Gesture Based Tool for Sterile Browsing of Radiology Images

Done by:
IBM ID PNT2022TMID52128 – Kalpana V, Kartheesan R, Koushik SM, Lakshmi M

1 INTRODUCTION

Overview

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

Purpose

It is used to browse through the images obtained using radiology using hand gestures rather than using mouse,keyboard,etc thereby maintaining sterility.

2 LITERATURE SURVEY

S.NO	*Title*	*Author*	*Year*	*Inference*
1	***Hand-gesture-based sterile interface for the operating room using contextual cues for the navigation of radiological images.***	***1. M. Jacob*** ***2. J. Wachs*** ***3. R. Packer*** ***4. JAMIAs***	*2013*	***•This paper presents a method to improve the navigation and manipulation of radiological images through a sterile hand gesture recognition interface based on attentional contextual cues.*** ***•Computer vision algorithms were developed to extract intention and attention cues from the surgeon's behavior and combine them with sensory data from a commodity depth camera.*** ***• The developed interface was tested in a usability experiment to assess the effectiveness of***

				the new interface.
2	*A gesture-based tool for sterile browsing of radiology images.*	*1.J. Wachs, 2. H. Stern, 3.Y. Edan 4. M. Gillam, 5. J.Handler ,6.C.Feied, 7.Mark S. Smith*	*2008*	*•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 .• "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 easyinteraction .*
3	*Gesture-controlled image system positioning for minimally invasive interventions*	*1.Benjamin Fritsch, 2. T.Hoffmann ,3.A.Mewes, 4. G. Rose*	*2021*	*•Abstract This work examines how a touchless interaction concept contributes to*

				an efficient, direct, and sterile interaction workflow during CT- guided interventions. • *Two hand gesture sets were designed specifically under consideration of the clinical workflow and the hardware capabilities.*
• *4*	*A gesture-controlled projection display for CT-guided interventions*	*1. A. Mewes,* *2. P.Saalfeld,* *3. Oleksandr Riabikin,* *4. M. Skalej,* *5. C. Hansen*	*2015*	•*A gesture set to control basic functions of intervention software such as gestures for 2D image exploration, 3D object manipulation and selection and is well suited to become an integral part of future interventional suites.* • *Purpose The interaction with interventional imaging systems within a sterile environment is*

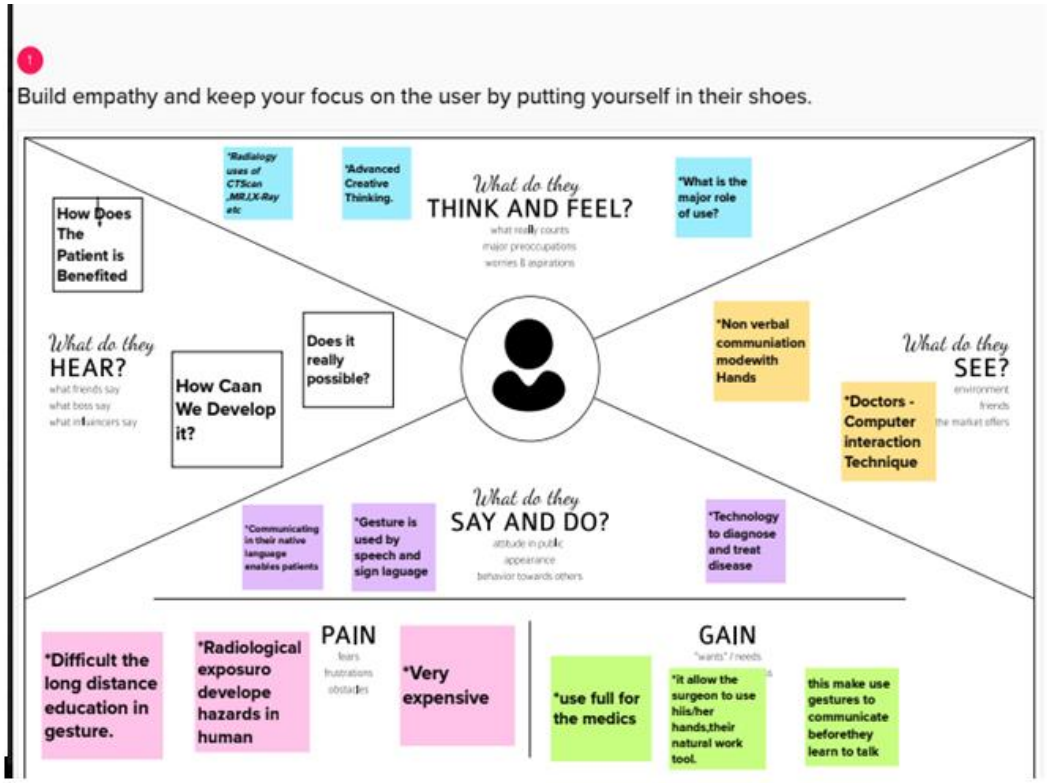
				a challenging task for physicians
5	***Gesture-Controlled Image Management for Operating Room***	***1.Rolf Wipfli 2.V. Dubois-Ferrière, 3. SylvainBudry4. P.Hoffmeyer, 5.C. Lovis***	***2016***	***•Under the premise that mouse cannot be used directly during surgery, gesture-controlled approaches demonstrate to be superior to oral instructions for image manipulation. •Objective In this work, we aim at comparing formally three different interaction modes for image manipulation that are usable in a surgery setting: 1) A gesture-controlled approach using Kinect ®; 2) oral instructions to a third part dedicated to manipulate the images; 3) direct manipulation using a mouse.***

6	***Introducing a brain-computer interface to facilitate intraoperative medical imaging contol***	***1.H.Esfandiari, 2.PascalTroxler 3.S.Hodel 4.DanielSuter 5.M.Farshad, 6.Nicola Cavalcanti, 7. O.Wetzel, 8.SylvanoMania9.F.Cornaz, 10.Farah Selman,***	***2022***	***•Background Safe and accurate execution of surgeries to date mainly rely on preoperative plans generated based on preoperative imaging. •Frequent intraoperative interaction with such patient images during the intervention is needed, which is currently a cumbersome process given that such images are generally displayed on peripheral two-dimensional (2D) monitors and controlled through interface devices that are outside the sterile filed.***

3 IDEATION AND PROPOSED SOLUTION:

3.1 EMPATHY MAP CANVAS

An empathy map is a collaborative tool teams can use to gain a deeper insight into their customers**. Much like a user persona, an empathy map can represent a group of users, such as a customer segment. The empathy map was originally created by Dave Gray and has gained much popularity within the agile community.**



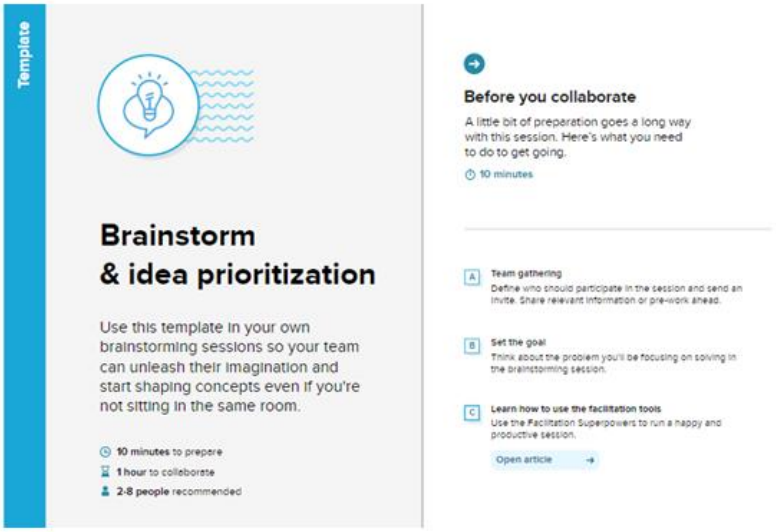
3.2 IDEATION AND BRAINSTORMING

Ideation is often closely related to the practice of brainstorming, a specific technique that is utilized to generate new ideas. A principal difference between ideation and brainstorming is that **ideation is commonly more thought of as being an individual pursuit, while brainstorming is almost always a group activity**

PROBLEM STATEMENT
BRAINSTORMING

Brain Storming 1

STEP-1 Team Gathering collaboration and select the problem statement



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

the doctor hand should be strike inside the operation theater.they would oftenneed to pick some object.they sould be able to communicate without touchingany object.so we tryhand gesture of the doctors using AI model.

