Faculty of Computers & Artificial Intelligence

Benha University

Real-Time Face Mask Detector

By

Abdallah Tamer Mohamed El Ghamry

Section: 3 CS

June 2021

Table of Contents

1. Overview

1.1 Description and problem statement

At the end of 2019, COVID-19 (Coronavirus) was identified for the first time in Wuhan, China. During the past months, the world suffered from its impact. Many things in our life have been changed because of it. No treatment has been discovered officially until now so, many countries decided to live with the virus. It is hard to live with the virus so, people need to take some precautions to reduce chances of infection of the virus.

The World Health Organization (WHO) offers some instruction that help people to take caution. One of these instructions is wearing masks. In this paper, we will represent a real-time face mask detector using computer vision algorithms and deep learning approach. We aim to build a Convolution Neural Network (CNN) to ensure your safety and the safety of others. The output of the network should be a label which indicating whether the person wears a mask or not. This is a supervised binary classification task.

1.2 Scope

Project Includes
Detecting faces in images/video.
Face mask detection and classification.
Webcam video stream.

Project Excludes
Safe distance.
Detecting if the mask is a medical one or not.

1.3 Phases and individual steps for building the face mask detector

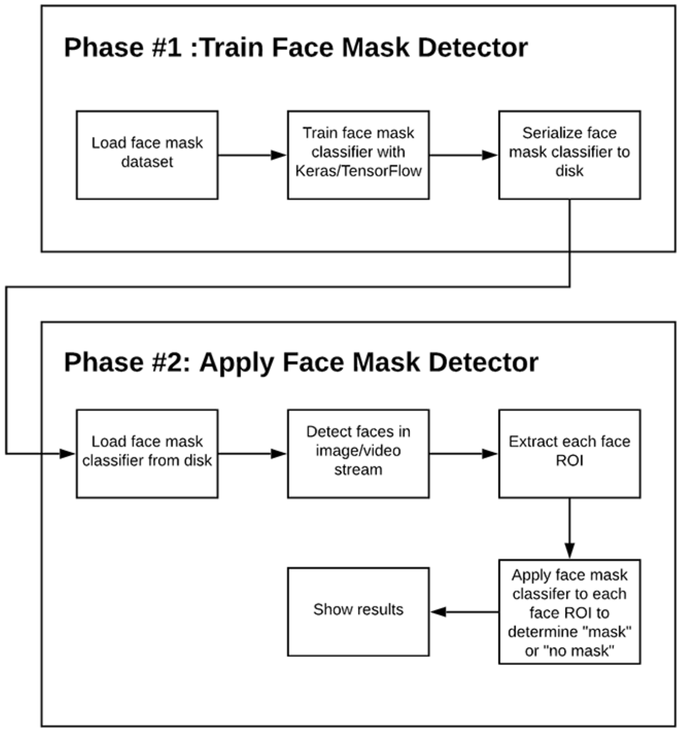

Figure 1 Phases and individual steps for building the face mask detector

2. Discussion

2.1 Data Preprocessing

This dataset is used for face mask classification with images. The dataset consists of approximately 11.8K RGB images belonging to two classes. All the images with the face mask (approximately 6K) are scrapped from google search and all the images without the face mask are preprocessed from the CelebFace dataset created by Jessica Li. The dataset is available on Kaggle.

https://www.kaggle.com/ashishjangra27/face-mask-12k-images-dataset

The dataset includes different types of masks to train the model. We trained the model over a balanced dataset.

Table 1 Training set basic information.

Label	Number of Images
With Mask	5000 images
Without Mask	5000 images

Table 2 Validation set basic information.

Label	Number of Images
With Mask	400 images
Without Mask	400 images

Table 3 Testing set basic information.

