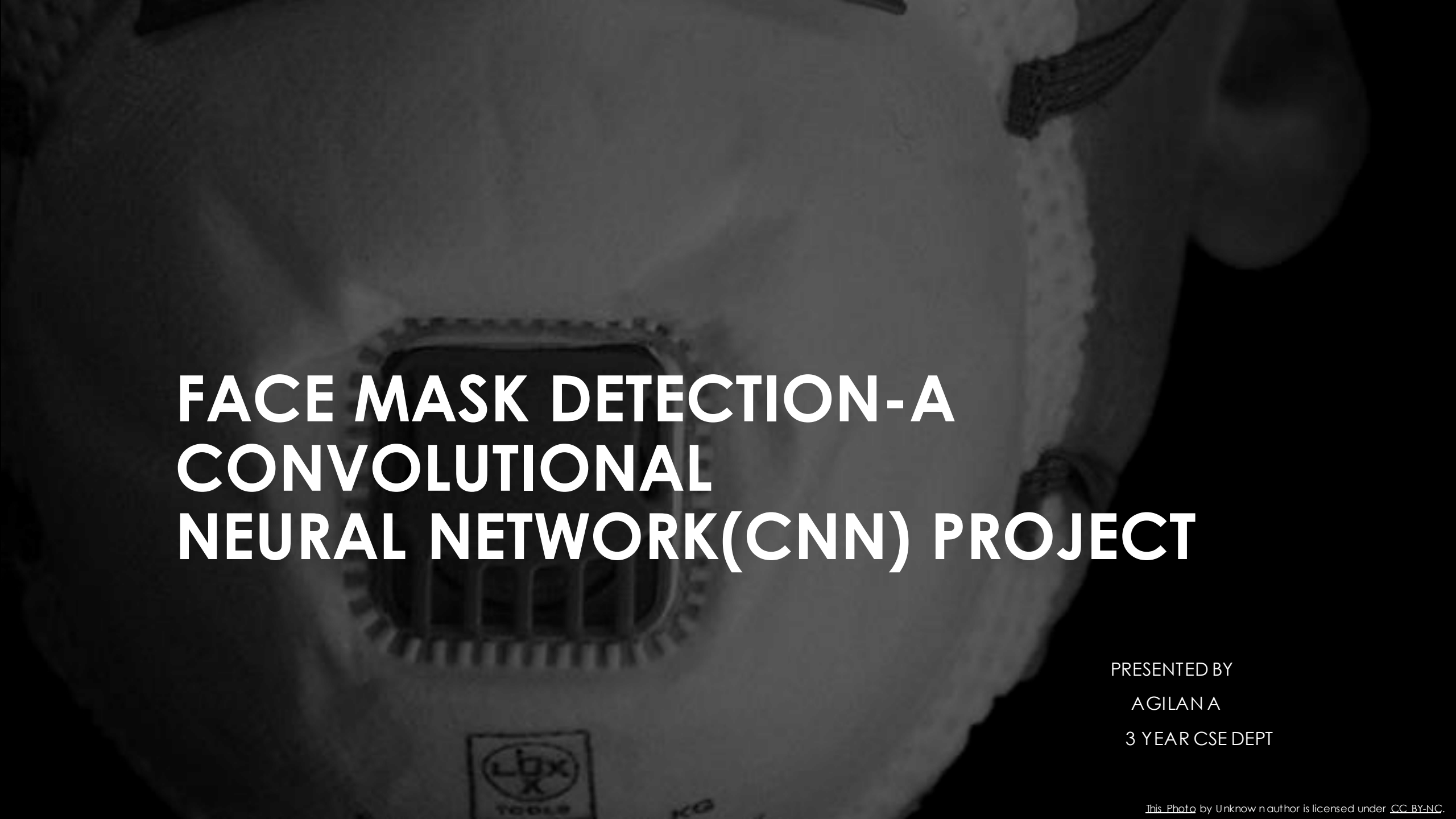


FACE MASK DETECTION-A CONVOLUTIONAL NEURAL NETWORK(CNN) PROJECT

PRESENTED BY

AGILAN A

3 YEAR CSE DEPT

AGENDA

- Introduction
- Problem Statement
- Proposed System/Solution
- System Development Approach
- Algorithm and Deployment
- Result
- Conclusion
- Reference

INTRODUCTION

- •In the realm of computer vision, face mask detection has emerged as a critical application. The goal is to determine whether an individual is wearing a face mask or not. We frame this task as a **binary classification problem**, where the model categorizes images into two classes: “with mask” or “without mask.”
- •The COVID-19 pandemic underscored the importance of preventive measures, including mask-wearing. However, human error often leads to incorrect mask usage—such as not covering the nose or mouth—which renders the mask ineffective. To address this, deep learning techniques, particularly CNNs, have been employed to detect masked faces effectively.

