A

Major project

on

FACE MASK DETECTION USING CONVOLUTION NEURAL NETWORKS

(Submitted in partial fulfillment of the requirements for the award of Degree)

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE AND ENGINEERING BY

V.PAVAN (187R1A05P3)

B.CHARANSAI (187R1A05M6)

T.JESSICA (187R1A05P2)

Under the Guidance of

J.SRIVIDYA

(Assistant Professor)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CMR TECHNICAL CAMPUS

UGC AUTONOMOUS

(Accredited by NAAC, NBA, Permanently Affiliated to JNTUH, Approved by AICTE, New Delhi) Recognized Under Section 2(f) & 12(B) of the UGCAct,1956.

Kandlakoya (V), Medchal Road, Hyderabad-5014

2018-22

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the project entitled "**FACEMASK DETECTION USING CNN**" being submitted by **V.PAVAN (187R1A05P3), B.CHARANSAI (187R1A05M6) & T.JESSICA (187R1A05P2)** in partial fulfillment of the requirements for the award of the degree of B.Tech in Computer Science and Engineering to the Jawaharlal Nehru Technological University Hyderabad, is a record of bonafide work carried out by them under our guidance and supervision during the year 2021-22.

The results embodied in this thesis have not been submitted to any other University or Institute for the award of any degree or diploma.

J. Srividya
Assistant Professor
INTERNAL GUIDE

Dr. A. Raji Reddy
DIRECTOR

Dr. K. Srujan Raju
HOD

EXTERNAL EXAMINER

Submitted for viva voice Examination held on _____

ACKNOWLEDGEMENT

Apart from the efforts of us, the success of any project depends largely on the encouragement and guidelines of many others. We take this opportunity to express our gratitude to the people who have been instrumental in the successful completion of this project.

We take this opportunity to express my profound gratitude and deep regard to my guide **J. Srividya,** Assistant Professor for his exemplary guidance, monitoring and constant encouragement throughout the project work. The blessing, help and guidance given by him shall carry us a long way in the journey of life on which we are about to embark. We also take this opportunity to express a deep sense of gratitude to the Project Review Committee (PRC) **Mr. A. Uday Kiran, Mr. J. Narasimha Rao, Dr. T. S. Mastan Rao, Mrs. G. Latha, Mr. A. Kiran Kumar,** for their cordial support, valuable information and guidance, which helped us in completing this task through various stages

We are also thankful to **Dr. K. Srujan Raju,** Head, Department of Computer Science and Engineering for providing encouragement and support for completing this project successfully.

Apart from the efforts of us, the success of any project depends largely on the encouragement and guidelines of many others. We take this opportunity to express our gratitude to the people who have been instrumental in the successful completion of this project.

We take this opportunity to express my profound gratitude and deep regard to my guide **Mr. Malyadri,** Associate Professor for his exemplary guidance, monitoring and constant encouragement throughout the project work. The blessing, help and guidance given by him shall carry us a long way in the journey of life on which we are about to embark. We also take this opportunity to express a deep sense of gratitude to the Project Review Committee (PRC) **Mr. A. Uday Kiran, Mr. J. Narasimha Rao, Dr. T. S. Mastan Rao, Mrs. G. Latha, Mr. A. Kiran Kumar,** for their cordial support, valuable information and guidance, which helped us in completing this task through various stages.

We are also thankful to **Dr. K. Srujan Raju,** Head, Department of Computer Science and Engineering for providing encouragement and support for completing this project successfully.

V.PAVAN **(187R1A05P3)**

B.CHARANSAI **(187R1A05M6)**

T.JESSICA **(187R1A05P2)**

ABSTRACT

In order to prevent the spread of CORONA virus, everyone must wear a mask during the pandemic. In these tough times of COVID-19 it is necessary to build a model that detects people with and without masks in real-time as it works as a simple precautionary measure to prevent the spread of virus. If deployed correctly, this machine learning technique helps in simplifying the work of frontline warriors and saving their lives. A basic Convolutional Neural Network (CNN) model is built using TensorFlow, Keras, Scikit-learn and OpenCV to make the algorithm as accurate as possible. Javascript API helps in accessing webcams for real-time face mask detection. Since Google Colab runs on a web browser it can't access local hardware like a camera without APIs. The proposed work contains three stages: (i) pre-processing, (ii) Training a CNN and (iii) Real- time classification. The first part is the Pre-processing section, which can be divided into "Grayscale Conversion " of RGB images, "image resizing and normalization" to avoid false predictions. Then the proposed CNN classifies faces with and without masks as the output layer of the proposed CNN architecture contains two neurons with Softmax activation to classify the same. Categorical cross-entropy is employed as a loss function. The proposed model has a Validation accuracy of 96%. If anyone in the video stream is not wearing a protective mask a Red coloured rectangle is drawn around the face with a dialog entitled as NO MASK and a Green coloured rectangle is drawn around the face of a person wearing a MASK.

