*International Journal For Technological Research In Engineering*
*Volume 9, Issue 10, June-2022* *ISSN (Online): 2347 - 4718*

# COVID-19: FACE MASK DETECTOR WITH OPENCV, KERAS/TENSORFLOW, AND DEEP LEARNING

[1]Kshitiz Badola, [2]Abhishek Soni, [3]Prateek Bansal, [4]Gurpreet Kaur
[1,2,3]Student, [4]Assistant Professor
[1,2,3]Department of Computer Science Engineering
Mahavir Swami Institute of Technology, Sonipat, India

***Abstract - COVID-19 pandemic has rapidly affected our day-to-day life disrupting world trade and movements. Wearing a protective face mask has become a new normal. In the near future, many public service providers will ask the customers to wear masks correctly to avail of their services. Therefore, face mask detection has become a crucial task to help global society. This paper presents a simplified approach to achieve this purpose using some basic Machine Learning packages like TensorFlow, Keras, OpenCV and Scikit-Learn. The proposed method detects the face from the image 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 attains accuracy up to 95.77% and 94.58% respectively on two different datasets. We explore optimized values of parameters using the Sequential Convolutional Neural Network model to detect the presence of masks correctly without causing over-fitting.***
.

## 1. INTRODUCTION

According to the World Health Organization (WHO)'s official Situation Report – 205, coronavirus disease 2019 (COVID-19) has globally infected over 20 million people causing over 0.7million deaths. Individuals with COVID-19 have had a wide scope of symptoms reported – going from mellow manifestations to serious illness. Respiratory problems like shortness of breath or difficulty in breathing is one of them. Elder people having lung disease can possess serious complications from COVID-19 illness as they appear to be at higher risk. Some common human coronaviruses that infect the public around the world are 229E, HKU1, OC43, and NL63. Before debilitating individuals, viruses like 2019-nCoV, SARS-CoV, and MERS-CoV infect animals and evolve to human coronaviruses. Person having respiratory problems can expose anyone (who is in close contact with them) to infective beads. Surroundings of a tainted individual can cause contact transmission as droplets carrying virus may arrive on his adjacent surfaces. To curb certain respiratory viral ailments, including COVID-19, wearing a clinical mask is very necessary. The public should be aware of whether to put on the mask for source control or aversion of COVID-19. Potential points of interest of the utilization of masks lie in reducing vulnerability of risk from a noxious individual during the "pre-symptomatic" period and stigmatization of discrete persons putting on masks to restrain the spread of virus. WHO stresses on prioritizing medical masks and respirators for health care assistants. Therefore, face mask detection has become a crucial task in present global society. 7 Face mask detection involves detecting the location of the face and then determining whether it has a mask on it or not.

The issue is approximately cognate to general object detection to detect the classes of objects. Face identification categorically deals with distinguishing a specific group of entities i.e. Face. It has numerous applications, such as autonomous driving, education, surveillance, and so on. This project presents a simplified approach to serve the above purpose using the basic Machine Learning (ML) packages such as TensorFlow, Keras, OpenCV and Scikit-Learn. In the face detection method, a face is detected from an image that has several attributes in it. According to, research into face detection requires expression recognition, face tracking, and pose estimation. Given a solitary image, the challenge is to identify the face from the picture. Face detection is a difficult errand because the faces change in size, shape, color, etc and they are not immutable. It becomes a laborious job for an opaque image impeded by some other thing not confronting the camera, and so forth. I also think occlusive face detection comes with two major challenges: 1) unavailability of sizably voluminous datasets containing both masked and unmasked faces, and 2) exclusion of facial expression in the covered area. Utilizing the locally linear embedding (LLE) algorithm and the dictionaries trained on an immensely colossal pool of masked faces, synthesized mundane faces, several mislaid expressions can be recuperated and the ascendancy of facial cues can be mitigated to great extent. According to the work report in convolutional neural networks (CNNs), computer vision comes with a strict constraint regarding the size of the input image. The prevalent practice reconfigures the images before fitting them into the network to surmount the inhibition. 8 Here the main challenge of the task is to detect the face from the image correctly and then identify if it has a mask on it or not. In order to perform surveillance tasks, the proposed method should also detect a face along with a mask in motion.

  

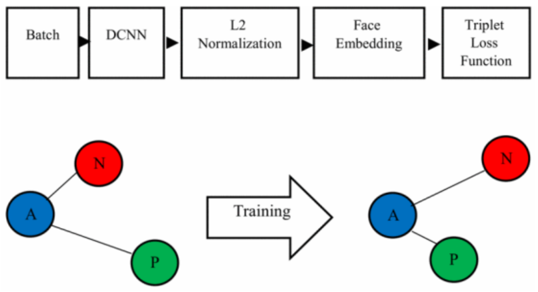


Fig. 1
FRAMEWORK USED IN THE PAPER

Fig. 2 illustrates the procedures and methodologies that were implemented in this study. This includes acquiring and preprocessing the data and the training and evaluation of the model. Fig. 1. Methods of Dimensionality Reduction