PROBLEM STATEMENT

- •The goal is to differentiate between images of people with and without masks.
- •A **Convolutional Neural Network (CNN)** is employed for this task.
- •The CNN achieves an impressive **98.2% accuracy on the training set** and **97.3% accuracy on the test set**.
- •The trained model's weights are then used to classify whether a person is wearing a mask or not in real time using **OpenCV**.
- •The model works efficiently with no noticeable lag time between wearing/removing a mask and displaying predictions.

PROPOSED SYSTEM/ SOLUTION

- **Binary Classification**:
- The most straightforward approach is binary classification:
- **Class 1**: Represents faces with masks.
- **Class 0**: Represents faces without masks.
- CNNs learn to distinguish between these two classes based on image features.

SYSTEM DEVELOPMENT APPROACH

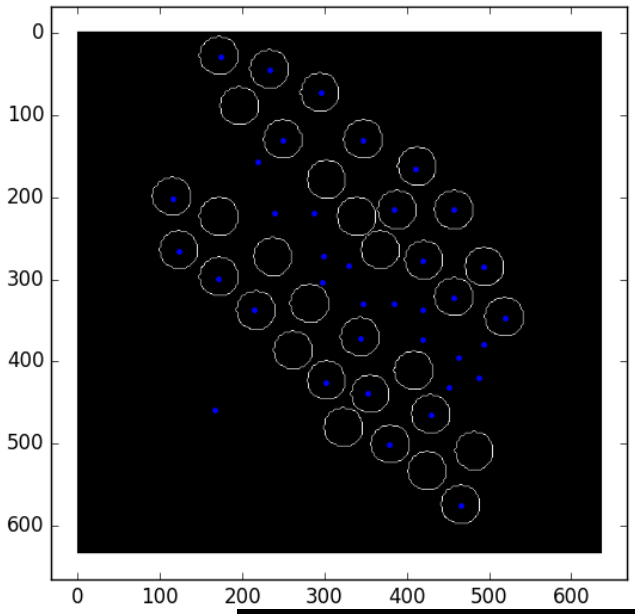
- **Hardware Requirements:**
- **Webcam**:
- A functional webcam is necessary for capturing real-time video streams.
- Ensure your system has a working webcam.

- **System Resources**:
- At least **4 GB RAM** (more is preferable) for smooth execution.
- Adequate storage space (e.g., 1TB hard disk).
- A 64-bit processor (e.g., Intel Core i5 or equivalent).

- **Operating System**:
- The software components mentioned above are compatible with various operating systems (Windows, Linux, macOS).

- **Power Supply**:
- 1.Ensure a stable power supply to avoid interruptions during model training or inference.

SYSTEM DEVELOPMENT APPROACH (CONT.)



- **Software Requirements:**
- **Python** (version 3.x):
- Python is essential for implementing the face mask detection model.
- Install Python on your system.

- **OpenCV** (version 4.4.0 or higher):
- OpenCV provides tools for image processing, computer vision, and real-time video analysis.
- Install OpenCV using pip install opencv-python.

- **NumPy** (version 1.19.3 or compatible):
- NumPy is used for numerical computations and array manipulation.
- Install NumPy using pip install numpy.

- **TensorFlow** (version 2.5.0 or compatible):
- TensorFlow is a deep learning framework.
- Install TensorFlow using pip install tensorflow.