LISTS OF FIGURES

FIGURE NO	FIGURE NAME	PAGE NO
Figure 3.1	Project Architecture	11
Figure 3.2	Data Flow Diagram	12
Figure 3.3	Use case diagram	14
Figure 3.4	Class diagram	15
Figure 3.5	Activity diagram	16
Figure 3.6	Sequence diagram	17

LISTS OF SCREENSHOTS

SCREENSHOT NO	SCREENSHOT NAME	PAGE NO
Screenshot 5.1.1	Face Mask Detection Image 1	24
Screenshot 5.1.2	Face Mask Detection Image 2	25
Screenshot 5.1.3	Face Mask Detection Image 3	25
Screenshot 5.1.4	Face Mask Detection Image 4	26
Screenshot 5.1.5	Face Mask Detection Image 5	26
Screenshot 5.1.6	Face Mask Detection Image 6	27
Screenshot 5.1.7	Face Mask Detection Image 7	28
Screenshot 5.1.8	Face Mask Detection Image 8	29

TABLE OF CONTENTS

ABSTRACT i

LIST OF FIGURES ii

LIST OF SCREENSHOTS iii

1. INTRODUCTION 1

1.1 PROJECT SCOPE 2

1.2 PROJECT PURPOSE 2

1.3 PROJECT FEATURES 2

2. SYSTEM ANALYSIS 3

2.1 PROBLEM DEFINITION 4

2.1.1 EXISTING SYSTEM 4

2.1.2 LIMITATIONS OF THE EXISTING SYSTEM 4

2.2 PROPOSED SYSTEM 4

2.2.1 ADVANTAGES OF PROPOSED SYSTEM 5

2.3 SYSTEM REQUIREMENTS 5

2.3.1 SOFTWARE REQUIREMENTS 5

2.3.2 HARDWARE REQUIREMENTS 6

2.3.3 FUNCTIONAL REQUIREMENTS 6

2.3.4 NON FUNCTIONAL REQUIREMENTS 7

2.4 FEASIBILITY STUDY 8

2.4.1 ECONOMICAL STUDY 8

2.4.2 TECHNICAL STUDY 9

2.4.3 SOCIAL STUDY 9

3. ARCHITECTURE 10

3.1 PROJECT ARCHITECTURE 11

3.2 DATA FLOW DIAGRAM 12

3.3 USE CASE DIAGRAM 14

3.4 CLASS DIAGRAM 15

3.5 ACTIVITY DIAGRAM 16
3.6 SEQUENCE DIAGRAM 17

4. **IMPLEMENTATION** 18
4.1 SAMPLE CODE 19
5. **RESULTS** 23
5.1 SCREENSHOTS 24
6 **SYSTEM TESTING** 30
6.1 INTRODUCTION TO TESTING 31
6.2 TYPES OF TESTING 31
6.2.1 UNIT TESTING 31
6.2.2 INTEGRATION TESTING 32
6.2.3 FUNCTIONAL TESTING 33
6.3 TEST CASES 34
6.3.1 USER REQUIREMENTS 34
7 **CONCLUSION & FUTURE SCOPE** 35
7.1 PROJECT CONCLUSION 36
7.2 FUTURE SCOPE 36
8. **BIBLIOGRAPHY** 37
8.1 REFERENCES 38
8.2 WEBSITES 39
8.3 GITHUB LINK 39

1.INTRODUCTION

1.INTRODUCTION

1.1 PROJECT SCOPE:

This project proposes a method to detect if a person wears a mask or not. We have used convolutional neural networks for the project. The model is trained based on real world dataset and tested with live video streaming with good accuracy. Further, the accuracy of the model with different hyper parameters and multiple people at different distances and locations of the frame is done.

If anyone in the video stream is not wearing a protective mask and a Red colored rectangle is drawn around the face with a dialog entitled NO MASK. Similarly, a Green color rectangle is drawn around the face of a person wearing MASK.

1.2 PROJECT PURPOSE:

The purpose of this project is to classify the persons in a group of people who wears mask or not in a video surveillance or any images. By this we can stop or reduce any viruses or diseases which can spread among society.

1.2 PROJECT FEATURES:

In this project we have used CNN Convolutional Neural Networks which is used for Image Recognition and python language and we used some libraries to be imported

2.SYSTEM ANALYSIS

SYSTEM ANALYSIS

1.1 EXISTING SYSTEM

Government and Public health agencies are recommending face masks as essential measures to keep us safe when venturing into public. To mandate the use of facemasks it becomes essential to devise some technique that forces individuals to apply a mask before exposure to public places. Face mask detection is used to detect whether a person is wearing a mask or not. In fact, the problem is reverse engineering of face detection where the face is detected using different machine learning algorithms for the purpose of security, authentication and surveillance. Face detection is a key area in the field of Computer Vision and Pattern Recognition. A significant body of research has contributed sophisticated algorithms for face detection in the past.

1.1.1 DISADVANTAGES OF EXISTING SYSTEM:

1.It is difficult to identify a person with a mask or without a mask in surveillance.

1.2 PROPOSED SYSTEM:

The proposed CNN classifies faces with and without masks as the output layer of the proposed CNN architecture contains two neurons with Softmax activation to classify the same. Categorical cross-entropy is employed as a loss function. The proposed model has a Validation accuracy of 96%. If anyone in the video stream is not wearing a protective mask a Red coloured rectangle is drawn around the face with a dialog entitled as NO MASK and a Green coloured rectangle is drawn around the face of a person wearing a MASK.

1.2.1 ADVANTAGES OF PROPOSED SYSTEM:

1.The proposed model has Validation accuracy of 96%. If anyone in the video stream is not wearing a protective mask a Red coloured rectangle is drawn around the face with a dialog entitled as NO MASK and a Green coloured rectangle is drawn around the face of a person wearing a MASK

1.3 SYSTEM REQUIREMENTS:

1.3.1 SOFTWARE REQUIREMENTS

The functional requirements or the overall description documents include the product perspective and features, operating system and operating environment, graphics requirements, design constraints and user documentation.