Label	Number of Images
With Mask	509 images
Without Mask	483 images

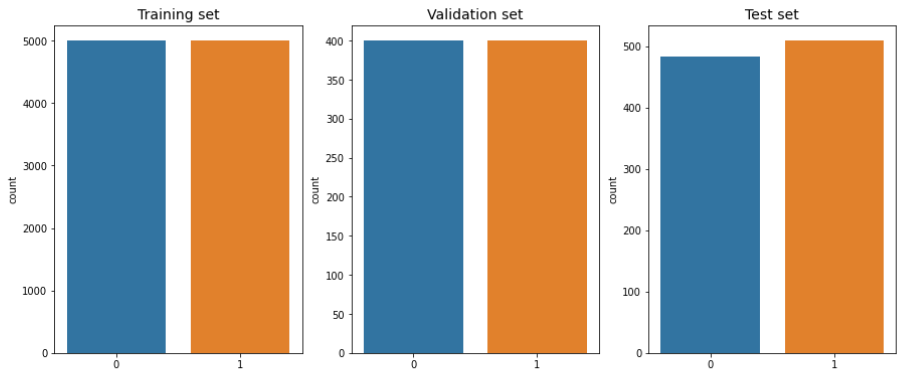

Figure 2 Dataset frequency

Figure 3 Sample of the dataset images

Data preprocessing includes the following steps:

1. Reading the images.
2. Resizing the images (64 x 64 x 3).
3. Normalizing the pixels (Each pixel has a value from 0 to 1).

This can be done using Keras Image Data Generator class.

2.2 Model Building and Training

Convolutional layer has a number of filters that does convolutional operation (feature extraction).

Pooling layer simplifies the output by performing nonlinear down sampling, reducing the number of parameters that the network needs to learn about.

The following figure shows an example of a small architecture for a typical CNN. One can see that the first layers are the convolutional ones which serve the role of generating useful features for classification. Those layers can be thought of as implementing image filters.

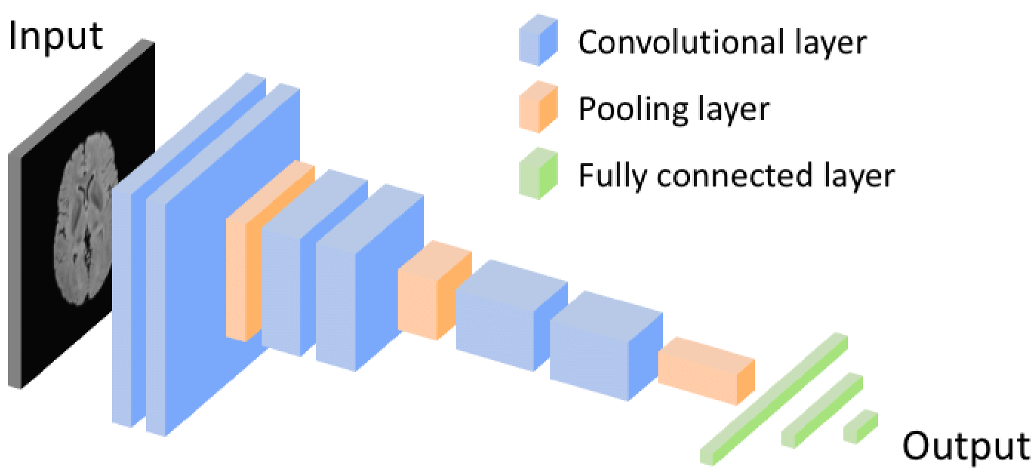

Figure 4 CNN architecture

The input of the network is the image and the output of the network is a label which indicating if the person wears a mask or not.

We used the following architecture to train the model.

