IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Vol. 10, Issue 6, June 2021

DOI 10.17148/IJARCCE.2021.10614

Facemask Detection using OpenCv

Shruti Gupta, Vaibhav Dhok, Amol Chandrayan, Sonal Tiwari

Department of Computer Science and Engineering, Rajiv Gandhi College of Engineering Research, Nagpur

Abstract: COVID-19 pandemic has tremendously affected our day-to-day life affecting the world trade and movements. Wearing a protective face mask has become mandatory. In the near future, many public service providers will ask the customers to wear masks to avail of their services. Therefore, face mask detection has become a essential task to help global society. This paper presents a simplified approach to achieve this purpose using some basic deep Learning packages like TensorFlow, Keras, OpenCV. The proposed methodology detects the face from the image/video stream correctly and then identifies if it has a mask on it or not. As a surveillance task performer, it can also detect a face along with a mask in motion. The method obtains accuracy up to 95.55% and 94.23% respectively on two different datasets. We explore optimized values of parameters using the Convolutional Neural Network model to detect the presence of masks correctly without causing over-fitting.

Keywords: Convolutional Neural detection, TensorFlow, Deep Learning, Keras.

I. INTRODUCTION

Coronavirus disease (COVID-19) is an infectious disease caused by a newly discovered coronavirus.Most people infected with the COVID-19 virus will experience mild to moderate respiratory illness and recover without requiring special treatment. Older people, and those with underlying medical problems like cardiovascular disease, diabetes, chronic respiratory disease, and cancer are more likely to develop serious illness.The best way to prevent and slow down transmission is to be well informed about the COVID-19 virus, the disease it causes and how it spreads. Protect yourself and others from infection by washing your hands or using an alcohol based rub frequently and not touching your face.

Face Mask detection has turned up to be an astonishing problem in the domain of image processing and computer vision. Face detection has various use cases ranging from face recognition to capturing facial motions, where the latter calls for the face to be revealed withvery high precision. Due to the rapid advancement in the domain ofmachine learning algorithms, the jeopardies of face mask detection technology seem to be well addressed yet. This technology is more relevant today because it is used to detect faces not only in static images and videos but also in real-time inspection and supervision. With the advancements of convolution neural networks (Lawrence, Giles, Tsoi, & Back, 1997) and deep learning, very high accuracy in image classification and object detection can be achieved.Probably because of the sudden emergence of the COVID-19 pandemic, at present, there are various facial recognition technology applied to people wearing masks. HanvonTechnology Wang et al. (2020) reported that the accuracy of masked face recognition is about 85 %. An accuracy of over 90 % was obtained from Minivision Technology. The face-eye-based multi-granularity model (Wang et al., 2020) achieves 95% recognition accuracy.

II. TECHNIQUES FOR DETECTING FACE MASK:

- PHASE 1: Training face mask detector

I.Load face mask dataset.
II.Train face mask classifier with keras and tensorflow
III.Serialize face mask detector to disk

- PHASE 2: Apply face mask detector

IV. Load face mask classifier from disk
V. Detect faces in the video streams
VI.Extract each face ROI
VII.Apply face mask classifier to each face ROI to determine “mask” or “no mask”

This work is licensed under a Creative Commons Attribution 4.0 International License

IJARCCE

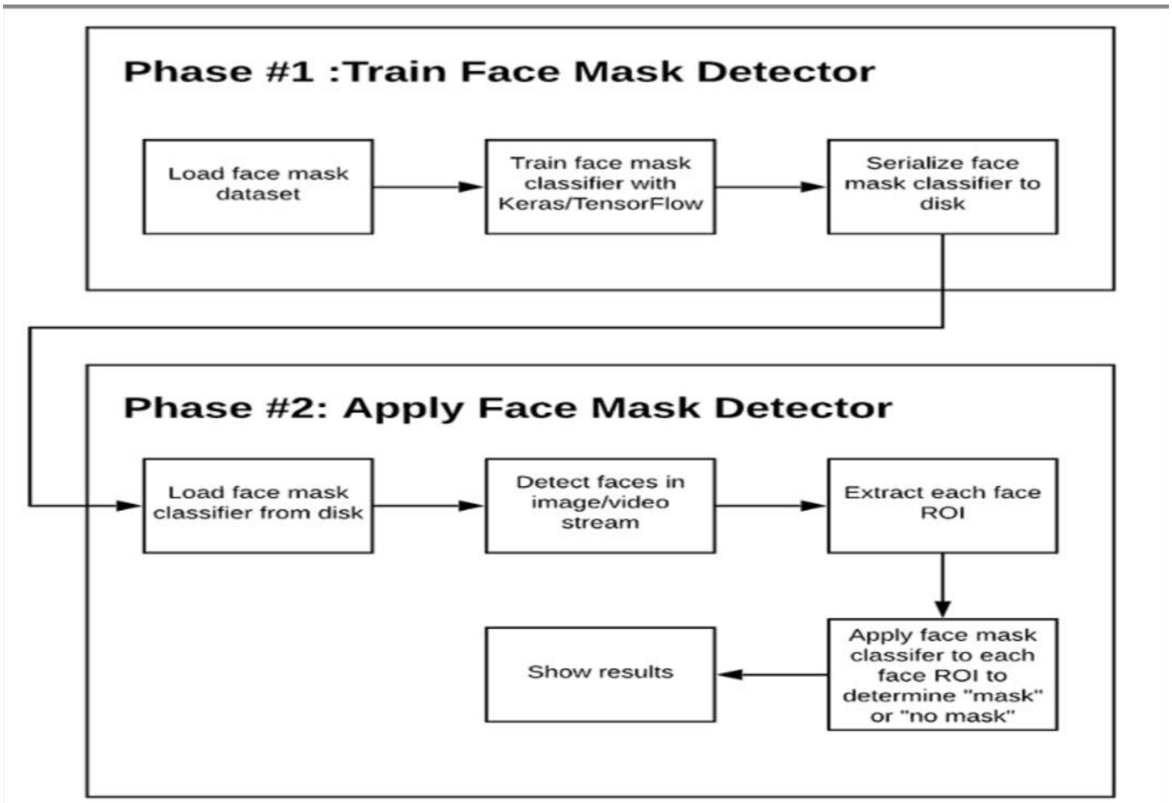
ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Vol. 10, Issue 6, June 2021