2

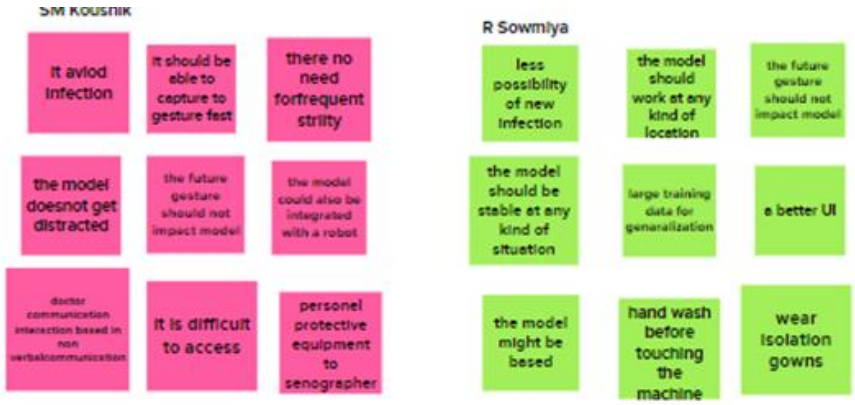
Brainstorm

Write down any ideas that come to mind that address your problem statement.

10 minutes

V Kalpana

R Kartheesan

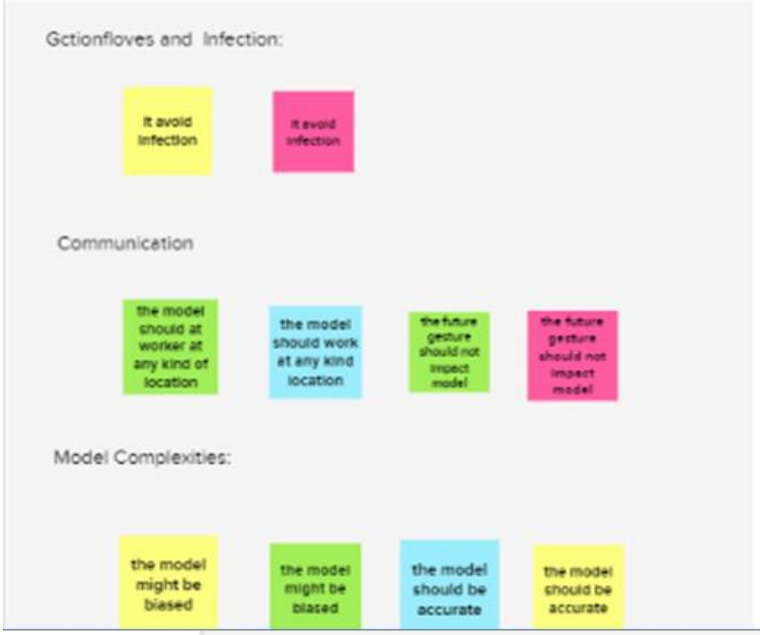


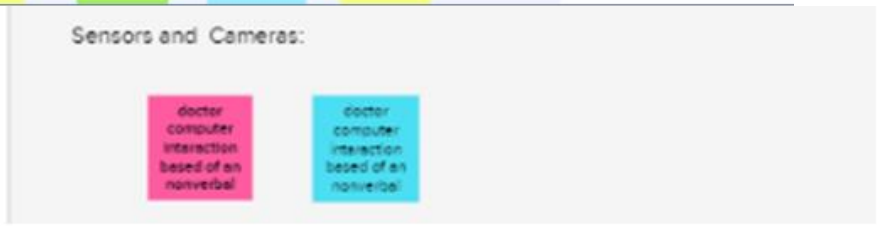
3

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





3.3 proposed solution:

S.NO	PARAMETER	DESCRIPTION
1.	Problem Statement (Problem to be solved)	Hand Gesture tool to do Contactless navigation of radiology images
2.	Idea / Solution description	Use technology to assist doctors by taking hand gestures as input and perform necessary actions.
3.	Novelty / Uniqueness	These Gesture helps us to visualize the Words and help Gain the Listener's Attention.
4.	Social Impact / Customer Satisfaction	The proposed system should maintain a good balance between complexity, accuracy and applicability.
5.	Business Model (Revenue Model)	A Hand-based Gesture Recognition System used for detecting any kind of Gestures which when the given input Gesture matches with the trained image.
6.	Scalability of the Solution	The proposed approach allows the learning of new gestures with no need of recording real subjects.

3.4 problem solution fit:

4 REQUIREMENTS ANALYS

Requirements analysis, also called requirements engineering, is **the process of determining user expectations for a new or modified product**. These features, called requirements, must be quantifiable, relevant and detailed. In software engineering, such requirements are often called functional specifications.

4.1 function requirements:

Following are the functional requirements of the proposed solution.

FR No	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Experience	A UI/UX shall be designed to help users interact with the system by using hand gestures
FR-2	Cloud Deployment	Deploy the trained CNN on the cloud.
FR-3	Hand Gesture Identification	Classify the images of hand gestures captured by a camera
FR-4	Application Domain	The CNN used by the system shall be trained on data that is relevant to the application domain.

4.2 nonfunction requirements:

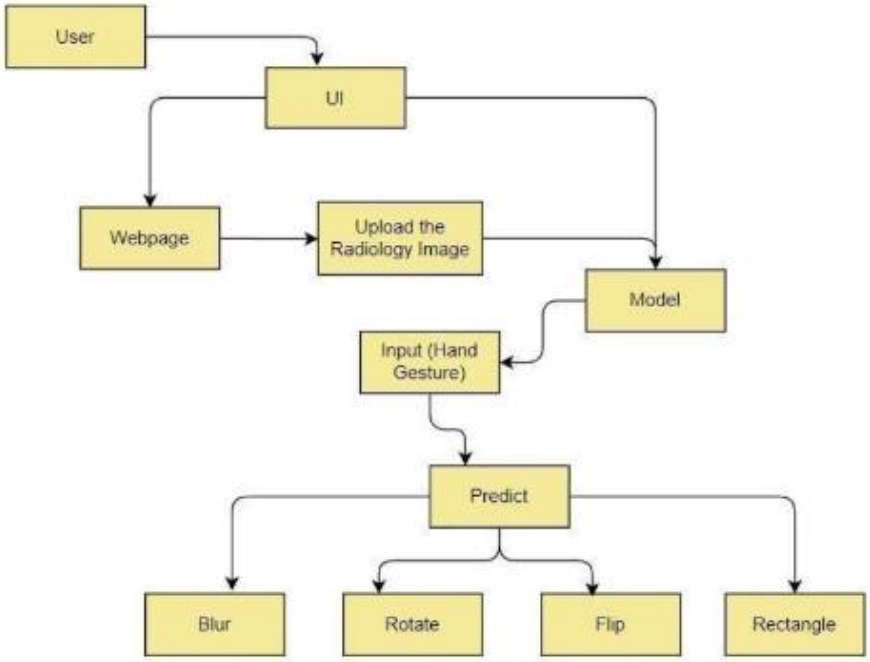
Following are the non-functional requirements of the proposed solution.