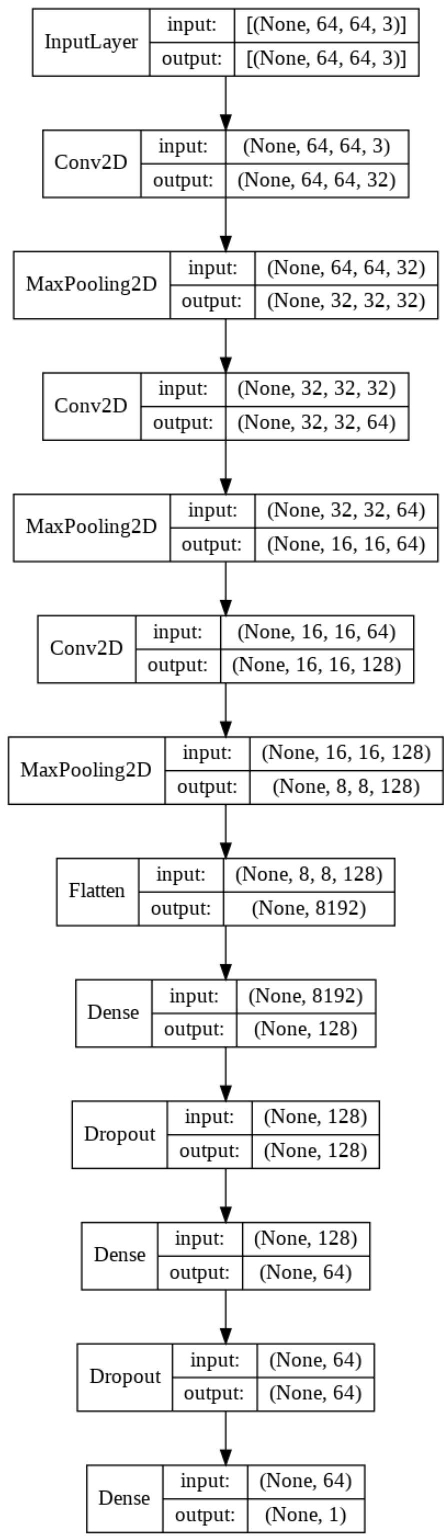

Figure 5 CNN used model architecture

2.3 Model Evaluation

The following curves illustrates the accuracy and the loss of the model.