DOI 10.17148/IJARCCE.2021.10614

III. FLOW CHART OF THE PROJECT:


Phase #1 :Train Face Mask Detector
Load face mask dataset
Train face mask classifier with Keras/TensorFlow
Serialize face mask classifier to disk
Phase #2: Apply Face Mask Detector
Load face mask classifier from disk
Detect faces in image/video stream
Extract each face ROI
Apply face mask classifer to each face ROI to determine "mask" or "no mask"
Show results

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:

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
2. **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.

Our set of

tensorflow.keras

imports allow for:

- Data augmentation
- Loading the MobilNetV2 classifier (we will fine-tune this model with pre-trained IMAGENET weights)
- Building a new fully-connected (FC) head
- Pre-processing
- Loading image data

construct the argument parser and parse the arguments

```
ap = argparse.ArgumentParser()
ap.add_argument("-d", "--dataset", required=True,
help="path to input dataset")
ap.add_argument("-p", "--plot", type=str, default="plot.png",
help="path to output loss/accuracy plot")
ap.add_argument("-m", "--model", type=str,
default="mask_detector.model",
help="path to output face mask detector model")
args = vars(ap.parse_args())
```

This work is licensed under a Creative Commons Attribution 4.0 International License

Command line arguments include:

- --dataset

: The path to the input dataset of faces and and faces with masks

- --plot

: The path to your output training history plot, which will be generated using
matplotlib

- MODEL: The path to the resulting serialized face mask classification model

III. REQUIREMENTS OF THE PROJECT

Imports:
Import all the libraries and modules required.
2. Build the neural network:
This convolution network consists of two pairs of Conv and MaxPool layers to extract features from the dataset. Which is then followed by a Flatten and Dropout layer to convert the data in 1D and ensure overfitting.
And then two Dense layers for classification.
3. Image Data Generation/Augmentation:
4. Initialize a callback checkpoint to keep saving best model after each epoch while training:
5. Train the model:

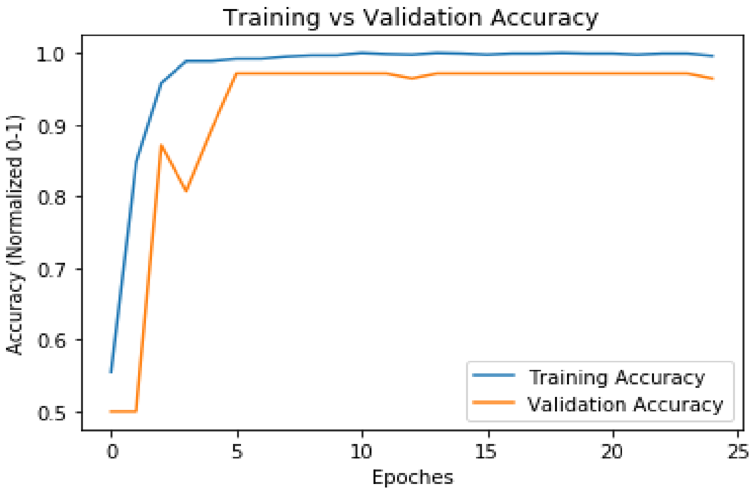
Data Preprocessing
Preprocessing steps as mentioned below was applied to all the raw input images to convert them into clean versions, which could be fed to a neural network machine learning model.
1.Resizing the input image (256 x 256)

2.Applying the color filtering (RGB) over the channels (Our model MobileNetV2 supports 2D 3 channel image)

3.Scaling / Normalizing images using the standard mean of PyTorch build in weights
4.Center cropping the image with the pixel value of 224x224x3

5.Finally Converting them into tensors (Similar to NumPy array)
Deep Learning Frameworks
To implement this deep learning network we have the following options.
1.TensorFlow

2.Keras
Accuracy Overview
The data set has been divided into two sets, likely a training and validation set. The accuracy of image classifier over the training set vs validation.