NFR No	Non-Functional Requirement	Description
NFR-1	Usability	This software shall be easy to use for all users with minimal instructions
NFR-2	Security	The Application shall permit users to access the system who are able to use it
NFR-3	Reliability	The Application shall be in the operational mode for at least 5 months after it may be Updated.
NFR-4	Performance	It will be able to respond to a user gesture in few milliseconds or 3 seconds

NFR-5	Availability	The model shall be available for HealthCare use if it remains operational
NFR-6	Scalability	The system shall be accessible to over thousands of concurrent users without any loss of performance

5 PROJECT DESIGN

Project design is an early phase of the project lifecycle where ideas, processes, resources, and deliverables are planned out**. A project design comes before a project plan as it's a broad overview whereas a project plan includes more detailed information.**



5.1 DATAFLOW DIAGRAM:

Table-1 : Components & Technologies:

S.No	Component	Description	Technology
1.	User Interface	Web UI	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-2 Building Model	Building CNN model to recognize the gesture.	Python, Keras
4.	Application Logic-3 Creation 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.	Server and Local Filesystem
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

- 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 input image.

5.2 SOLUTION AND TECHNICAL ARCHITECTURE:

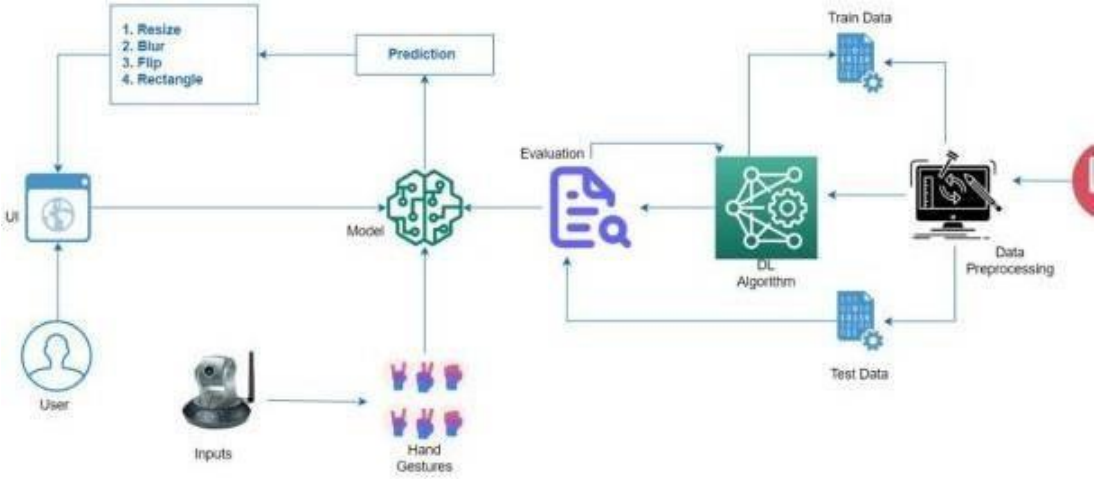


Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	For development of code, package manager, for building model	Visual Studio Code, Conda, TensorFlow
2.	Resilient	Gestures can be captured in different environments (variable brightness and distance).	OpenCV, TensorFlow
3.	Availability	Deploy on highly available server	IBM Cloud
4.	Performance	CNN model is used to predict the input gesture in a shorter span of time.	TensorFlow, Keras
5.	Diverse Dataset	Data augmentation to generate more data from limited set of images.	Keras

6PROJECT PLANNING AND SHEDULING:

Fundamentally, 'Project planning' is all about choosing and designing effective policies and methodologies to attain project objectives. While 'Project scheduling' is a procedure of assigning tasks to get them completed by allocating appropriate resources within an estimated budget and time-frame.

6.1SPRINT PLANNING AND ESTIMATION:

S.No	Title	Description	Date
1.	Literature Survey	Collect the relevant information on project use case, refer the existing solutions, technical papers, research publications etc.	15/10/2022
2.	Prepare empathy map	Preparation of Empathy Map Canvas and List of problem statements.	30/09/2022
3.	Ideation	By organizing this brainstorm session and to prioritize the best ideas based on the customer importance.	19/09/2022
4.	Proposed solution	This document, includes social impact, scalability of solution, future improvement etc.	20/10/2022
5.	Problem solution fit	Prepare problem - solution fit document & Solution Architecture	20/10/2022
6.	Solution Architecture	Prepare Solution Architecture document	20/10/2022
7.	Customer journey map	Understand the user interactions & experiences with the application.	03/10/2022
8.	Solution requirement	Prepare the Functional Requirement Document.	03/10/2022
9.	Data flow diagrams	Prepare the Data Flow Diagrams.	03/10/2022
10.	Technology architecture	Prepare Technology Architecture of the solution.	03/10/2022
11.	Milestone & activity list	Prepare Milestone & activity list.	01/11/2022
12.	Project development – delivery of sprint – 1,2,3 & 4	Develop & submit the developed code by testing it.	In progress

6.2 Sprint Delivery Schedule:

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	**Data Collection**	USN-1	Download the Dataset	10	High	V. Kalpana R. Kartheesan M.S. Koushik M. Lakshmi
Sprint-1		USN-2	Image Pre-processing	10	High	V. Kalpana R. Kartheesan M.S. Koushik M. Lakshmi
Sprint-1		USN-3	Import and Configure the Image Data Generator Library and Class	10	High	V. Kalpana R. Kartheesan M.S. Koushik M. Lakshmi

Sprint-1		USN-4	Apply Image Data Generator Functionalityto Train-Set and Test-Set	10	High	V. Kalpana R. Kartheesan M.S. Koushik M. Lakshmi

