



Face Mask Detection

Deena Nath

EasyChair preprints are intended for rapid dissemination of research results and are integrated with the rest of EasyChair.

July 1, 2021

Face Mask Detection

Deena Nath

1st July 2021

1 Abstract

COVID-19 pandemic is proving to be one of the biggest challenges faced by the world. Different safety measures are being taken by the governments such as lockdowns and mandatory application of face mask. Wearing a face mask is one of most efficacious methods of prevention according to the World Health Organisation (WHO). In this paper, a convolutional neural network model has been proposed which identifies whether a person is wearing a face mask or not. The model uses the robust TensorFlow library to work constructively. The model has been trained on an image dataset consisting of 3835 images where 1916 images are with face masks and 1919 images are of people without face mask. The images in the dataset has been collected from Bing Search API, Kaggle Datasets and RMFD Datasets. The proposed Deep Learning model gave an accuracy of 99 when trained on TensorFlow cpu 2.3.0.

2 Introduction

In this paper, we'll discuss our two-phase COVID-19 face mask detector, detailing how our computer vision/deep learning pipeline will be implemented. From there, we'll review the dataset we'll be using to train our custom face mask detector. I'll then show you how to implement a Python script to train a face mask detector on our dataset using Keras and TensorFlow. We'll use this Python script to train a face mask detector and review the results. Given the trained COVID-19 face mask detector, we'll proceed to implement two more additional Python scripts used to:

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

3 Deployment

In order to train a custom face mask detector, we need to break our project into two distinct phases, each with its own respective sub-steps (as shown by Figure 1 below):

Two-phase COVID-19 face mask detector

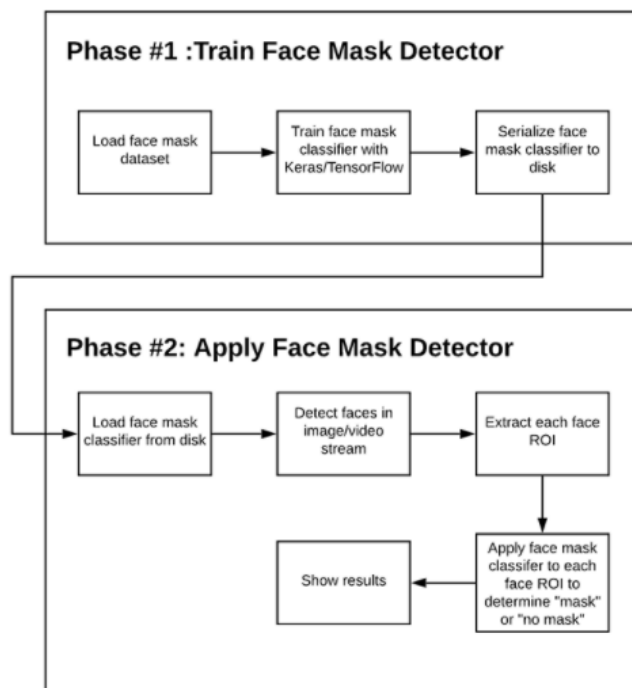


Figure 1: Phases and individual steps for building a COVID-19 face mask detector with computer vision and deep learning using Python, OpenCV, and TensorFlow/Keras.

Figure 1:

Training: Here we'll focus on loading our face mask detection dataset from disk, training a model (using Keras/TensorFlow) on this dataset, and then serializing the face mask detector to disk. Deployment: Once the face mask detector is trained, we can then move on to loading the mask detector, performing face detection, and then classifying each face as with mask or without mask. We'll review each of these phases and associated subsets in detail in the remainder of this tutorial, but in the meantime, let's take a look at the dataset we'll be using to train our COVID-19 face mask detector.

4 Conclusion

In this we learned how to create a COVID-19 face mask detector using OpenCV, Keras/TensorFlow, and Deep Learning. To create our face mask detector, we trained a two-class model of people wearing masks and people not wearing masks. We fine-tuned MobileNetV2 on our mask/no mask dataset and obtained a classifier that is 99% accurate. We then took this face mask classifier and applied it to both images and real-time video streams by:

Detecting faces in images/video
Extracting each individual face
Applying our face mask classifier

Our face mask detector is accurate, and since we used the MobileNetV2 architecture, it's also computationally efficient, making it easier to deploy the model to embedded systems (Raspberry Pi, Google Coral, Jetson, Nano, etc.)

References

<https://www.pyimagesearch.com/2020/05/04/covid-19-face-mask-detector-with-opencv-keras-tensorflow-and-deep-learning/>