The appropriation of requirements and implementation constraints gives the general overview of the project in regards to what the areas of strength and deficit are and how to tackle them

- **Python idel 3.7 version (or)**
- **Anaconda 3.7 (or)**
- **Jupiter (or)**
- **Google colab**

1.3.2 HARDWARE REQUIREMENTS

Minimum hardware requirements are very dependent on the particular software being developed by a given Enthought Python / Canopy / VS Code user. Applications that need to store large arrays/objects in memory will require more RAM, whereas applications that need to perform numerous calculations or tasks more quickly will require a faster processor

- **Operating system : windows, linux**
- **Processor : minimum intel i2**
- **Ram : minimum 4 gb**
- **Hard disk : minimum 250gb**

1.3.3 FUNCTIONAL REQUIREMENTS

1.Data Collection

2.Data Preprocessing

3.Training And Testing

4.Modiling

5.Predicting

1.3.4 NON FUNCTIONAL REQUIREMENTS

NON-FUNCTIONAL REQUIREMENT (NFR) specifies the quality attribute of a software system. They judge the software system based on Responsiveness, Usability, Security, Portability and other non-functional standards that are critical to the success of the software system. Example of nonfunctional requirement, *"how fast does the website load?"* Failing to meet non-functional requirements can result in systems that fail to satisfy user needs. Non-functional Requirements allow you to impose constraints or restrictions on the design of the system across the various agile backlogs. Example, the site should load in 3 seconds when the number of simultaneous users are > 10000. Description of non-functional requirements is just as critical as a functional requirement

- Usability requirement
- Serviceability requirement
- Manageability requirement
- Recoverability requirement
- Security requirement
- Data Integrity requirement
- Capacity requirement
- Availability requirement
- Scalability requirement
- Interoperability requirement
- Reliability requirement
- Maintainability requirement
- Regulatory requirement
- Environmental requirement

SYSTEM STUDY

1.4 FEASIBILITY STUDY

The feasibility of the project is analyzed in this phase and a business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential

Three key considerations involved in the feasibility analysis are

- ECONOMICAL FEASIBILITY
- TECHNICAL FEASIBILITY
- SOCIAL FEASIBILITY

1.4.1 ECONOMICAL FEASIBILITY

This study is carried out to check the economic impact that the system will have on the organization. The amount of funds that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus the developed system was within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased

1.4.2 TECHNICAL FEASIBILITY

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

1.4.3 SOCIAL FEASIBILITY

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

3.ARCHITECTURE

3.1 SYSTEM ARCHITECTURE:

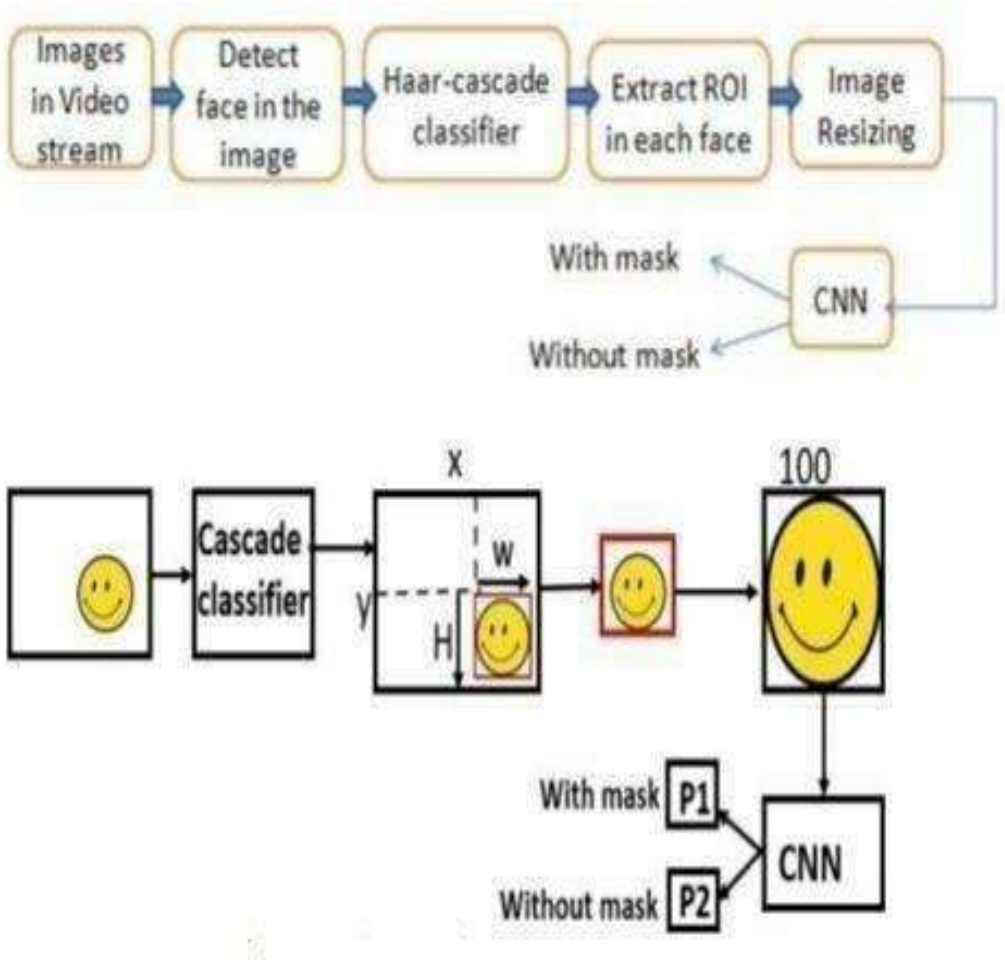

Figure 3.1: Project Architecture

3.2 DATA FLOW DIAGRAM:

1. The DFD is also called a bubble chart. It is a simple graphical formalism that can be used to represent a system in terms of input data to the system, various processing carried out on this data, and the output data is generated by this system.