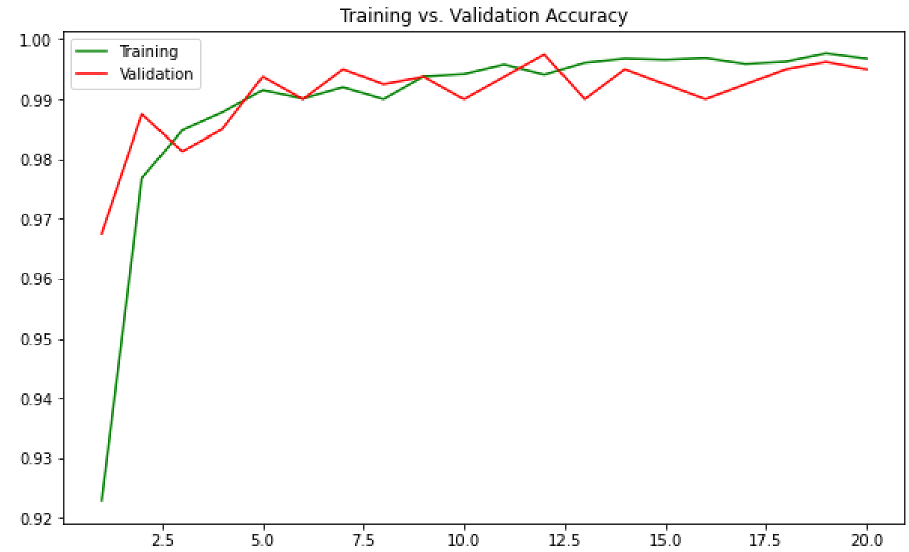

Figure 6 Training vs validation accuracy

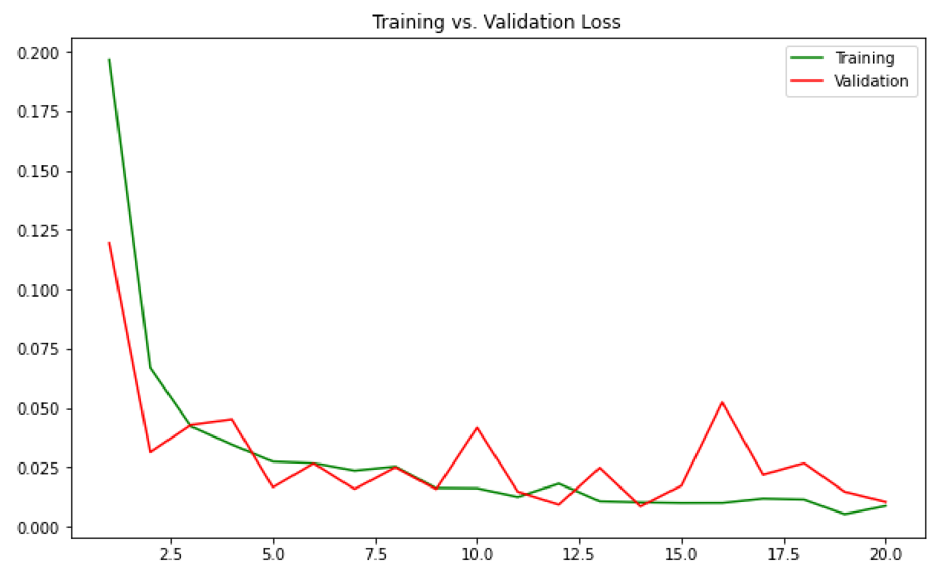

Figure 7 Training vs validation loss

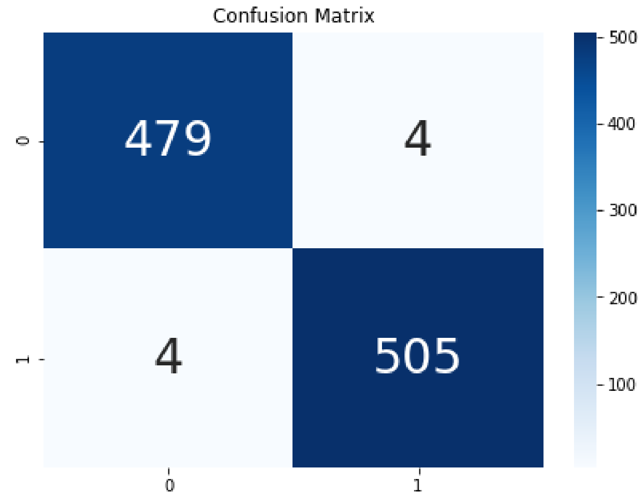

Figure 8 Confusion matrix

Because the dataset is approximately balanced, we should consider the accuracy as the main metric to evaluate the performance of the model. For the confusion matrix we can obtain the following metrics:

Metric	Value (%)
Accuracy	99.19
Precision	99.21
Recall	99.21
F1	99.21

2.4 Deployment

The deployment of the model involves the following steps:

1. Loading the trained CNN model.
2. Loading the Haar Cascade font face detector.
3. Starting video capturing using a camera.
4. Using the Haar Cascade to detect the faces in the frame.
5. Extracting each ROI (face).
6. Predicting the class of the face using the CNN model.
7. Drawing a box around each face to identify whether the person is wearing a mask or not.

2.4.1 Face Detection with Haar Cascade

Face detection is a common thing in computer vision tasks. We need to locate each face in the image to send the face to the CNN model.

Haar Cascade is a rapid object detection algorithm used to identify faces in an image or a real time video. The algorithm uses edge or line detection features to detect the face. We will use this algorithm to detect the front face of each face in the image.

