

**Computer Vision Project Report On**

**Face Mask Detection**

**Team Members** –

Rohit Narganve[1601000136]

Rajat Verma[2141005288]

Under the Guidance of

Asst. Prof. Ayesha Choudhary

**Introduction**

In order to protect ourselves from the COVID-19 Pandemic, the foremost thing for every one of us tend to wear a face mask.

As we don’t have danger against one type of virus, as viruses have tendency to mutate themselves according to the physical conditions. So, one should increase their immune power through more involvement in physical exercises like Yoga and follow the COVID protocols.

It becomes increasingly necessary to check if the people in the crowd wear face masks in most public gatherings such as Malls, Theatres, Parks, Airports, Hospitals, Offices and Educational Departments.

The development of a solution to detect if the person is wearing a face mask and allow their entry would be of great help to the society.

This developed model can be used to develop a full-fledged software to scan every person before they can enter the public gathering.

Where there is a rule, there are some who do not follow the rules and are a liability to the society so, they should be warned followed by punishment.

So, for the security purpose and Covid-19 prevention, face mask detection is a critical and challenging task.

**Motivation**

Hence many countries following the rule like “No entry without mask”.

In the case of medical field, mask reduces potential exposure risk from an infected person whether they have symptoms or not.

The face recognition without mask is easier but face recognition with mask is critical one because feature extraction of masked face is very complicated than normal face.

That is so many face features such as nose, mouth and chin are absent in the masked face.

In medical field, mask reduces potential exposures risk from an infected person whether they have symptoms or not. So many faces mask detection can be concentrated in two steps.

1) Face Recognition

2) Feature Extraction

Mainly there is a problem such as detecting the multiple mask and unmasked faces in an image. It can be solved by using a traditional object detection algorithm.

Several techniques are analysed for checking a person wear mask or not and identify the face recognition of a person.

**Challenges**

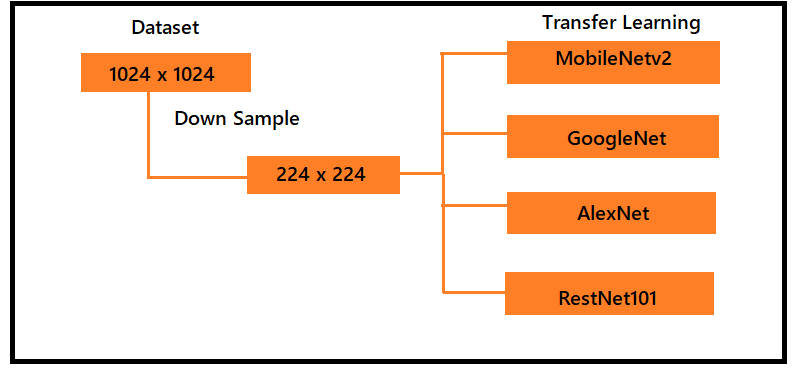
In the Machine Learning model, we have two phases – Training phase and recall phase. According to the **Free-Lunch theorem**, recall phase is predicting the result i.e., person wearing mask or not, which is easy step, so to compensate the easiness of one phase, another phase i.e., Training phase is over-difficult to implement it.

In training phase, we have to create a model that can serve our purpose.

We ease our work of creating the desired model by using the base model **[MobileNet -version 2]** from Tensorflow library and modifying some of its layers according our requirement of this project.