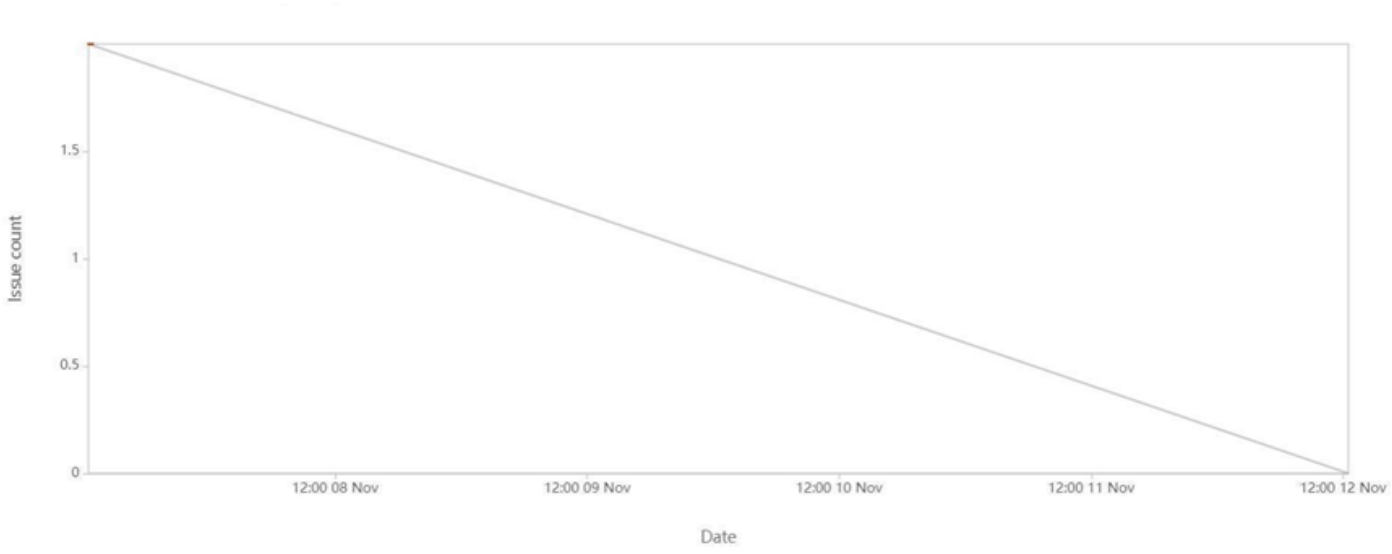
Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-2	**Model Building**	USN-5	Import the Model Building Libraries andInitializing the Model	10	High	V. Kalpana R. Kartheesan M.S. Koushik M. Lakshmi
Sprint-2		USN-6	Adding CNN Layers and Dense Layers	10	High	V. Kalpana R. Kartheesan M.S. Koushik M. Lakshmi
Sprint-2		USN-7	Configure the Learning Process	10	High	V. Kalpana R. Kartheesan M.S. Koushik M. Lakshmi
Sprint-2		USN-8	Train the Model, Save the Model and Test the Model	10	High	V. Kalpana R. Kartheesan M.S. Koushik M. Lakshmi
Sprint-3	**Applicatio nBuilding**	USN-9	Create Web Application using HTML, CSS,JavaScript	10	High	V. Kalpana R. Kartheesan M.S. Koushik M. Lakshmi
Sprint-3		USN-10	Build Python code	10	High	V. Kalpana R. Kartheesan M.S. Koushik M. Lakshmi
Sprint-3		USN-10	Run the Application	10	High	V. Kalpana R. Kartheesan M.S. Koushik M. Lakshmi
Sprint-4	**Train The Model onIBM**	USN-11	Register for IBM Cloud	10	High	V. Kalpana R. Kartheesan M.S. Koushik M. Lakshmi

Sprint-4		USN-12	Train the Model and Test the Model and its Overall Performance	10	High	V. Kalpana R. Kartheesan M.S. Koushik M. Lakshmi

Project Tracker, Velocity & Burndown Chart: (4 Marks)

Sprint	Total StoryPoints	Duration	Sprint Start Date	Sprint End Date (Planned)	Stories Completed on Planned Date)
Sprint-1	10	6 Days	24 Oct 2022	29 Oct 2022	10
Sprint-2	10	6 Days	31 Oct 2022	05 Nov 2022	10
Sprint-3	10	6 Days	07 Nov 2022	12 Nov 2022	10
Sprint-4	10	6 Days	14 Nov 2022	19 Nov 2022	10

6.3 Reports From JIRA:



7 CODING AND SOLUTIONING:

Gesture Based Tool for Sterile Browsing of Radiology Images

In this project we have used Convolutional Neural Network to first train the model on the images of different hand gestures, like showing numbers with fingers as 0,1,2,3,4,5. Then we made a web portal using Flask where user can input any image on which he wants to perform the operations. After uploading the image, our portal uses the integrated webcam to capture the video frame using OpenCV. The gesture captured in the video frame is compared with the Pre-trained model and the gesture is identified. If the prediction 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.

Model Building

Importing libraries

In [ ]:

```
# This library helps add support for large, multi-dimensional arrays and matrices
import numpy as np
#open source used for both ML and DL for computation
import tensorflow as tf
#it is a plain stack of layers
from tensorflow.keras.models import Sequential
#Dense layer is the regular deeply connected neural network layer
from tensorflow.keras.layers import Dense,Flatten, Dropout
#Faltten-used fot flattening the input or change the dimension, MaxPooling2D-for downsampling the image for Convolutional layer
from tensorflow.keras.layers import Convolution2D,MaxPooling2D
#Its used for different augmentation of the image
from tensorflow.keras.preprocessing.image import ImageDataGenerator
```

Augmenting the data

]:
```
#setting parameter for Image Data agumentation to the traing data
train_datagen = ImageDataGenerator(rescale=1./255,
                                   shear_range=0.2,
                                   zoom_range=0.2,
                                   horizontal_flip=True)
#Image Data agumentation to the testing data
test_datagen=ImageDataGenerator(rescale=1./255)
```

Loading our data and performing data agumentation

]:
```
#performing data agumentation to train data
x_train = train_datagen.flow_from_directory(r'C:\Users\darsh\OneDrive\Pictures\Desktop\Smart Bridge\Project Smart Interns\Dataset\train',
                                            target_size=(64, 64),
                                            batch_size=3,
                                            color_mode='grayscale',
                                            class_mode='categorical')
#performing data agumentation to test data
x_test = test_datagen.flow_from_directory(r'C:\Users\darsh\OneDrive\Pictures\Desktop\Smart Bridge\Project Smart Interns\Dataset\test',
                                          target_size=(64, 64),
                                          batch_size=3,
                                          color_mode='grayscale',
                                          class_mode='categorical')
```

```
Found 594 images belonging to 6 classes.
Found 30 images belonging to 6 classes.
```

]:
```
print(x_train.class_indices)#checking the number of classes
```

```
{'0': 0, '1': 1, '2': 2, '3': 3, '4': 4, '5': 5}
```

Model Creation

```
# Initializing the CNN
model = Sequential()
```

```
# First convolution layer and pooling
model.add(Convolution2D(32, (3, 3), input_shape=(64, 64, 1), activation='relu'))
model.add(MaxPooling2D(pool_size=(2, 2)))
```

```
# Second convolution layer and pooling
model.add(Convolution2D(32, (3, 3), activation='relu'))
# input_shape is going to be the pooled feature maps from the previous convolution layer
model.add(MaxPooling2D(pool_size=(2,2)))
```

