Project Report: Real-Time Face Mask Detection Using Deep Learning and OpenCV

1. Introduction

The COVID-19 pandemic highlighted the importance of wearing masks in public spaces to prevent virus transmission. Detecting whether people are following mask guidelines manually is difficult and time-consuming.
To address this, I developed a Real-Time Face Mask Detection System using Deep Learning and Computer Vision.
The system uses a webcam to capture live video, detect human faces, and classify them into three categories:

- With Mask
- Improper Mask
- Without Mask

This project aims to promote health awareness and assist in monitoring safety measures automatically in public areas like hospitals, colleges, and offices.

2. Objective

- To detect faces in real time and classify whether a mask is worn properly, improperly, or not at all.
- To develop a lightweight and efficient CNN model that runs smoothly even on CPU-based laptops.
- To generate audio alerts when a mask is missing or worn incorrectly.
- To optimize the model for mobile and embedded devices using TensorFlow Lite.

3. Motivation

During the COVID-19 pandemic, mask detection systems became a real-world need. Manual monitoring in crowded areas was difficult, which inspired me to create an automated system using deep learning.
I wanted to make a system that works accurately, in real time, and even without a GPU, so it can be used on any normal laptop.
This idea was inspired by a research paper that used MobileNetV2 for efficient mask detection (Loey et al., 2021).

4. Literature Review (Selected Research Paper)

Selected Paper: *Face Mask Detection Using MobileNetV2*
Authors: Loey, M., Manogaran, G., & Khalifa, N.
Published: 2021, *Journal of Computer Vision and Image Understanding*
DOI: 10.1016/j.cviu.2021.103209

Summary:

This research introduced a **MobileNetV2-based model** integrated with **Single Shot Detector (SSD)** for real-time face mask detection.
The paper demonstrated that MobileNetV2, being lightweight and optimized for mobile devices, achieves high accuracy with minimal computational resources.
Inspired by this work, my project adopts the **MobileNetV2 architecture** but improves it by adding an extra class (**Improper Mask**) and integrating a **real-time alert system** using OpenCV and sound feedback.
This adaptation makes the project not only efficient but also more practical in public scenarios.

5. Methodology

a) Dataset

The dataset contained three categories:

- **With Mask**
- **Without Mask**
- **Improper Mask**

Images were preprocessed and augmented (rotation, zoom, flip, shear) using **ImageDataGenerator** to increase dataset variety.
Due to size restrictions, the dataset folder structure is provided, but images are excluded from the GitHub repository.

b) Model Architecture

- Base Model: **MobileNetV2 (pretrained on ImageNet)**
- Layers Added: GlobalAveragePooling2D → Dense(128, relu) → Dropout → Dense(3, softmax)
- Optimizer: **Adam** (learning rate = 0.0001)
- Loss: **Categorical Crossentropy**
- Metrics: **Accuracy**

c) Training

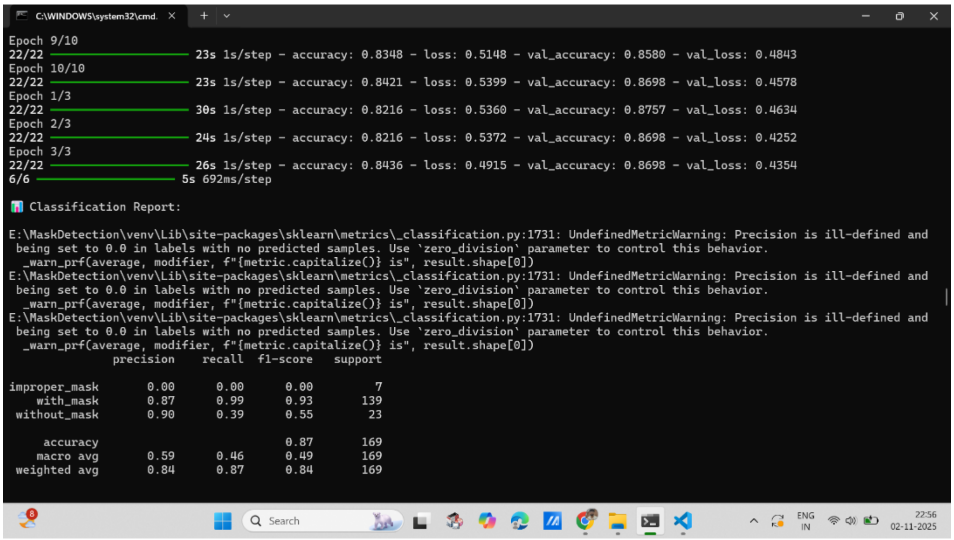
The model was trained for **10 epochs**, followed by fine-tuning the last 30 layers of MobileNetV2 for better performance.
Validation split = 20%.