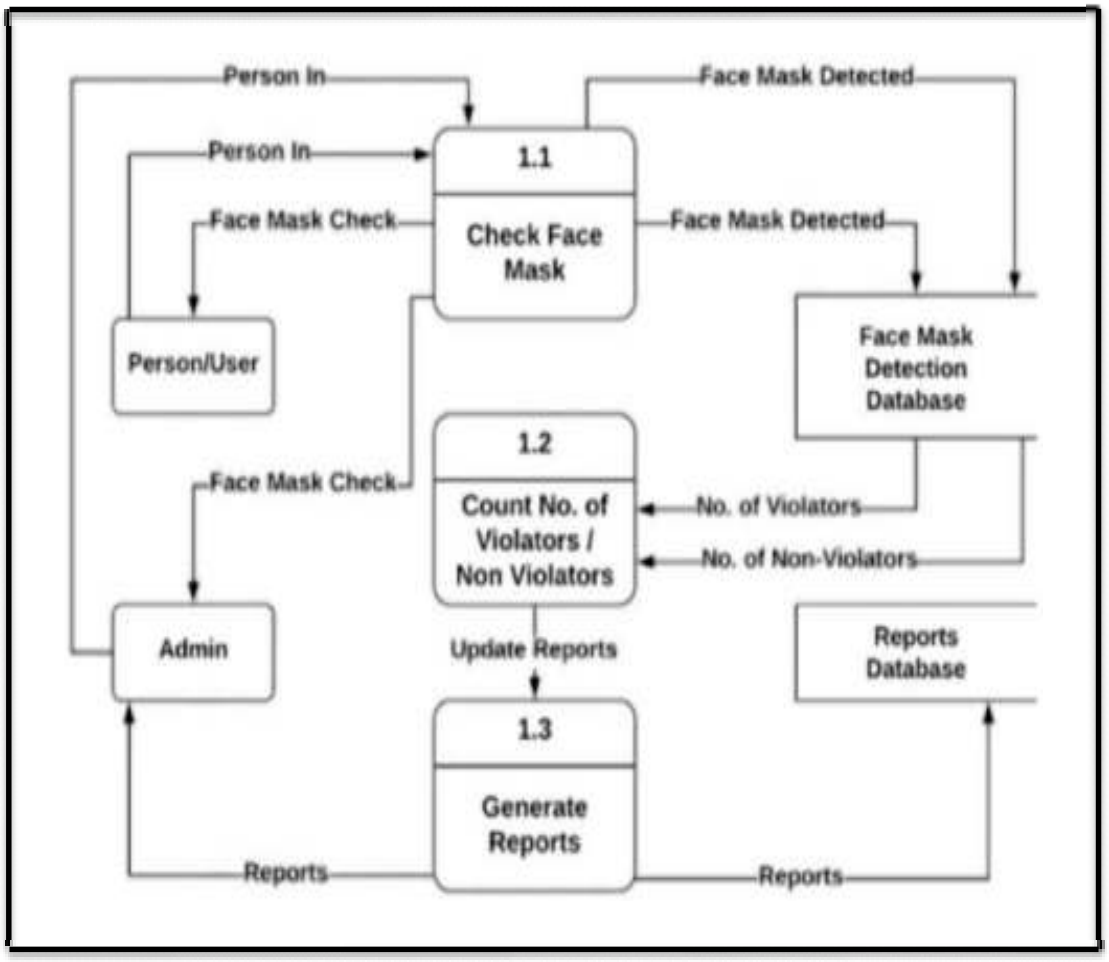

Figure 3.2: Data Flow Diagram

UML DIAGRAMS

UML stands for Unified Modeling Language. UML is a standardized general-purpose modeling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group.

The goal is for UML to become a common language for creating models of object oriented computer software. In its current form UML has two major components: a Meta-model and a notation. In the future, some form of method or process may also be added to; or associated with, UML.

The Unified Modeling Language is a standard language for specifying, Visualization, Constructing and documenting the artifacts of software systems, as well as for business modeling and other non-software systems.

The UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems.

UML is a very important part of developing objects oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects.

GOALS:

The Primary goals in the design of the UML are as follows:

1. Provide users a ready-to-use, expressive visual modeling Language so that they can develop and exchange meaningful models.
2. Provide extendibility and specialization mechanisms to extend the core concepts.