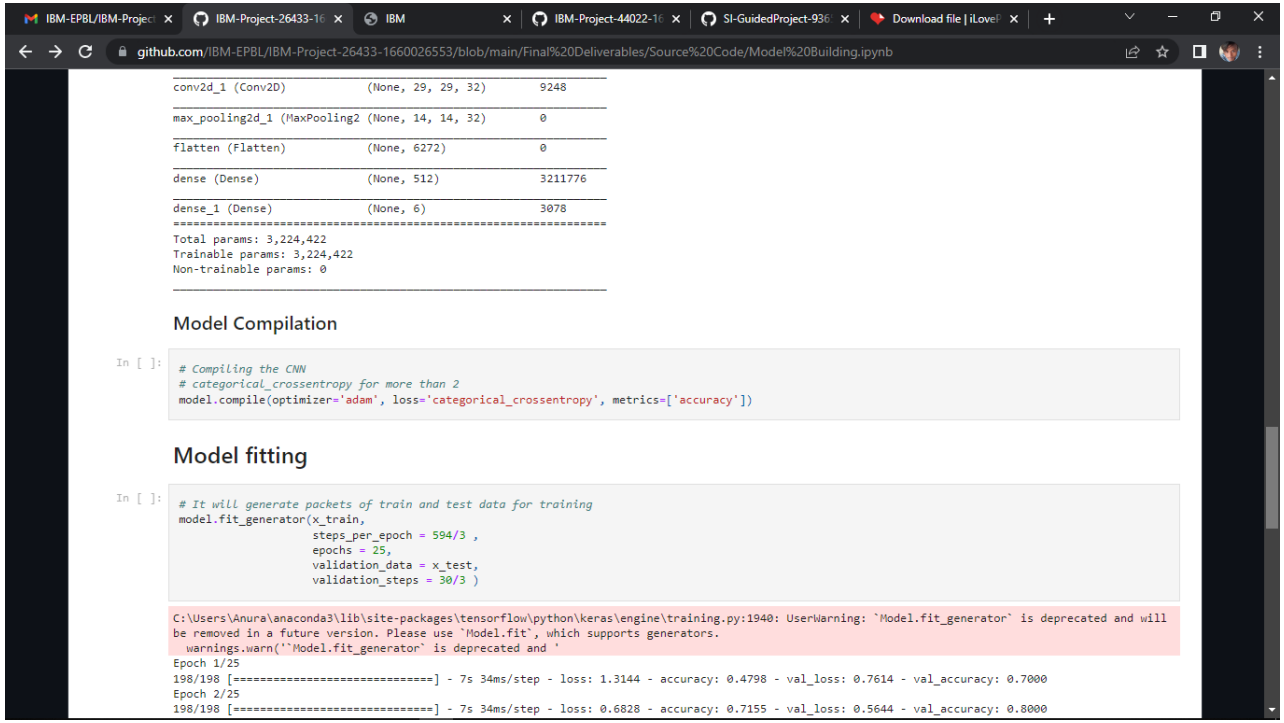
```
# Flattening the layers i.e. input layer
model.add(Flatten())
```

```
# Adding a fully connected layer, i.e. Hidden Layer
model.add(Dense(units=512 , activation='relu'))
```

```
# softmax for categorical analysis, Output Layer
model.add(Dense(units=6, activation='softmax'))
```

```
model.summary()#summary of our model
```

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d (Conv2D)              (None, 62, 62, 32)        320
_________________________________________________________________
max_pooling2d (MaxPooling2D) (None, 31, 31, 32)        0
```


IBM-EPBL/IBM-Project
IBM-Project-26433-16
IBM
IBM-Project-44022-16
SI-GuidedProject-936
Download file | iLovePDF
github.com/IBM-EPBL/IBM-Project-26433-1660026553/blob/main/Final%20Deliverables/Source%20Code/Model%20Building.ipynb
conv2d_1 (Conv2D) (None, 29, 29, 32) 9248
max_pooling2d_1 (MaxPooling2 (None, 14, 14, 32) 0
flatten (Flatten) (None, 6272) 0
dense (Dense) (None, 512) 3211776
dense_1 (Dense) (None, 6) 3078
Total params: 3,224,422
Trainable params: 3,224,422
Non-trainable params: 0
Model Compilation
In [ ]:
# Compiling the CNN
# categorical_crossentropy for more than 2
model.compile(optimizer='adam', loss='categorical_crossentropy', metrics=['accuracy'])
Model fitting
In [ ]:
It will generate packets of train and test data for training
model.fit_generator(x_train,
steps_per_epoch = 594/3 ,
epochs = 25,
validation_data = x_test,
validation_steps = 30/3)
C:\Users\Anura\anaconda3\lib\site-packages\tensorflow\python\keras\engine\training.py:1940: UserWarning: `Model.fit_generator` is deprecated and will be removed in a future version. Please use `Model.fit`, which supports generators.
warnings.warn('`Model.fit_generator` is deprecated and '
Epoch 1/25
198/198 [==============================] - 7s 34ms/step - loss: 1.3144 - accuracy: 0.4798 - val_loss: 0.7614 - val_accuracy: 0.7000
Epoch 2/25
198/198 [==============================] - 7s 34ms/step - loss: 0.6828 - accuracy: 0.7155 - val_loss: 0.5644 - val_accuracy: 0.8000

```
_________________________________________________________________
conv2d_1 (Conv2D)            (None, 29, 29, 32)        9248
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 14, 14, 32)        0
_________________________________________________________________
flatten (Flatten)            (None, 6272)              0
_________________________________________________________________
dense (Dense)                (None, 512)               3211776
_________________________________________________________________
dense_1 (Dense)              (None, 6)                 3078
=================================================================
Total params: 3,224,422
Trainable params: 3,224,422
Non-trainable params: 0
_________________________________________________________________
```

Model Compilation

[ ]:

```
# Compiling the CNN
# categorical_crossentropy for more than 2
model.compile(optimizer='adam', loss='categorical_crossentropy', metrics=['accuracy'])
```

Model fitting

[ ]:

```
# It will generate packets of train and test data for training
model.fit_generator(x_train,
                    steps_per_epoch = 594/3 ,
                    epochs = 25,
                    validation_data = x_test,
                    validation_steps = 30/3 )
```

```
C:\Users\Anura\anaconda3\lib\site-packages\tensorflow\python\keras\engine\training.py:1940: UserWarning: `Model.fit_generator` is deprecated and will be removed in a future version. Please use `Model.fit`, which supports generators.
  warnings.warn('`Model.fit_generator` is deprecated and '
Epoch 1/25
198/198 [==============================] - 7s 34ms/step - loss: 1.3144 - accuracy: 0.4798 - val_loss: 0.7614 - val_accuracy: 0.7000
Epoch 2/25
198/198 [==============================] - 7s 34ms/step - loss: 0.6828 - accuracy: 0.7155 - val_loss: 0.5644 - val_accuracy: 0.8000
```