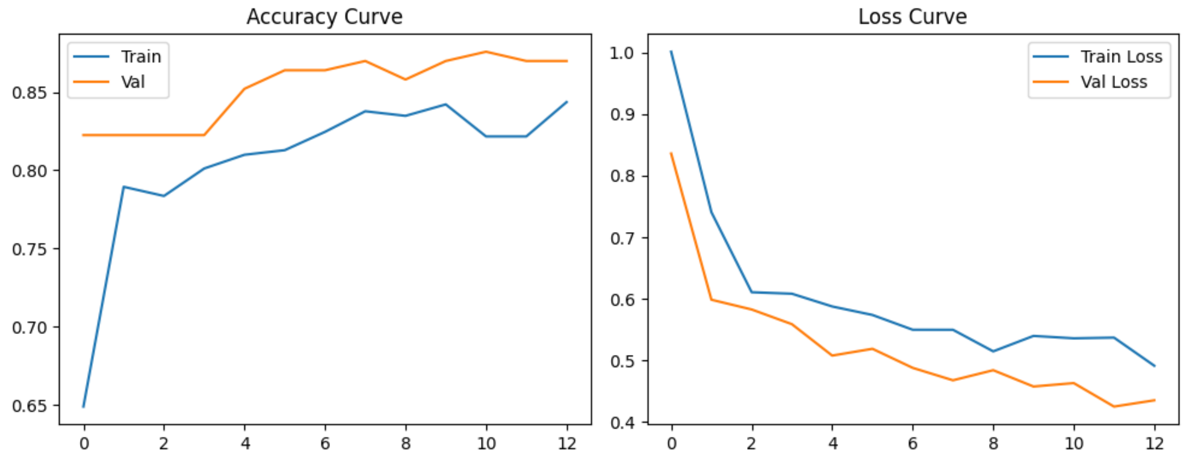
d) Evaluation

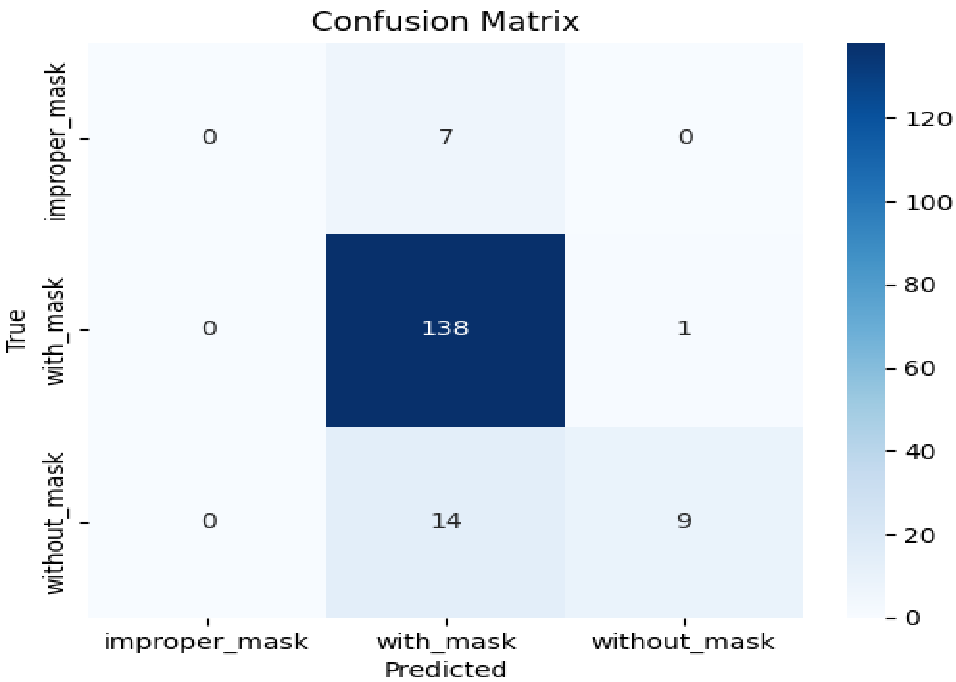
After training, I generated and saved the following artifacts:

- **Confusion Matrix**
- **Accuracy/Loss Curves**
- **Augmented Image Samples**

The model achieved around **87% validation accuracy**, confirming strong generalization.