3.3 Use case diagram:

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted

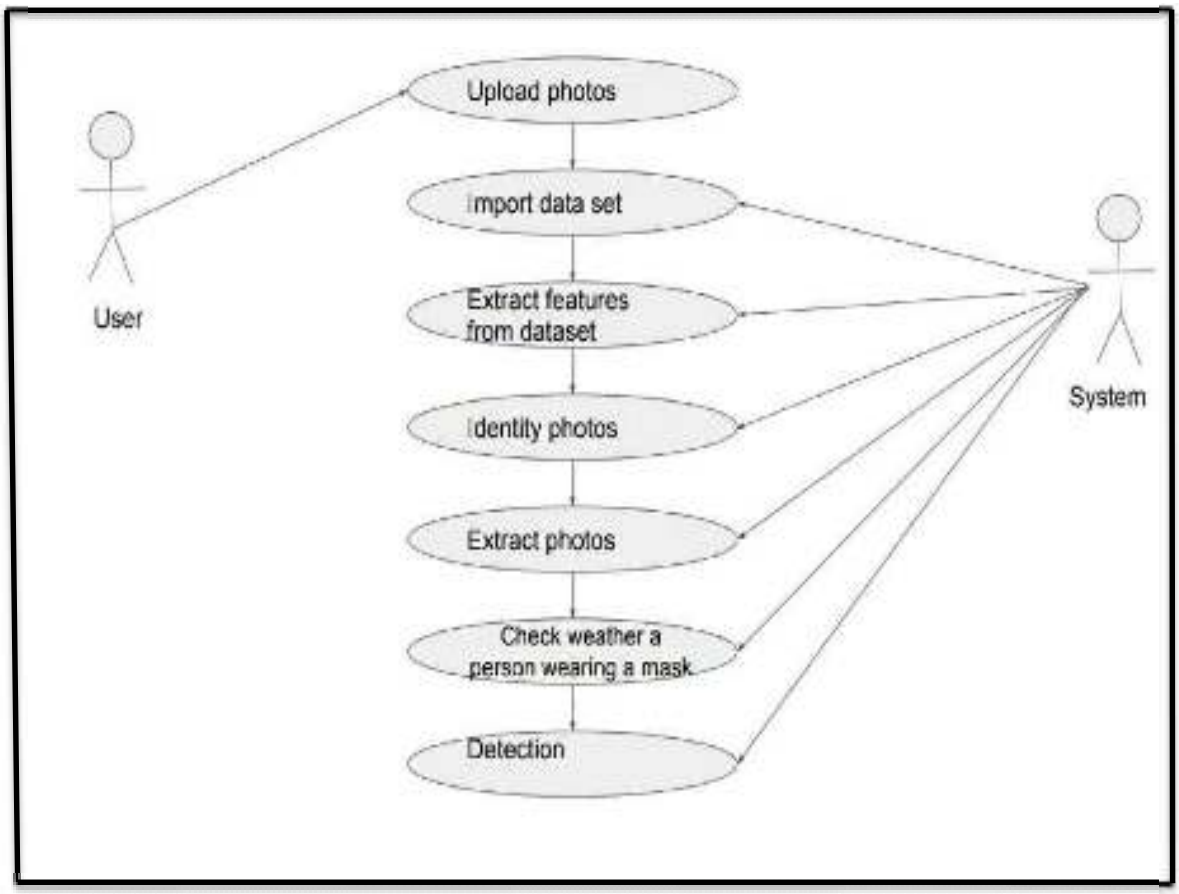

Figure 3.3: Use Case Diagram for Face Mask Detection using CNN

3.4 Class diagram:

The class diagram is used to refine the use case diagram and define a detailed design of the system. The class diagram classifies the actors defined in the use case diagram into a set of interrelated classes. The relationship or association between the classes can be either an "is-a" or "has-a" relationship. Each class in the class diagram may be capable of providing certain functionalities. These functionalities provided by the class are termed "methods" of the class. Apart from this, each class may have certain "attributes" that uniquely identify the class.

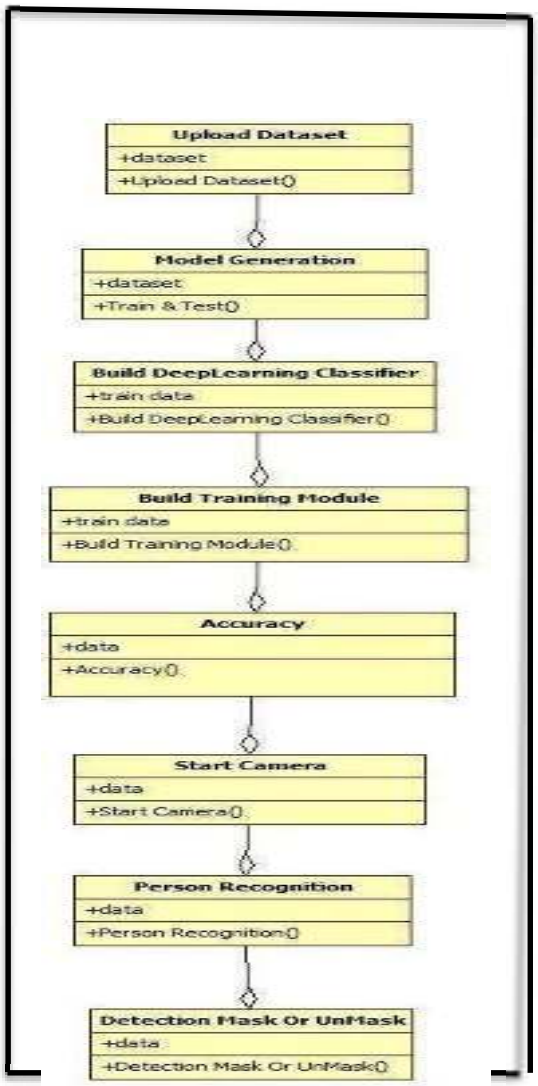

Figure 3.4 Class Diagram for Face Mask Detection using CNN

3.5 Activity diagram:

The process flows in the system are captured in the activity diagram. Similar to a state diagram, an activity diagram also consists of activities, actions, transitions, initial and final states, and guard conditions.