```
198/198 [==============================] - 7s 34ms/step - loss: 0.0474 - accuracy: 0.9865 - val_loss: 0.4769 - val_accuracy: 0.9333
Epoch 13/25
198/198 [==============================] - 7s 35ms/step - loss: 0.0570 - accuracy: 0.9848 - val_loss: 0.1794 - val_accuracy: 0.9667
Epoch 14/25
198/198 [==============================] - 7s 34ms/step - loss: 0.0704 - accuracy: 0.9714 - val_loss: 0.4142 - val_accuracy: 0.9333
Epoch 15/25
198/198 [==============================] - 7s 34ms/step - loss: 0.0749 - accuracy: 0.9697 - val_loss: 0.4670 - val_accuracy: 0.9667
Epoch 16/25
198/198 [==============================] - 7s 34ms/step - loss: 0.0529 - accuracy: 0.9832 - val_loss: 0.3779 - val_accuracy: 0.9667
Epoch 17/25
198/198 [==============================] - 7s 34ms/step - loss: 0.0412 - accuracy: 0.9899 - val_loss: 0.4824 - val_accuracy: 0.9000
Epoch 18/25
198/198 [==============================] - 7s 35ms/step - loss: 0.0394 - accuracy: 0.9798 - val_loss: 0.2046 - val_accuracy: 0.9667
Epoch 19/25
198/198 [==============================] - 7s 35ms/step - loss: 0.0870 - accuracy: 0.9747 - val_loss: 0.3575 - val_accuracy: 0.9667
Epoch 20/25
198/198 [==============================] - 7s 34ms/step - loss: 0.0308 - accuracy: 0.9899 - val_loss: 0.3367 - val_accuracy: 0.9667
Epoch 21/25
198/198 [==============================] - 7s 35ms/step - loss: 0.0381 - accuracy: 0.9865 - val_loss: 0.2932 - val_accuracy: 0.9333
Epoch 22/25
198/198 [==============================] - 7s 35ms/step - loss: 0.0422 - accuracy: 0.9865 - val_loss: 0.5818 - val_accuracy: 0.8667
Epoch 23/25
198/198 [==============================] - 7s 34ms/step - loss: 0.0319 - accuracy: 0.9865 - val_loss: 0.3459 - val_accuracy: 0.9667
Epoch 24/25
198/198 [==============================] - 7s 35ms/step - loss: 0.0385 - accuracy: 0.9815 - val_loss: 0.3301 - val_accuracy: 0.9667
Epoch 25/25
198/198 [==============================] - ETA: 0s - loss: 0.0138 - accuracy: 0.99 - 7s 34ms/step - loss: 0.0138 - accuracy: 0.9966 - val_loss: 0.2801
- val_accuracy: 0.9667
```

[ ]:

Saving model

[ ]:
```
# Save the model
model.save('gesture.h5')
```

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

Testing the model

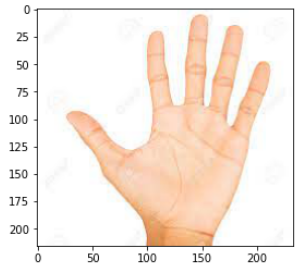
Importing Libraries

In [ ]:
```
from tensorflow.keras.models import load_model
from tensorflow.keras.preprocessing import image
model = load_model("gesture.h5") #loading the model for testing
path = "C:\Users\darsh\OneDrive\Pictures\Desktop\Smart Bridge\Project Smart Interns\Dataset\test\1\1.jpg"
```

Plotting the image

In [ ]:
```
%pylab inline
import matplotlib.pyplot as plt
import matplotlib.image as mpimg
imgs = mpimg.imread(path)
imgplot = plt.imshow(imgs)
plt.show()
```

```
Populating the interactive namespace from numpy and matplotlib
```



In [ ]:
```
#Loading of the image
img = image.load_img(path,
                     color_mode='grayscale',
                     target_size= (64,64))
x = image.img_to_array(img)#image to array
x.shape
```

Out[ ]:
```
(64, 64, 1)
```

In [ ]:
```
type(x)
```

Out[ ]:
```
numpy.ndarray
```

In [ ]:
```
#changing the shape
x = np.expand_dims(x,axis = 0)
```

In [ ]:
```
x.shape
```

Out[ ]:
```
(1, 64, 64, 1)
```

Predicting our results

In [ ]:
```
pred = model.predict_classes(x)#predicting the classes
pred
```

```
C:\Users\Anura\anaconda3\lib\site-packages\tensorflow\python\keras\engine\sequential.py:455: UserWarning: `model.predict_classes()` is deprecated and will be removed after 2021-01-01. Please use instead:* `np.argmax(model.predict(x), axis=-1)`,   if your model does multi-class classification   (e.g. if it uses a `softmax` last-layer activation).* `(model.predict(x) > 0.5).astype("int32")`,   if your model does binary classification   (e.g. if it uses a `sigmoid` last-layer activation).
  warnings.warn('`model.predict_classes()` is deprecated and '
```

Out[ ]:
```
array([5], dtype=int64)
```

In [ ]:
```
index=['0','1','2','3','4','5']
result=str(index[pred[0]])
result
```

Out[ ]:
```
'5'
```

In [ ]:
```
import numpy as np
p = []

for i in range(0,6):
    for j in range(0,5):
        path = "C:\Users\darsh\OneDrive\Pictures\Desktop\Smart Bridge\Project Smart Interns\Dataset\test\"+str(i)+"\\"+str(j)+".jpg"
        img = image.load_img(path,color_mode = "grayscale",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)
        p.append(pred)

print(p)
```

```
[array([0], dtype=int64), array([0], dtype=int64), array([0], dtype=int64), array([0], dtype=int64), array([0], dtype=int64), array([1], dtype=int64),
array([1], dtype=int64), array([1], dtype=int64), array([1], dtype=int64), array([1], dtype=int64), array([2], dtype=int64), array([2], dtype=int64),
array([1], dtype=int64), array([2], dtype=int64), array([2], dtype=int64), array([3], dtype=int64), array([3], dtype=int64), array([3], dtype=int64),
array([3], dtype=int64), array([3], dtype=int64), array([4], dtype=int64), array([4], dtype=int64), array([4], dtype=int64), array([4], dtype=int64),
array([4], dtype=int64), array([5], dtype=int64), array([5], dtype=int64), array([5], dtype=int64), array([5], dtype=int64), array([5], dtype=int64)]
```

In [ ]:
```
result = []
index=['0','1','2','3','4','5']
for i in p:
    result.append(index[i[0]])