This work is licensed under a Creative Commons Attribution 4.0 International License

DOI 10.17148/IJARCCE.2021.10614

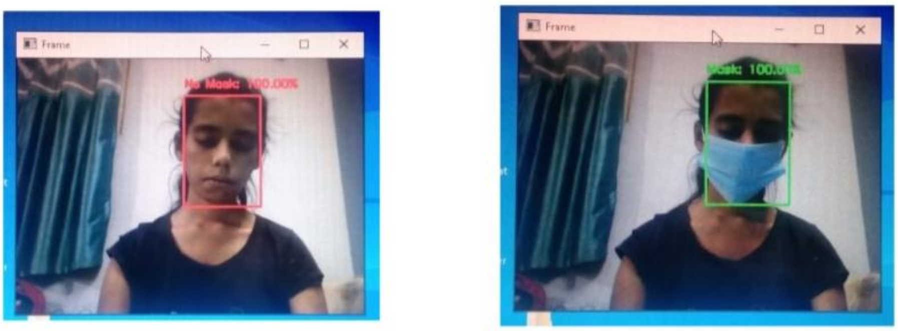
IV. EXPERIMENTAL SET-UP

In this project, we have developed a deep learning model for face mask detection using Python, Keras, and OpenCV. We developed the face mask detector model for detecting whether person is wearing a mask or not. We have trained the model using Keras with network architecture. Training the model is the first part of this project and testing using webcam using OpenCV is the second part.This is a nice project for beginners to implement their learnings and gain expertise.

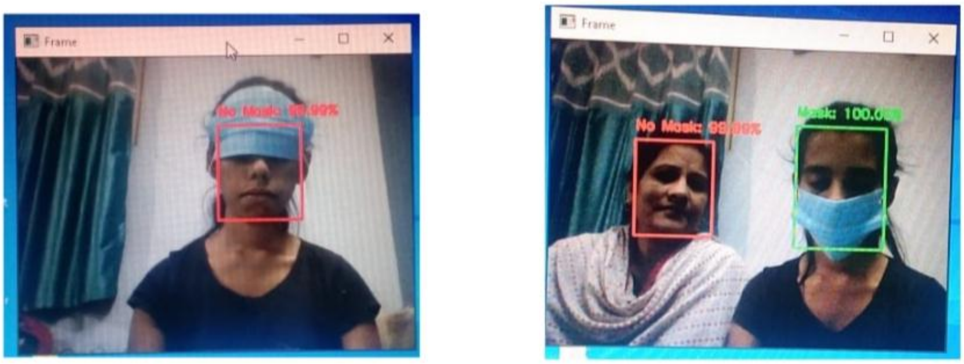
V. CONCLUSION

The current study used OpenCV and CNN to detect whether people were wearing face masks or not. The models were tested with images and real-time video streams. The accuracy of the model is around 99%, the optimization of the model is a continuous process and we are building a highly accurate solution by tuning the hyperparameters. We have successfully trained the module with the help of keras and tensorflow.

Output :



Output :



REFERENCES

[1]. P. Voila and M. Jones, “Rapid object detection using a boosted cascade of simple features,” in Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition. CVPR 2001, vol. 1. IEEE, 2001, pp. I-I

[2]. R. Girshick, “Fast r-cnn,” in Proceedings of the IEEE international conference on computer vision, 2015, pp. 1440-1448.

[3]. S. Ren,K. He, R. Girshick, and J. Sun, “Faster r-cnn: Towards real-time object detection with region proposal networks,” in Advances in nueral information processing systems, 2015, pp. 91-99

[4]. T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, “Focus loss for dense object detection.” 2017

[5]. Sammy v. militante, Nanettev. dionisioReal time face mask recognition with alarm system using deep learning 2020 11th IEEE control and system graduate research colloquium

[6]. Mohammad marufurmd. Motalebhossenmd. Milonislam Saifuddinmahmud An automated system to limit covid 19 using facial mask detection in smart city network 2020 IEEE international IOT

[7]. Toshanlalmeenpal, Ashutoshbalakrishnan, Amitverma, Face mask detection using semantic segmentation 2019,4th international conference on 4th International computing, communications and security(ICCCS)

[8]. Wenyunsun, Yusong, Changsheng, Face spoofing detection based on local ternary label supervision in fully convolutional networks IEE transactions on information forensis and security 2020

This work is licensed under a Creative Commons Attribution 4.0 International License