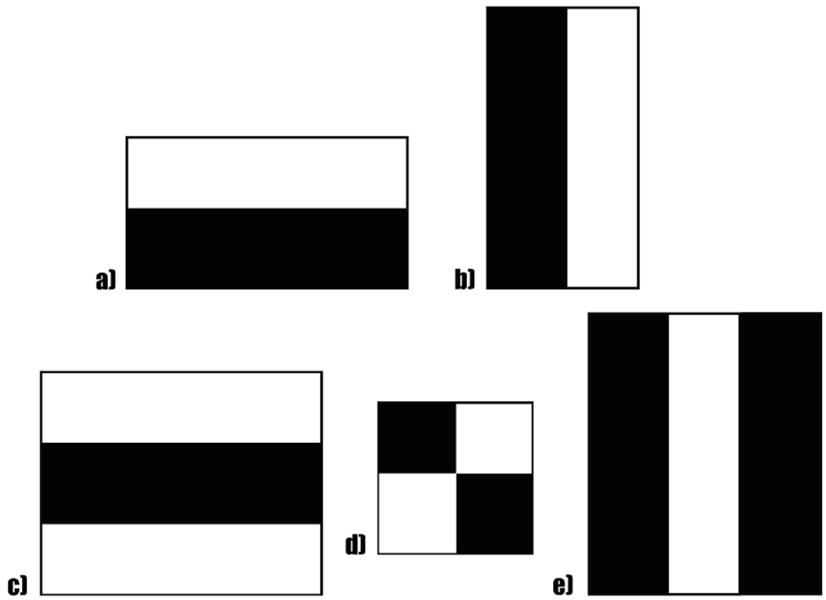

Figure 9 A sample of Haar features used in the original research paper.

2.4.2 Libraries

We used some Python libraries for data preprocessing, data visualization, deep learning, and computer vision tasks.

Table 4 Used libraries and their purposes.

Library	Usage
Numpy	Scientific computing
Pandas	Data analysis
Matplotlib & Seaborn	Data visualization
Tensorflow & Keras	Neural Networks (NN) and Deep Learning
Sklearn	Machine learning and evaluation metrics
OpenCV	Computer vision

2.4.3 Model operation

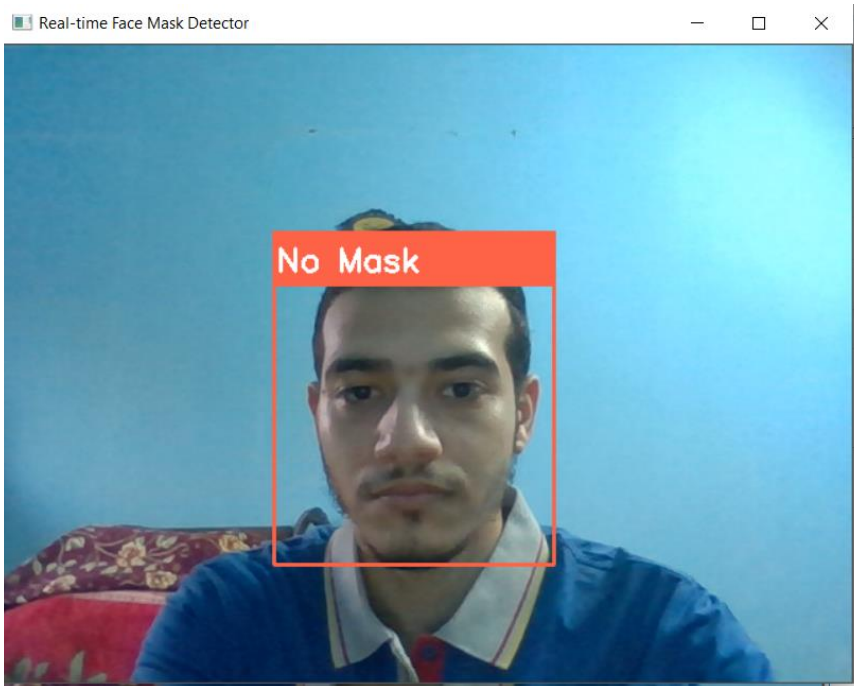

Figure 10 the output of the program without mask.