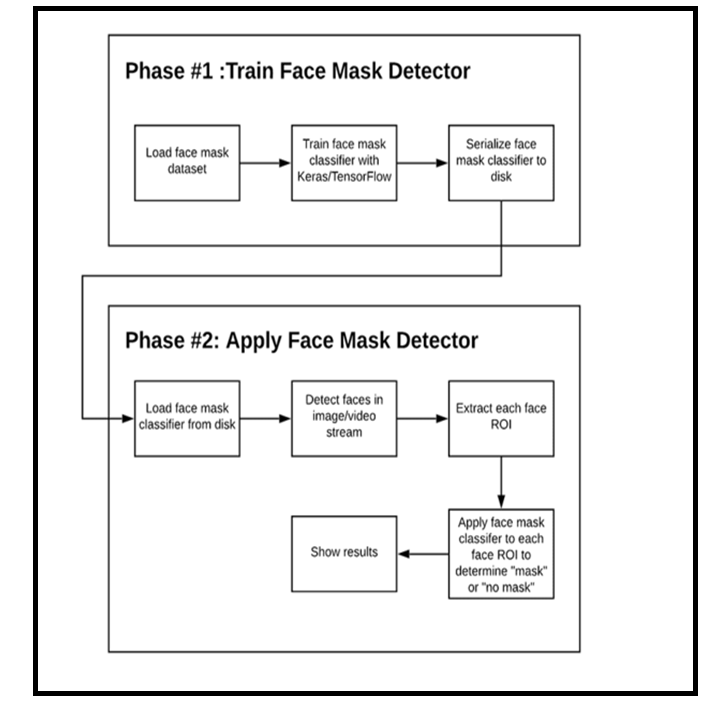
**State-of-the-art**

Main focus of project depends on Tensorflow.Keras allows us to data augmentation, loading the MobileNet classifier, pre-processing of the images and loading image data in array form.



Above figure is describing the pre-processing the images in pre-defined form.

Below figure is describing the phases of the discussed project.



**Proposed work**

Given the trained COVID-19 face mask detector, we’ll proceed to implement two Python scripts used to:

* Detect COVID-19 face masks in images
* Detect face masks in real-time video streams

We apply face detection to compute the bounding box location of the face in the image:

* Once we know where in the image the face is, we can extract the face Region of Interest (ROI).
* We apply facial landmarks, allowing us to localize the eyes, nose, mouth, etc.

Facial landmarks allow us to automatically infer the location of facial structures:

* Eyes
* Eyebrows
* Nose
* Mouth
* Jawline

We are reviewing two Python scripts in this tutorial:

1. Model\_Creation.ipynb - A model is trained using given datasets.
2. Face\_Mask\_Detection.ipynb - Using your webcam, this script applies face mask detection to every frame in the stream.

Need of Tensorflow import allow to:

* Loading the MobilNetV2 classifier (we will fine-tune this model with pre-trained [MobileNet](http://www.image-net.org/) weights)
* Building a new fully-connected layer
* Pre-processing of images
* Loading image data

Training of the model involves:

* + - Grabbing all of the paths of the images in the dataset [Dataset link – [ImageDataset](https://arxiv.org/abs/2008.08016)].
    - Initializing data and labels.
    - Looping over the images path.

Pre-processing steps include resizing to 224×224 pixels, conversion to array format, and scaling the pixel intensities in the input image to the range [-1, 1].

Appending the pre-processed image and associated

label to the data and labels respectively.

Ensuring our training data is in NumPy array format.

Fine-tuning setup of the base model is a three-step process:

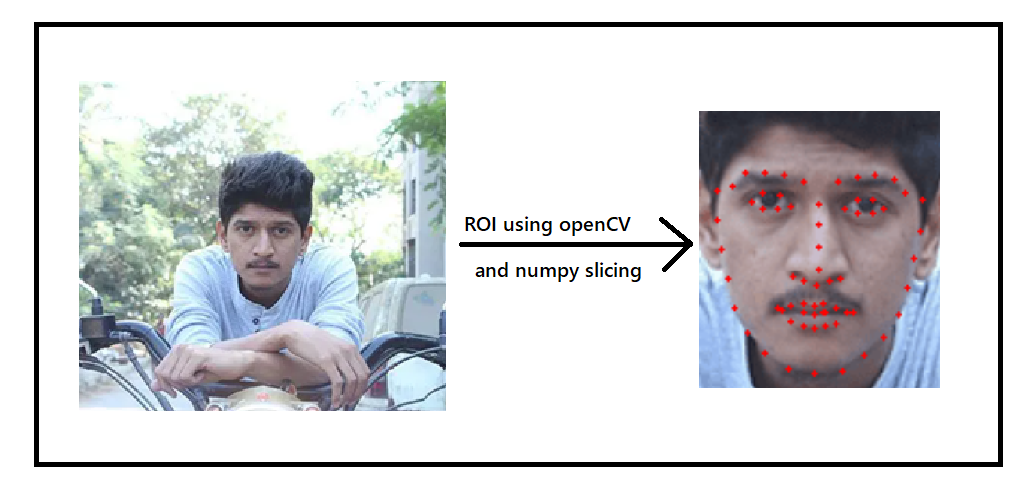
* Load MobileNet with pre-trained MobileNet weights, leaving off head of network.
* Construct a new FC head, and append it to the base in place of the old head.
* Freeze the base layers of the network. The weights of these base layers will not be updated during the process of backpropagation, whereas the head layer weights will be tuned.

