_for('home')}}">Home</a>
      <a class="active" href="{{
url_for('intro')}}">Introduction</a>
      <a class="active" href="{{
url_for('image1')}}"><u>Predict</u></a>
    </div>
  </div>
  <br>

    <div1 style =""><h1><font color="Black" size="6" font-
family="Roboto">TRANSFORMATION OF IMAGES,MADE EASY !</h1><br>
    <p><i><font color="Black" size="4"
fonr-family="sans-serif"></i>Please upload an image for which you
would like to perform various operations</p>
    <br>
          <div>
            <h4>Upload Your Image Here</h4>
        <form action = "http://localhost:5000/" id="upload-file"
method="post" enctype="multipart/form-data">
          <label for="imageUpload" class="new">
            Choose image
          </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>
```

App.py

```
from time import sleep
import time
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 keras import models
# from tensorflow.keras.models import load_model #to load our
trained model
from tensorflow import keras
from keras.layers import Dense
from keras.models import Sequential, load_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)
```

Model Training on IBM:

```
!pip install tensorflow
!pip install keras==2.2.4

!pip install watson-machine-learning-client

import os, types
import pandas as pd
from botocore.client import Config
import ibm_boto3

def __iter__(self): return 0

# @hidden_cell
# The following code accesses a file in your IBM Cloud Object Storage. It
includes your credentials.
# You might want to remove those credentials before you share the notebook.
cos_client = ibm_boto3.client(service_name='s3',
    ibm_api_key_id='nZwaQ3bXf7aulVbSlgB0T3lgUXjzVME_ENOm2UrnKia4',
    ibm_auth_endpoint="https://iam.cloud.ibm.com/oidc/token",
    config=Config(signature_version='oauth'),
    endpoint_url='https://s3.private.us.cloud-object-
storage.appdomain.cloud')

bucket = 'agesturebasedtoolforsterilebrowsi-donotdelete-pr-ahlrdk5m7icspf'
object_key = 'Dataset.zip'

streaming_body_1 = cos_client.get_object(Bucket=bucket, Key=object_key)
['Body']

from io import BytesIO
import zipfile
unzip = zipfile.ZipFile(BytesIO(streaming_body_1.read()),'r')
file_paths = unzip.namelist()
for path in file_paths:
    unzip.extract(path)

from ibm_watson_machine_learning import APIClient
wml_credentials = {
    "url": "https://us-south.ml.cloud.ibm.com",
    "apikey":"gE076jds2asxXK8dk6ZLLC1bw4kp3iSe_97D9dao4arr"

}
client = APIClient(wml_credentials)

def guide_from_space_name(client,space_name):
    space = client.spaces.get_details()
    return (next(item for item in space['resources'] if item['entity']
["name"]==space_name)['metadata']['id'])

space_uid = guide_from_space_name(client,"GestureModel")
space_uid
```

```
client.set.default_space(space_uid)
software_spec_uid=client.software_specifications.get_uid_by_name("tensorflow_
rt22.1-py3.9")
software_spec_uid

model_details=client.repository.store_model(model='Number_gesture_classifier_
new.tgz',meta_props={
    client.repository.ModelMetaNames.NAME:"CNN Model Building",
    client.repository.ModelMetaNames.TYPE:'tensorflow_2.7',
    client.repository.ModelMetaNames.SOFTWARE_SPEC_UID:software_spec_uid
})
model_id=client.repository.get_model_uid(model_details)

client.repository.download(model_id,'gesture_model.tar.gz')

# Model Download on local device
model_id = 'a8393184-0aff-4a98-a9dd-e0ffc2172ade'
client.repository.download(model_id,'gesture_model.tar.gz')
```

GitHub Link:

https://github.com/IBM-EPBL/IBM-Project-31926-1660206612

Demo Link:

https://www.youtube.com/watch?v=GrOkK_lpmcU