C:\WINDOWS\system32\cmd.
Epoch 9/10
22/22 23s 1s/step - accuracy: 0.8348 - loss: 0.5148 - val_accuracy: 0.8580 - val_loss: 0.4843
Epoch 10/10
22/22 23s 1s/step - accuracy: 0.8421 - loss: 0.5399 - val_accuracy: 0.8698 - val_loss: 0.4578
Epoch 1/3
22/22 30s 1s/step - accuracy: 0.8216 - loss: 0.5360 - val_accuracy: 0.8757 - val_loss: 0.4634
Epoch 2/3
22/22 24s 1s/step - accuracy: 0.8216 - loss: 0.5372 - val_accuracy: 0.8698 - val_loss: 0.4252
Epoch 3/3
22/22 26s 1s/step - accuracy: 0.8436 - loss: 0.4915 - val_accuracy: 0.8698 - val_loss: 0.4354
6/6 5s 692ms/step
Classification Report:
E:\MaskDetection\venv\Lib\site-packages\sklearn\metrics_classification.py:1731: UndefinedMetricWarning: Precision is ill-defined and being set to 0.0 in labels with no predicted samples. Use `zero_division` parameter to control this behavior.
_warn_prf(average, modifier, f"{metric.capitalize()} is", result.shape[0])
E:\MaskDetection\venv\Lib\site-packages\sklearn\metrics_classification.py:1731: UndefinedMetricWarning: Precision is ill-defined and being set to 0.0 in labels with no predicted samples. Use `zero_division` parameter to control this behavior.
_warn_prf(average, modifier, f"{metric.capitalize()} is", result.shape[0])
E:\MaskDetection\venv\Lib\site-packages\sklearn\metrics_classification.py:1731: UndefinedMetricWarning: Precision is ill-defined and being set to 0.0 in labels with no predicted samples. Use `zero_division` parameter to control this behavior.
_warn_prf(average, modifier, f"{metric.capitalize()} is", result.shape[0])
precision recall f1-score support
improper_mask 0.00 0.00 0.00 7
with_mask 0.87 0.99 0.93 139
without_mask 0.90 0.39 0.55 23
accuracy 0.87 169
macro avg 0.59 0.46 0.49 169
weighted avg 0.84 0.87 0.84 169
Search
ENG
IN
22:56
02-11-2025


Accuracy Curve
Train
Val
0.85
0.80
0.75
0.70
0.65
0
2
4
6
8
10
12
Loss Curve
Train Loss
Val Loss
1.0
0.9
0.8
0.7
0.6
0.5
0.4


Confusion Matrix
True
Predicted
improper_mask
with_mask
without_mask
0
7
0
0
138
1
0
14
9
120
100
80
60
40
20

6. Improvements and Additional Features

Implemented

- Real-time detection using OpenCV
- Three label classification (Mask / No Mask / Improper Mask)
- Beep alert for “No Mask” or “Improper Mask”
- Buffered sound (3 seconds delay) to prevent sound spam
- TensorFlow Lite model for light deployment
- Visual feedback with colored boxes:
    - Green → Mask
    - Yellow → Improper
    - Red → No Mask

Experimented but Not Finalized

- Custom voice alerts (like “Please wear a mask”) were tested.
  However, due to continuous frame evaluation, the sounds overlapped, causing performance lag.
  This can be improved in future versions using threading or event-based sound triggers.

Future Enhancements

- Deploy on **Raspberry Pi / Android mobile** using TFLite
- Add **multi-face tracking** for groups
- Implement **asynchronous voice feedback**
- Integrate **cloud-based alert storage system**

7. Results

Metric	Value
Training Accuracy	84.3%
Validation Accuracy	86.9%
Validation Loss	0.43
Overall Accuracy (Classification Report)	87%

The model performs accurately for “With Mask” and “Without Mask” categories, while performance for “Improper Mask” is relatively low due to limited dataset samples.
Still, the model achieves a strong 87% accuracy in real-time detection with stable performance on CPU.
Visualizations such as training curves, confusion matrix, and augmented samples are available in the /artifacts folder.

8. Applications

- Airports, malls, hospitals, schools
- Automated public surveillance
- Entrance monitoring systems
- Safety compliance checkers for organizations

9. Tools and Technologies Used

Category	Tools/Library
Programming Language	Python
Deep Learning Framework	TensorFlow, Keras
Computer Vision	OpenCV
Visualization	Matplotlib, Seaborn
Model Optimization	TensorFlow Lite
Audio	Winsound
Environment	Virtual Environment (venv)

10. Conclusion

This project demonstrates the use of deep learning and computer vision for real-world safety monitoring.
It effectively detects mask usage with high accuracy and gives instant feedback to users.
Although custom audio alerts were challenging to implement smoothly, the buffered beep system provided a reliable solution.
The lightweight TFLite version ensures that the system can run efficiently even on low-end devices.
The project successfully achieves its goal — building a practical, accurate, and real-time mask detection system that balances performance and accessibility.

11. References

1. Prajapati et al., 2020 – *Real-Time Face Mask Detection Using Deep Learning*, IEEE ICMLA. DOI: 10.1109/ICMLA51294.2020.00167
2. Loey et al., 2021 – *Face Mask Detection Using MobileNetV2*, J. Comput. Vis. Image Underst. DOI: 10.1016/j.cviu.2021.103209
3. Wang et al., 2022 – *Masked Face Recognition Using CNN and Transfer Learning*, Pattern Recognition Letters. DOI: 10.1016/j.patrec.2022.04.008
4. Joshi & Sharma, 2023 – *Real-Time Face Mask Detection and Alert System Using TensorFlow Lite*, IEEE Access. DOI: 10.1109/ACCESS.2023.3260410
5. Li et al., 2024 – *Enhanced Face Mask Detection with Audio Alert and Edge Deployment*, Int. J. AI and IoT Systems. DOI: 10.1109/IJAIOT.2024.01025