Face mask detector is trained through -

1. Load the images from disk.
2. Detect faces in the image.
3. Apply the face mask detector to classify the face as either ‘with\_mask’ or ‘without\_mask’.

We will run the face ROI through our trained model:

* + Extract the face ROI via NumPy slicing.
  + Pre-process the ROI the same way we did during training.
  + Perform mask detection to predict with\_mask or without\_mask.
  + We will annotate and display the result.



We determine the class label based on probabilities returned by the mask detector model and assign an associated color for the annotation.

The color will be “green” for with\_mask and “red” for without\_mask.

**Experimental Results**

Total dataset – 1950

Masked dataset – 950

Non-masked dataset – 1000

Loss = "binary\_crossentropy", optimizer = "adam", metrics = ["accuracy"]

Epochs = 1, validation split = 0.1

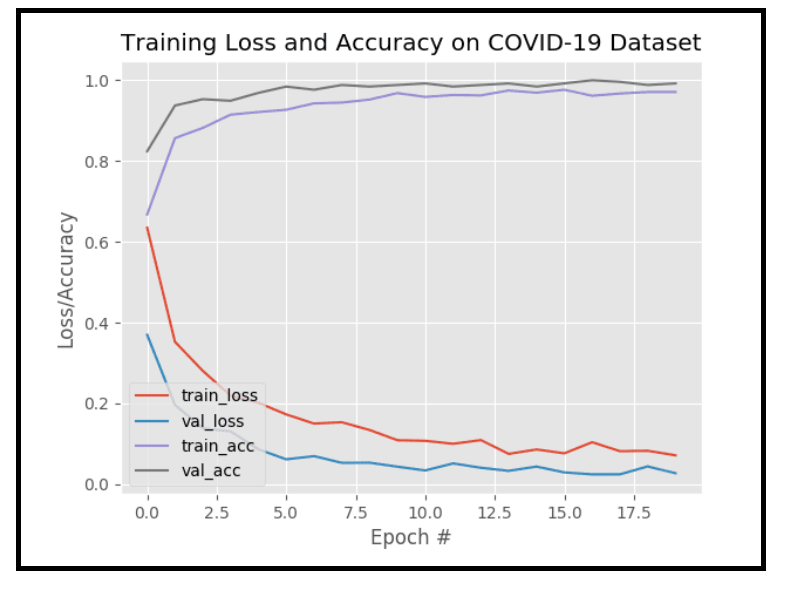
Time-taken to train modified model = 187seconds[3s/step]

Loss – 0.1750[17.50%]

Accuracy – 0.9840[98.40%]

Val\_loss – 0.1617[16.17%]

Val\_accuracy – 0.98[98.00%]



**Conclusion**

The artificial intelligence (AI) and machine learning (ML) are developing various models for face mask detection.

As we know nowadays mask detection is a very challenging task. The applications of Facial Mask Detection are used especially for the prevention of spreading Corona Virus, tracking & identifying criminals and anti-spoofing etc.

By using a Deep Convolutional Neural Network Algorithm, we can easily detect the facial mask.

But the facial mask detection and non-masked face detection accuracy provided high variations.