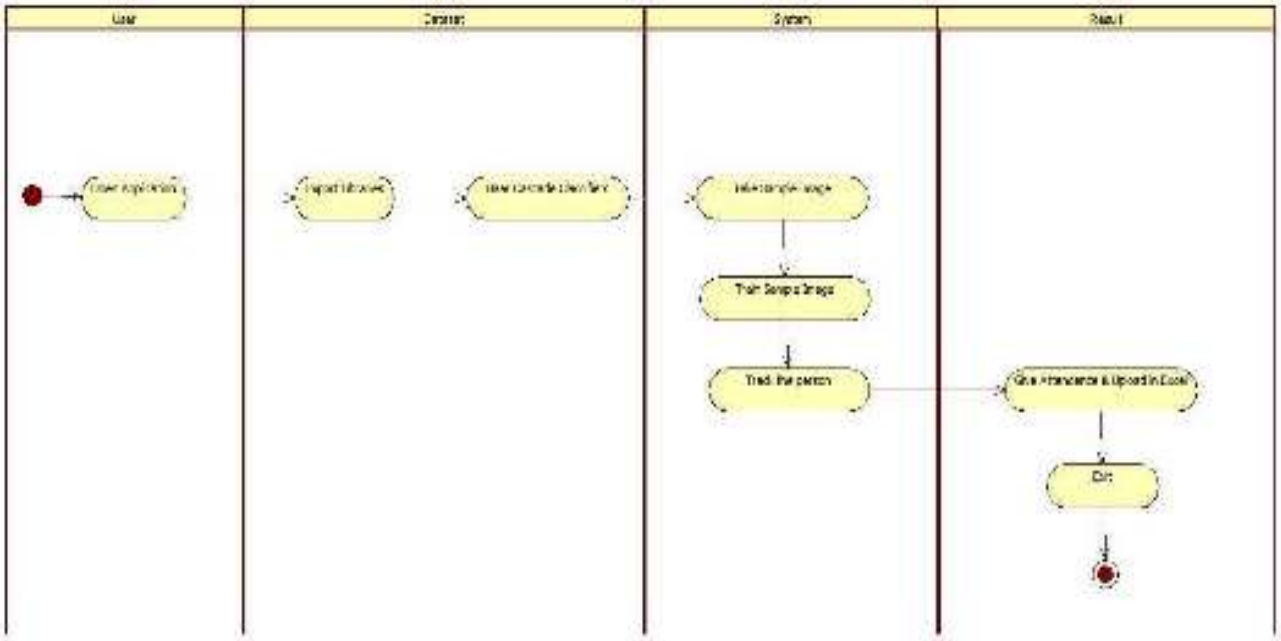

Figure 3.5: Activity Diagram for Face Mask Detection using CNN

3.6 Sequence diagram:

A sequence diagram represents the interaction between different objects in the system. The important aspect of a sequence diagram is that it is time-ordered. This means that the exact sequence of the interactions between the objects is represented step by step. Different objects in the sequence diagram interact with each other by passing "messages"

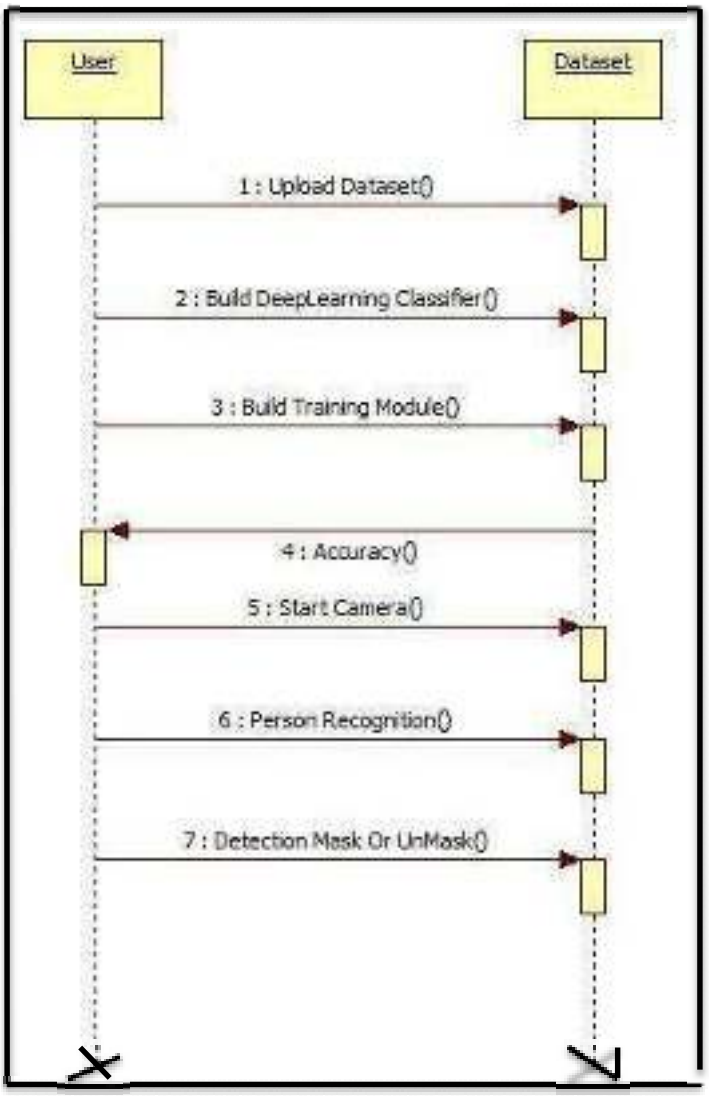

Figure 3.6: Sequencee Diagram for Face Mask Detection using CNN

4.IMPLEMENTATION

4.1 SAMPLE CODE

```
import sys

if sys.platform == 'linux': import
Xlib.threaded
else:

import Xlib.threaded

from flask import Flask, render_template, Response, request from
camera_desktop import Camera
app = Flask(_name_) @app.route('/')
def index():

return render_template('index.html' def
gen(camera):
while True:

frame = camera.get_frame()

yield (b'--frame\r\n' + b'Content-Type: image/jpeg\r\n\r\n' + frame +
b'\r\n\r\n') @app.route('/video_feed')
def video_feed():
return Response(gen(Camera()), mimetype='multipart/x-mixed-replace; boundary=frame')

if__name_____== "_main_":
app.run(host='127.0.0.1', threaded=True)
import time

try

from greenlet import getcurrent as get_ident except
ImportError:
try:

from thread import get_ident except
ImportError:
from _thread import get_ident class
CameraEvent(object):
"""An Event-like class that signals all active clients when a new frame is available.
"""

def_init_(self): self.events = {}
def wait(self):
```