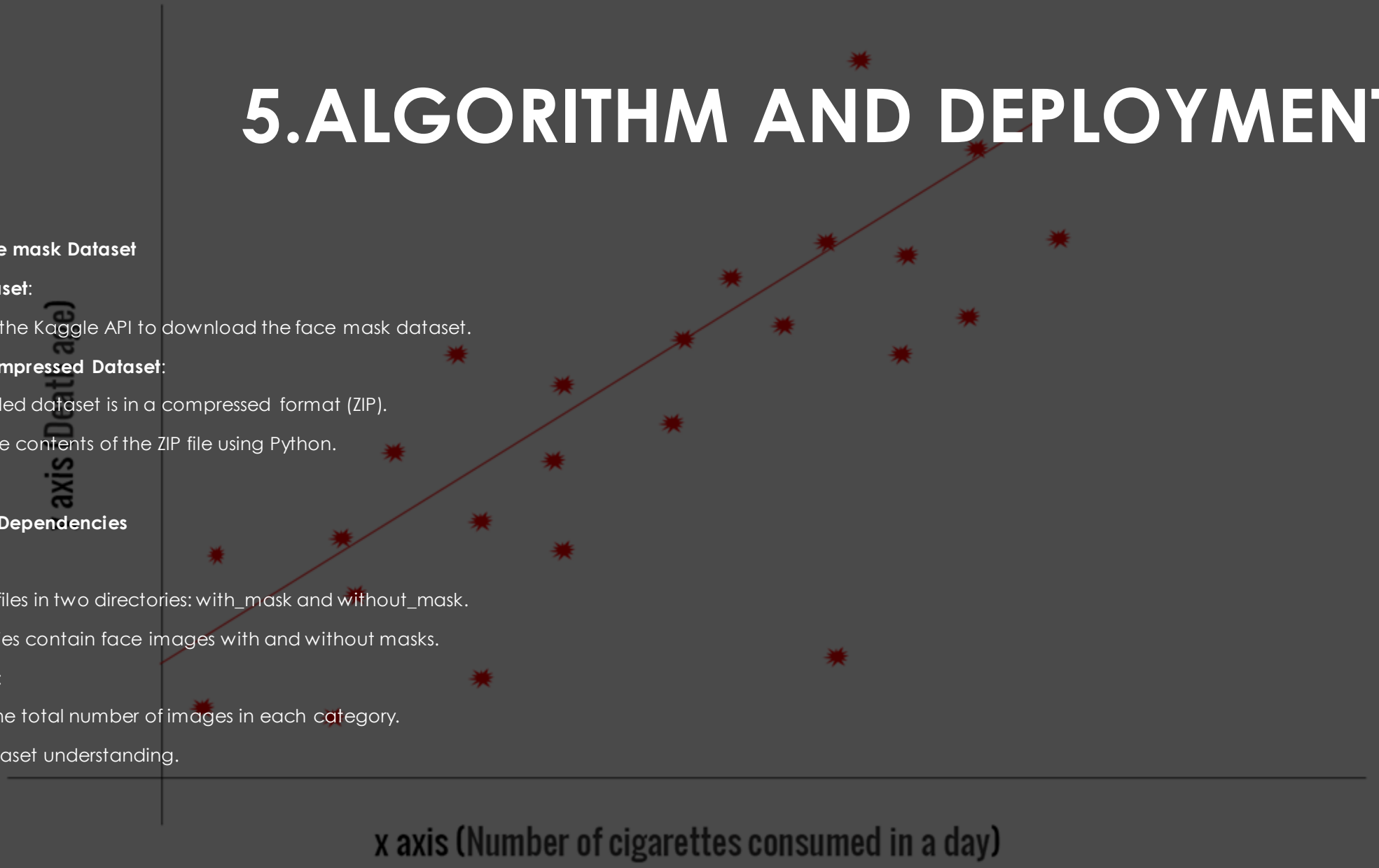
SYSTEM DEVELOPMENT APPROACH (CONT.)

- **Software Requirements(Cont.):**
- **Scikit-learn** (version 0.24.2 or compatible):
- Scikit-learn provides machine learning tools.
- Install Scikit-learn using pip install scikit-learn.
- **MediaPipe** (version 0.8.5 or compatible):
- MediaPipe offers solutions for face detection and other computer vision tasks.
- Install MediaPipe using pip install mediapipe.
- **Tqdm** (version 4.60.0 or compatible):
- Tqdm provides progress bars for loops and tasks.
- Install Tqdm using pip install tqdm.
- **Google Colab:** These platforms can be used for interactive development, experimentation, and documentation.
- These requirements will help you set up a system capable of running a face mask detection project using CNN binary classification.

5.ALGORITHM AND DEPLOYMENT

- **Importing Face mask Dataset**
- **Fetch the Dataset**:
- You are using the Kaggle API to download the face mask dataset.
- **Extract the Compressed Dataset**:
- The downloaded dataset is in a compressed format (ZIP).
- You extract the contents of the ZIP file using Python.