print(result)
```

```
['0', '0', '0', '0', '0', '1', '1', '1', '1', '1', '2', '2', '1', '2', '2', '3', '3', '3', '3', '3', '4', '4', '4', '4', '4', '5', '5', '5', '5', '5']
```

8 TESTING:

Test cases:

Test case ID	Feature Type	Component	Test Scenario	Pre-Requisite	Steps To Execute	Test Data	Expected Result	Actual Result	Status
Test Case - O1	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 - O2	UI	Home Page	Verify the UI elements in Hompage		1.Enter URL and click go2.webpage displayed		Application should show below UIelements: a.Hompage b.Introduction c.Launch	Working as expected	Pass
Test Case - O3	Functional	Home page	Verify user is able to See the deatils about the webpage		1.Enter URL and click go2.Webpage Displayed 3.Displays necessary details		User should navigate to homepage	Working as expected	Pass
Test Case - O4	Functional	Introduction	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 - O5	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 - O6	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

Comments	TC for Automation(Y/N)	BUG ID	Executed By
			Devi G Renukas
			Devi G Renukas
			Devi G Renukas
			Ilamugil B S Naveen J
			Ilamugil B S Naveen J
			Ilamugil B S Naveen J

User Acceptance Testing:

1. PurposeofDocument

The purpose of this document is to briefly explain thetestcoverageandopenissuesofthe [ProductName] project at the time of the release to User Acceptance Testing (UAT).

2. DefectAnalysis

Thisreportshowsthenumberofresolvedor closed bugs at each severity level, and how they were resolved

Resolution	Severity1	Severity2	Severity3	Severity4	Subtotal

By Design	6	2	0	1	9
Duplicate	2	0	3	0	5
External	3	2	0	0	5
Fixed	10	3	2	16	31
Not Reproduced	0	0	0	0	0
Skipped	0	0	1	1	2
Won'tFix	0	2	1	2	5
Totals	21	9	7	20	57

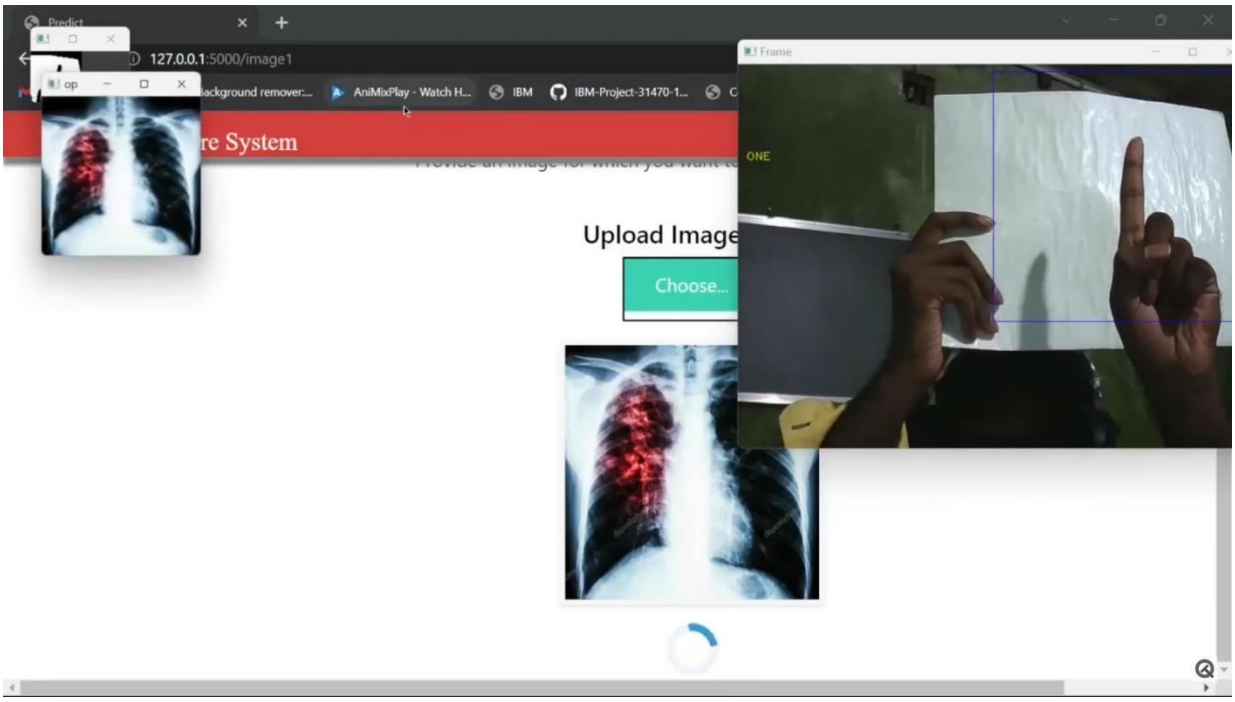
3. TestCaseAnalysis

Thisreportshowsthenumberoftestcasesthathavepassed,failed,anduntested

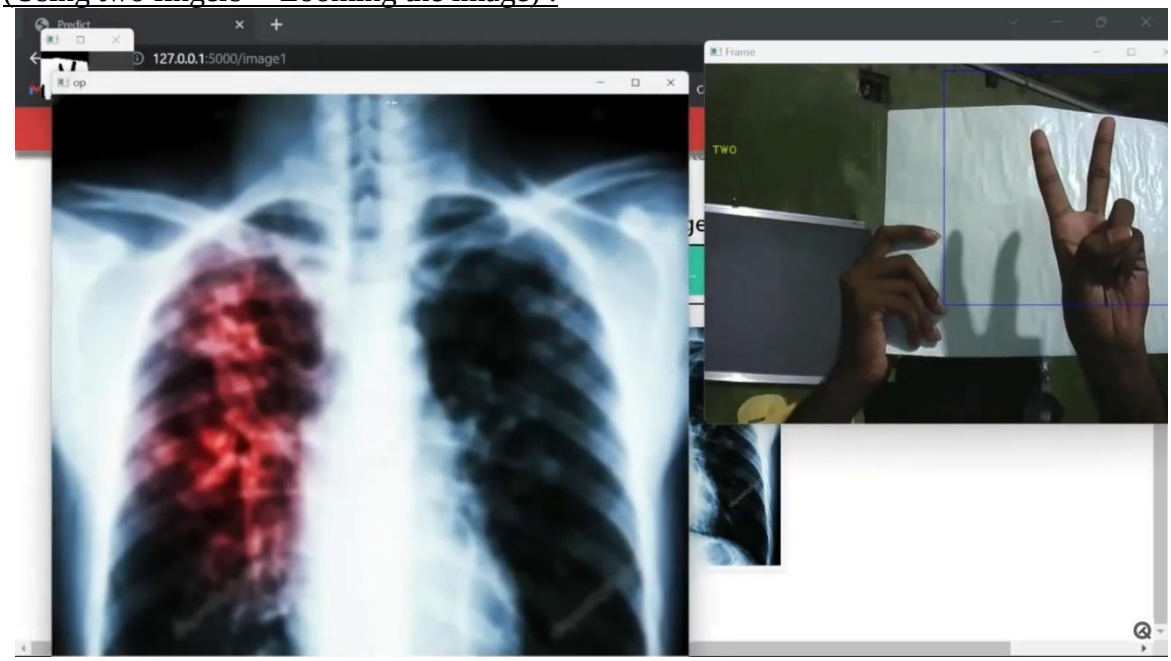
Section	TotalCases	Not Tested	Fail	Pass
PrintEngine	7	0	0	7
ClientApplication	10	1	0	10
Security	0	0	0	0
OutsourceShipping	6	0	0	6
ExceptionReporting	6	0	0	6
FinalReportOutput	5	0	0	5
VersionControl	1	0	0	1

9 RESULTS:

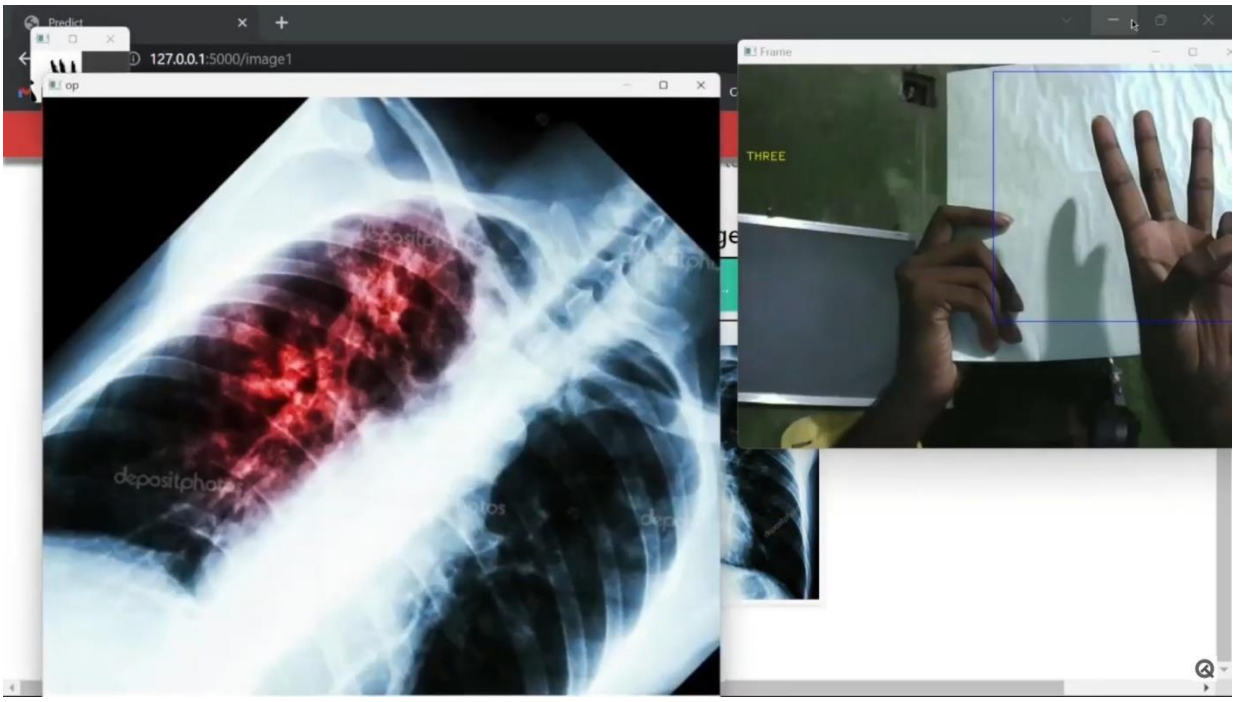
1. (Using One finger -- Opening Image):



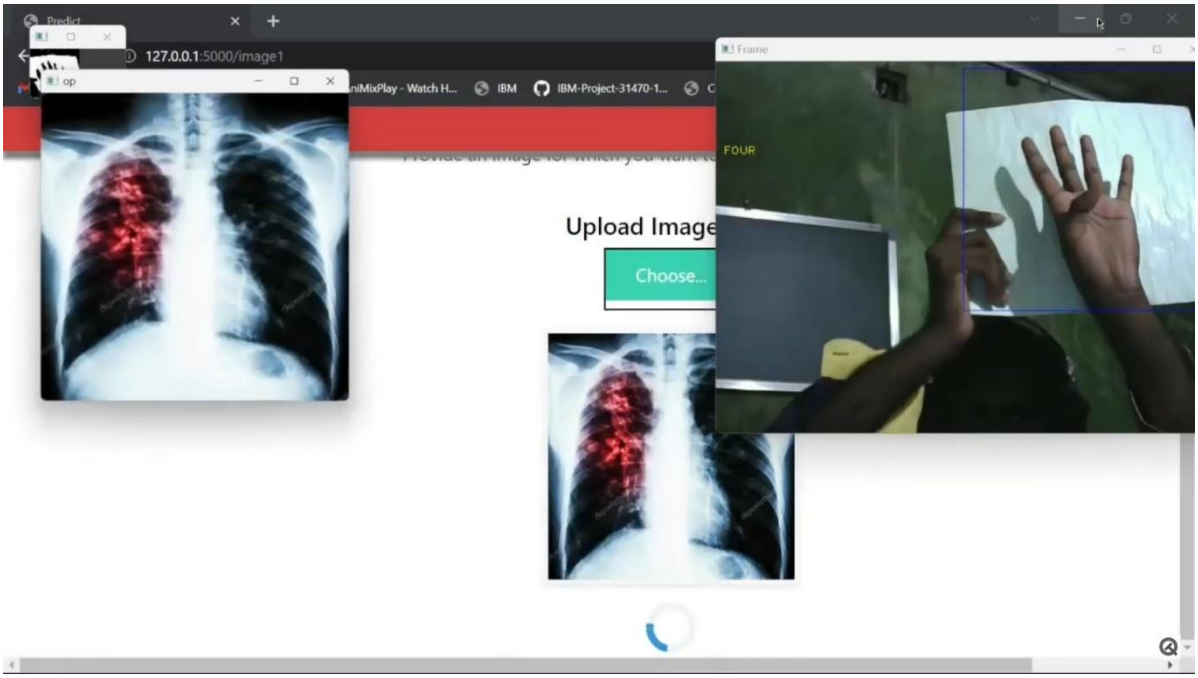
2. (Using two fingers -- Zooming the image) :



3. (Using 3 Fingers -- Rotating The Image) :



4. (Using four Fingers) :



Performance Metrics :

Model Performance Testing :

S.no	Parameter	Values	Screenshots
1.	Model Summary	Total Params : 3,224,422 Trainable Params : 3,224,422	Fig - 9.1
2.	Accuracy	Training Accuracy - 98%	Fig - 9.2 (a)
		Validation Accuracy - 96%	Fig - 9.2 (b)

Summary