"""Invoked from each client's thread to wait for the next frame."""
ident =
get_ident()
if ident not in self.events: # this is a new
client
add an entry for it in the self.events dict

each entry has two elements, a threading.Event() and a timestamp self.events[ident] =
[threading.Event(), time.time()]
return self.events[ident][0].wait() def set(self):
"""Invoked by the camera thread when a new frame is available."""
now =
time.time()

remove = None

for ident, event in self.events.items(): if not
event[0].isSet():
if this client's event is not set, then set it # also update the last
set timestamp to now event[0].set()
event[1] = now else:
# if the client's event is already set, it means the client # did not process a
previous frame
if the event stays set for more than 5 seconds, then assume # the client is gone
and remove it
if now - event[1] > 5: remove = ident
if remove:

del self.events[remove] def clear(self):
"""Invoked from each client's thread after a frame was processed."""
self.events[get_ident()][0].clear()
class BaseCamera(object):

thread = None # background thread that reads frames from camera frame = None
# current frame is stored here by background thread last_access = 0 # time of last
client access to the camera
event = CameraEvent()

def __init__(self):

"""Start the background camera thread if it isn't running yet."""
if
BaseCamera.thread is None:
BaseCamera.last_access = time.time()

start background frame thread

BaseCamera.thread = threading.Thread(target=self._thread)

```
BaseCamera.thread.start()

# wait until frames are available while
self.get_frame() is None:
time.sleep(0)


def get_frame(self):

"""Return the current camera frame."""
BaseCamera.last_access = time.time()


# wait for a signal from the camera thread
BaseCamera.event.wait() BaseCamera.event.clear()

return BaseCamera.frame


@staticmethod def frames():


"""Generator that returns frames from the camera."""
raise RuntimeError('Must be implemented by subclasses.')


@classmethod def _thread(cls):
"""Camera background thread."""
print('Starting camera thread.')
frames_iterator = cls.frames()
for frame in frames_iterator:
BaseCamera.frame = frame
BaseCamera.event.set() # send signal to clients time.sleep(0)


# if there hasn't been any clients asking for frames in # the last 10
seconds then stop the thread
if time.time() - BaseCamera.last_access > 10:
frames_iterator.close()
print('Stopping camera thread due to inactivity.') break
BaseCamera.thread = None # USAGE
# python train_mask_detector.py --dataset dataset # import the
necessary packages
from tensorflow.keras.preprocessing.image import ImageDataGenerator from
tensorflow.keras.applications import MobileNetV2
```

```
aug.flow(trainX, trainY, batch_size=BS),
steps_per_epoch=len(trainX) // BS, validation_data=(testX,
testY), validation_steps=len(testX) // BS,
epochs=EPOCHS)
# make predictions on the testing set
print("[INFO] evaluating network...")
predIdxs = model.predict(testX, batch_size=BS)

# for each image in the testing set we need to find the index of the # label with
corresponding largest predicted probability
predIdxs = np.argmax(predIdxs, axis=1)

# show a nicely formatted classification report
print(classification_report(testY.argmax(axis=1), predIdxs,
target_names=lb.classes_)) # serialize the
model to disk
print("[INFO] saving mask detector model...")
model.save(args["model"], save_format="h5") # plot the
training loss and accuracy
N = EPOCHS

plt.style.use("ggplot") plt.figure()
plt.plot(np.arange(0, N), H.history["loss"], label="train_loss")
plt.plot(np.arange(0, N), H.history["val_loss"], label="val_loss")
plt.plot(np.arange(0, N), H.history["accuracy"], label="train_acc")
plt.plot(np.arange(0, N), H.history["val_accuracy"], label="val_acc"
```

5.RESULTS

5.SCREENSHOTS

FACE MASK DETECTION IMAGE 1

Screenshot 5.1.1 : Face Mask Detection Image 1

It identifies a group of persons who wear a mask or not and classifies them into two categories

With Mask→GREEN RECTANGLE

Without Mask→RED RECTANGLE

FACE MASK DETECTION IMAGE 2

Screenshot 5.1.2 : Face Mask Detection Image 2

Screenshot 5.1.3 : Face Mask Detection Image 3

FACE MASK DETECTION IMAGE 4

Screenshot 5.1.4 : Face Mask Detection Image 4

FACE MASK DETECTION IMAGE 5

Screenshot 5.1.5 : Face Mask Detection Image 5

FACE MASK DETECTION IMAGE 6

Screenshot 5.1.6 : Face Mask Detection Image 6

FACE MASK DETECTION IMAGE 7

Screenshot 5.1.7 : Face Mask Detection Image 7

FACE MASK DETECTION IMAGE 8

Screenshot 5.1.8 : Face Mask Detection Image 8

6.TESTING

6.SYSTEM TESTING

6.1 INTRODUCTION TO TESTING

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub assemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner There are various types of tests. Each test type addresses a specific testing requirement.

6.2 TYPES OF TESTS

6.2.1 UNIT TESTING

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs All decision branches and internal code flow should be validated It is the testing of individual software units of the application .It is done after the completion of an individual unit before integration. This is a structural testing that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results

6.2.2 Integration testing

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfactory, as shown by successfully unit testing, the combination of

components are correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

6.2.3 Functional test

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items

Valid Input : identified classes of valid input must be accepted.