- **Importing the Dependencies**
- **List Files:**
- The code lists files in two directories: with_mask and without_mask.
- These directories contain face images with and without masks.
- **Count Images**:
- It calculates the total number of images in each category.
- Helpful for dataset understanding.



ALGORITHM AND DEPLOYMENT (CONT.)

- **Creating Labels for the two class of Images:**
- with mask --> 1
- without mask --> 0

.Image Processing :

- Resize the Images.
- Convert the images to NumPy arrays.

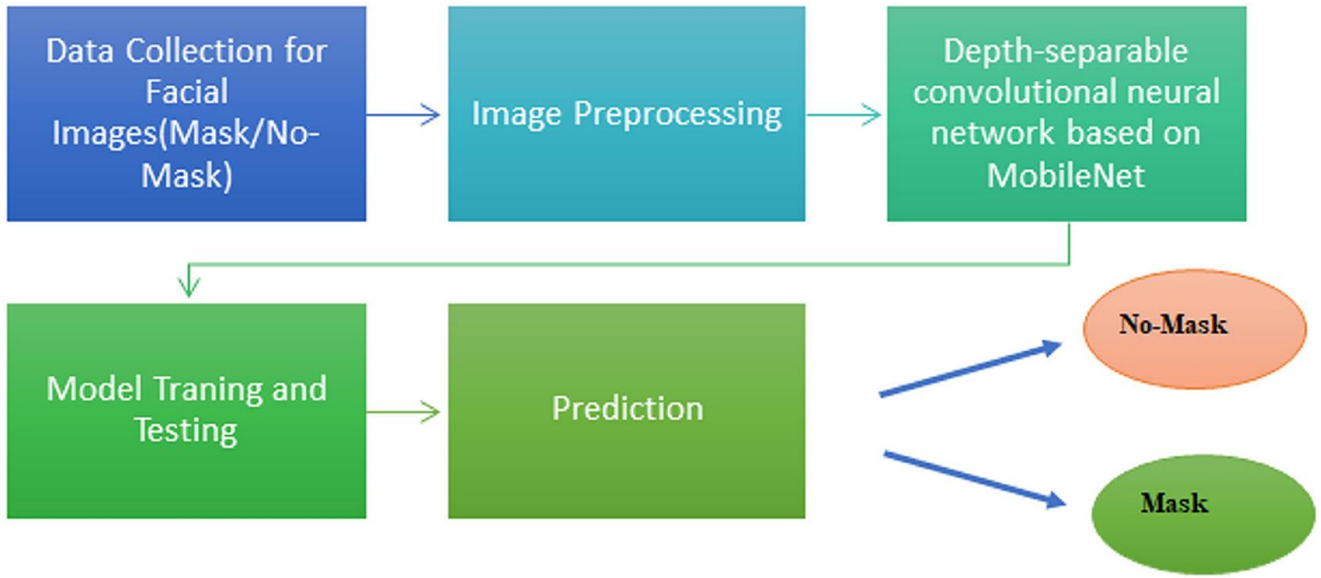
- **Train Test Split.**

- **Building a Convolutional Neural Networks (CNN).**

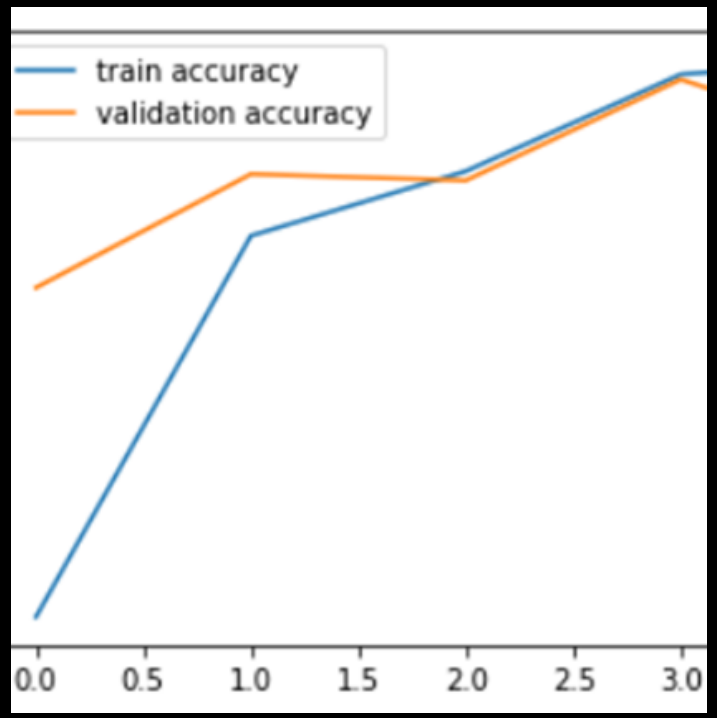
- **Model Evaluation.**

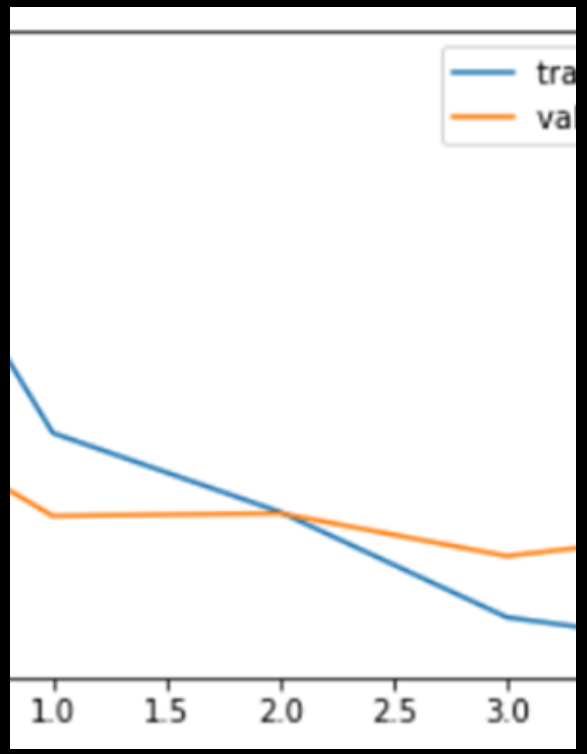
- **Building a Predictive System.**

FLOW CHART


Data Collection for Facial Images(Mask/No-Mask)
Image Preprocessing
Depth-separable convolutional neural network based on MobileNet
Model Traning and Testing
Prediction
No-Mask
Mask

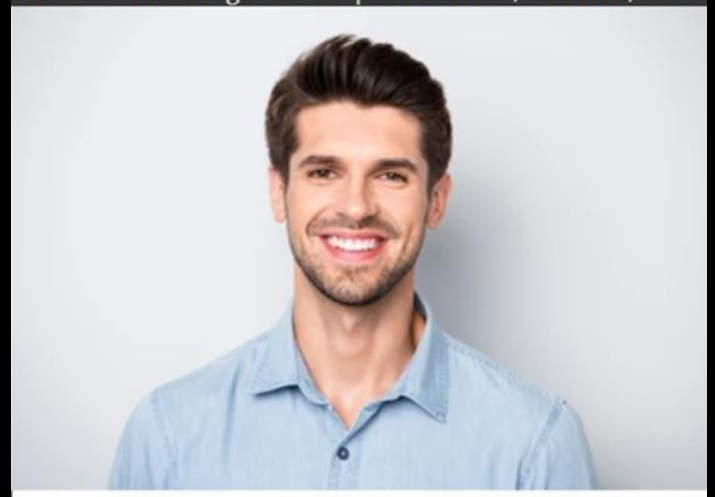
RESULT


train accuracy
validation accuracy
0.0
0.5
1.0
1.5
2.0
2.5
3.0


1.0
1.5
2.0
2.5
3.0

RESULT(CONT.)

```
Path of the image to be predicted: /content/test.
```




```
1/1 [==============================] - 0s 21ms/st
[[0.49811754 0.47740024]]
0
The person in the image is not wearing a mask
```

```
o be predicted: /content/test.png
```

```
==========] - 0s 176ms/step
7454]]

mage is wearing a mask
```

CONCLUSION

- Developing a custom **convolutional neural network (CNN)** for face mask detection is crucial for public health efforts. Trained on 12,000 mask images, the model achieved impressive accuracy during both training and testing. Its exceptional precision, recall, and F1 score contribute significantly to disease prevention, particularly during the COVID-19 pandemic.

REFERENCE

- •https://www.python.org/
- •https://www.tensorflow.org/
- •https://keras.io/
- •https://numpy.org/
- •https://matplotlib.org/
- •https://scikit-learn.org/
- •https://opencv.org/
- •https://www.kaggle.com/datasets/omkargurav/face-mask-dataset