```
In [8]: model.summary()
```

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d (Conv2D)             (None, 62, 62, 32)        320

 max_pooling2d (MaxPooling2D  (None, 31, 31, 32)       0
 )

 conv2d_1 (Conv2D)           (None, 29, 29, 32)        9248

 max_pooling2d_1 (MaxPooling  (None, 14, 14, 32)       0
 2D)

 flatten (Flatten)           (None, 6272)              0

 dense (Dense)               (None, 512)               3211776

 dense_1 (Dense)             (None, 6)                 3078

=================================================================
Total params: 3,224,422
Trainable params: 3,224,422
Non-trainable params: 0
_________________________________________________________________
```

Fig - 9.1

```
In [10]: train_loss = his.history['loss']
         val_loss   = his.history['val_loss']
         train_acc  = his.history['accuracy']
         val_acc    = his.history['val_accuracy']
         xc         = range(1,26)

         plt.figure()
         plt.plot(xc, train_loss)
         plt.plot(xc, val_loss)

Out[10]: [<matplotlib.lines.Line2D at 0x183c6b614e0>]
```

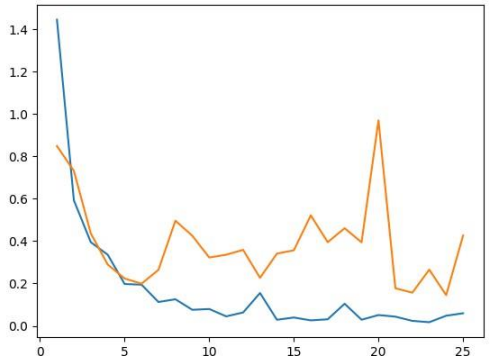


Fig - 9.2 (a)

```
In [11]: plt.plot(xc, train_acc)
         plt.plot(xc, val_acc)

Out[11]: [<matplotlib.lines.Line2D at 0x183c6c08d90>]
```

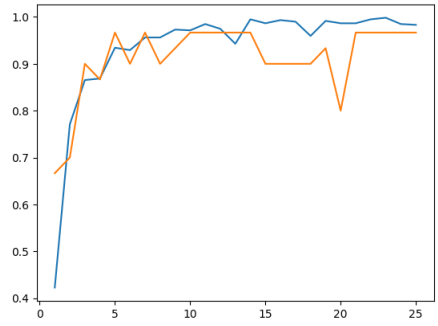


Fig - 9.2 (b)

10 ADVANTAGE AND DISADVANTAGES

Advantages:

- . As mention in Problem Statement, The Images can be viewed, Zoom, operated Without usingKeyboard or Mouse for acheiving Sterile Browsing.
- Not only Radiology Images, But also Other Image Files can be uploaded and Viewed.
- Can be easily implemented on low powered processor.
- Works in Both Linux and Windows.

Disadvantages:

- Light position should be correctly positioned in order to improve accuracy.
- Sometimes it will stop automatically after certain period of time on running, In order isContinue, there should be some Input in command prompt (Terminal, PS).

11CONCLUSION

In this project we developed a tool which recognises 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.

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 useability 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.

13 APPENDIX

Github&Project Demo Link:
https://careereducation.smartinternz.com/college/universal-college-of-engineering-and-technology-516