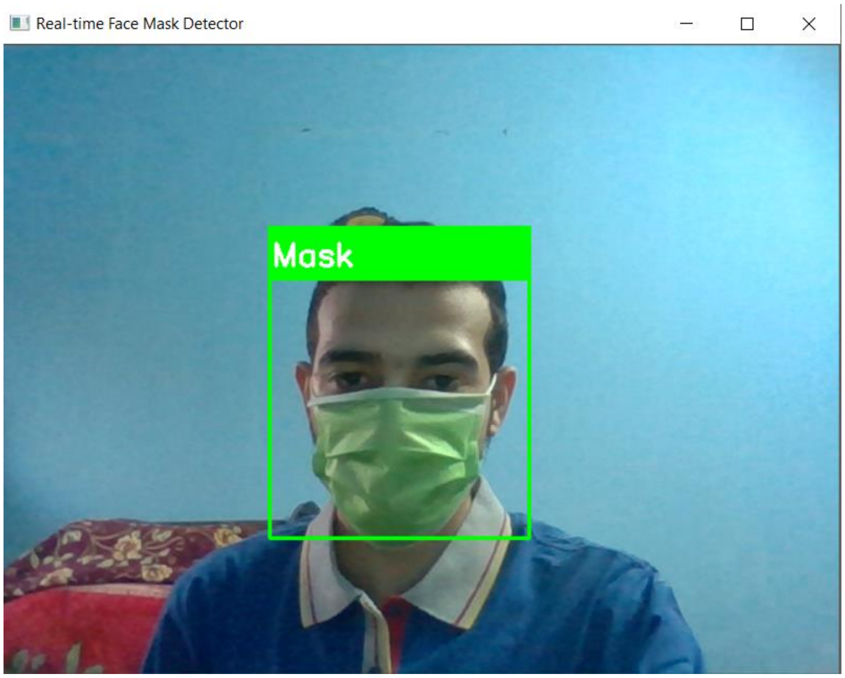

Figure 11 the output of the program with mask.

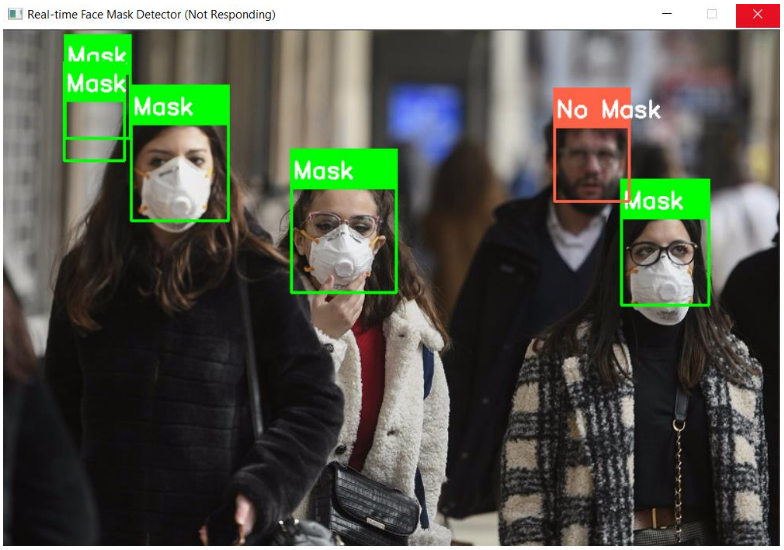

Figure 12the output of the program with many persons.

Figure 13 the output of the program with many persons.

3. Conclusion

In this paper, we introduced a real-time face mask detector using CNN and computer vision algorithms. We achieved 99.19% accuracy on the test set. This model can help to ensure the safety in schools, malls, offices, etc… Our face mask detector is accurate and efficient.

References

[1] Face Mask Detection ~12K Images Dataset
https://www.kaggle.com/ashishjangra27/face-mask-12k-images-dataset

[2] Face Detection with Haar Cascade
https://towardsdatascience.com/face-detection-with-haar-cascade-727f68dafd08

[3] Understanding CNN (Convolutional Neural Network)
https://towardsdatascience.com/understanding-cnn-convolutional-neural-network-69fd626ee7d4