Invalid Input : identified classes of invalid input must be rejected.

Functions : identified functions must be exercised.

Output : identified classes of application outputs must be exercised

Systems/Procedures : interfacing systems or procedures must be invoked

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identifying Business process flows, data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

Unit Testing

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

Test strategy and approach

Field testing will be performed manually and functional tests will be written in detail

Test objectives

- All field entries must work properly.
- Pages must be activated from the identified link
- The entry screen, messages and responses must not be delayed

Features to be tested

- Verify that the entries are of the correct format
- No duplicate entries should be allowed
- All links should take the user to the correct page.

Integration Testing

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects

The task of the integration test is to check that components or software applications e.g. components in a software system or – one step up – software applications at the company level – interact without error.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

Test Cases :

USER REQUIREMENTS:

Test case ID	Test case name	Purpose	Test Case	Output
1	Detecting a persor whether he is wearing a mask	Use it for detection	If a person wears a mask	Green Rectangle is displayed
2	Detecting a persor whether he is not wearing mask	Use it for detection	If a person doesn't wear a mask	Red Rectangle is displayed

7.CONCLUSION

7.CONCLUSION & FUTURE SCOPE

7.1 CONCLUSION

Face detection is performed and Using the two labels which were declared one for color and the other for title, faces in the image are classified as with mask or without mask. Accuracy percentage is displayed in white color at the top of the image.

7.2 FUTURE ENHANCEMENTS

The present model proposed gives great accuracy for a single face with and without a mask For multiple faces it also gives quite good accuracy. It works easily on any mobile device just by switching on the video stream with no external hardware requirement Further we will work for improving the accuracy for multiple face mask detection, to classify the faces into three categories that is. With mask, without mask. Improper mask instead of just the two with and without mask class by adding datasets with images of people wearing masks not covering their noses properly and also to detect the masked face using the FaceNet model of Convolutional Neural Network like in so as to further improve our model and add marking attendance feature in it by detecting the face even when the mask is on.

8.BIBLIOGRAPHY

BIBLIOGRAPHY

8.1 REFERENCES

Journals:

1. Programming Python, Mark Lutz
2 Head First Python, Paul Barry
3. Core Python Programming, R. Nageswara Rao
4. Learning with Python, Allen B. Downey

BOOKS:

[1] K. He X. Zhang S. Ren and J. Sun "Deep residual learning for image recognition" 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) pp. 770- 778 2016

[2] K. Li G. Ding and H. Wang "L-fcn: A lightweight fully convolutional network for biomedical semantic segmentation" 2018 IEEE International Conference on Bioinformatics and Biomedicine (BIBM) pp. 2363-2367 Dec 2018.

[3] S. Ge, J. Li, Q. Ye and Z. Luo, "Detecting Masked Faces in the Wild with LLE- CNNs,' 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, 2017, pp. 426-434, doi 10.1109/CVPR.2017.53

[4] J. Zheng, R. Ranjan, C. Chen, J. Chen, C. D. Castillo and R. Chellappa, "An Automatic System for Unconstrained Video-Based Face Recognition", *IEEE Transactions on Biometrics Behavior and Identity Science*, vol. 2, no. 3, pp 194-209, July 2020

[5] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features", *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, pp. I-I, 2001.

[6] G. T. S. Draughon, P. Sun and J. P. Lynch, "Implementation of a Computer Vision Framework for Tracking and Visualizing Face Mask Usage in Urban Environments", *2020 IEEE International Smart Cities Conference (ISC2)*, pp 1-8, 2020

[7] A. Oumina, N. El Makhfi and M. Hamdi, "Control The COVID-19 Pandemic: Face Mask Detection Using Transfer Learning", *2020 IEEE 2nd International Conference on Electronics Control Optimization and Computer Science (ICECOCS)*, pp. 1-5, 2020

[8] W. Han, Z. Huang, A. kuerban, M. Yan and H. Fu, "A Mask Detection Method for Shoppers Under the Threat of COVID-19 Coronavirus", *2020 International Conference on Computer Vision Image and Deep Learning (CVIDL)*, pp. 442-447, 2020

[9] S. Kanaga Suba Raja, A. Sathya, S. Karthikeyan and T. Janane, "Multi cloud-based secure privacy preservation of hospital data in cloud computing", *International Journal of Cloud Computing (Inderscience Enterprises Ltd)*, vol. 10, no. 1/2, pp. 101-111, 2021, [online] Available: https://doi.org/10.1504/IJCC.2021.10036376. ISSN 2043-9989.

[10] G. Deore, R. Bodhula, V. Udpikar and V. More, "Study of masked face detection approach in video analytics", *2016 Conference on Advances in Signal Processing (CASP)*, pp. 196-200, 2016

[11] W. Bu, J. Xiao, C. Zhou, M. Yang and C. Peng, "A cascade framework for masked face detection", *2017 IEEE International Conference on Cybernetics and Intelligent Systems (CIS) and IEEE Conference on Robotics Automation and Mechatronics (RAM)*, pp. 458-462, 2017.

[12] S. Ge, J. Li, Q. Ye and Z. Luo, "Detecting Masked Faces in the Wild with LLE-CNNs," 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, 2017, pp. 426-434, doi: 10.1109/CVPR.2017.53

8.2 WEBSITES:

1. https://www.w3schools.com/python/
2. https://www.tutorialspoint.com/python/index.htm
3. https://www.javatpoint.com/python-tutorial
4. https://www.learnpython.org/
5. https://www.pythontutorial.net/

8.3 GITHUB LINK

https://github.com/pavanvallapu23/facemaskdetectionusingconvolutionneuralnetwork