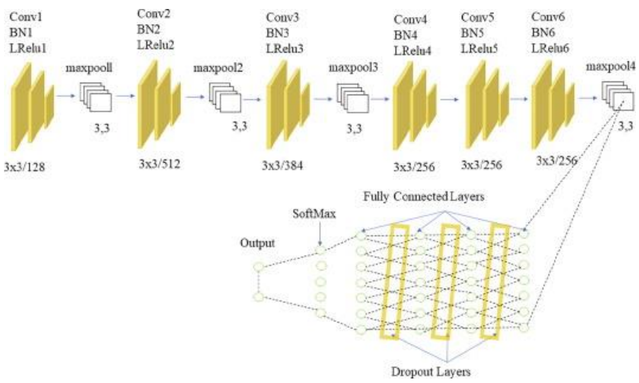


Fig 2

The authors used the following software tools to make for dataset preparation, model training, and evaluation: Google Colaboratory as the training and testing environment, Python 3 as the primary programming language; Tensorflow Object Detection API that is built on Tensorflow library, which is a framework that provides tools for quickly creating object detection models; Darknet, an open-source neural network framework, written in C, that is used in YOLOv4 framework, and a required dependency for both Tensorflow and Darknet framework, OpenCV (Open Source Computer Vision) library, an open-source computer vision library that is optimized for real-time computer vision. Other optional dependencies for training and inference include NVIDIA's CUDA and CuDNN for optimized deep learning calculations. A. Data Gathering The authors acquired datasets that are publicly available in the wild. The classes represented by the instances belong to the following labels: Wearing a Mask (WM), Not Wearing a Mask (NWM), and Improperly Wearing a Mask (IWM). To collect the instances for training, the authors manually collected publicly available images from search engines and consented public social media images of people. For the images taken from individuals, a consent form was signed by certain respondents. The instances in the WM class contain people wearing plain surgical or cloth masks. On the other hand, NWM class instances contain faces of people that are not wearing a mask. These images consisted of different conditions, including people looking at the camera, group meetings, and pedestrians. For the IWM instances, the authors gathered images containing different ways people improperly wear a mask. These include the mouth is covered, but the nose is visible, the mask is hanging on the ear, and the mask is only covering the neck. In order to meet a minimum number of 1,000 annotated instances per class, the authors gathered a total of 1,365 images from various sources. The gathered images vary in size, image lighting, angles where faces are looking, and image quality to best represent the real-life conditions of images taken on a day-to-day basis. The authors ensured that the collected images are mixed with different scales and how occluded the facial details were. The final number of instances that are annotated in WM class is 1,012 instances. The number of annotated instances for the NWM class is equivalent to 1,006 instances. For the IWM class, the final number of instances that are annotated is 1,007 instances. B. Data Preprocessing The authors applied image preprocessing techniques such as format conversion, metadata removal, rescaling, and cropping. This preprocessing was automatically done using a Python script programmed by the authors. Format conversion was applied to ensure format compatibility among the instances. Moreover, metadata removal was used to reduce the image size and to ensure disinformation. The resizing and cropping of instances were made to ensure the uniformity of the aspect ratio before training. After collecting unlabeled images, the authors performed annotations by manually drawing bounding boxes on the images using an open-source image annotation tool called LabelImg, where annotations files are exported in YOLO format (.txt) and later converted to PASCAL VOC (.xml) format. Data augmentation is a technique used to increase the size of the dataset by generating training instances by applying different image processing techniques. It is instrumental in applying face mask detectors because it can generalize the dataset in various conditions. To perform data augmentation, the authors used the augmentations Python library which has been created. This library supports augmenting datasets with bounding boxes because it also adjusts the bounding boxes whenever an augmentation technique that changes the position of objects is applied. After rescaling the images, the authors annotated the images using the LabelImg tool. The images were carefully and consistently annotated so that the models would generalize well. For the face mask detection model, the bounding boxes were precisely drawn in the area of the face, where the face is the object of interest, excluding the hair. All annotation files were saved in YOLO format for YOLOv4 training and later converted to PASCAL VOC format for RetinaNet and MobileNet-SSD frameworks. The authors split the dataset for training and testing. The dataset was split correctly into 80% dataset distribution for the training set and 20% instances for the test set. The splitting is based on the number of instances and not on the number of images. In addition, the splitting is balanced into an approximately 8:2 ratio per class. For example, in NWM class, images that contain 80% of instances were set as training data, and the remaining 20% were included for testing data. The authors chose to stick to the nearest possible value of the calculated values, which equates to 667 instances used in 300 images for the test set since there is no definite number of instances contained in the images used and the images used in the test set were randomly chosen. Data augmentation was applied to the training data, in exclusion of the test set. The transformations done to the images were: horizontal flip, random rotation, blur, and gaussian noise. First, the horizontal flip is applied to the original image, and both original and flipped images are augmented with combinations of random rotation, random blur, and random addition of noise. After augmentation, the number of images increased by 4-folds using the authors' techniques plus the factors added by the predefined augmentation techniques given in the frameworks. C. Face Mask Detection Model Training Object detection in computer vision locates instances of objects in an image or a video. Newer object detection frameworks utilize machine learning and deep learning networks to create more accurate predictions than before. Object detection works using two tasks: localization and classification, where localization tries to predict the object's location by segmenting the image or video, while classification identifies the object's correct label. For the model training, the authors used three different object detection frameworks. All three chosen frameworks utilize a one-stage technique that eliminates the process of region proposal, making the object detection results in faster predictions. MobileNet-SSD, RetinaNet, and YOLOv4 are state-of-the-art one-stage object detection frameworks that are all open-source frameworks. The SSD MobileNet V1 [6] implementation in Python was used to train the MobileNet model. The size of the training set used for this model was approximately 14,484 - 9,675 of which was the result of the authors' additional augmentation, and the remaining was from the random crop it does during training. The base learner, MobileNet V1, is a 28-layered deep learning architecture designed for real-time detections. For this architecture, the input size was set to 640x640. Fig. 2 illustrates the structure of the MobileNet framework used in this study.

## 2. ACCURACIES PRESENTED BY FRAMEWORK

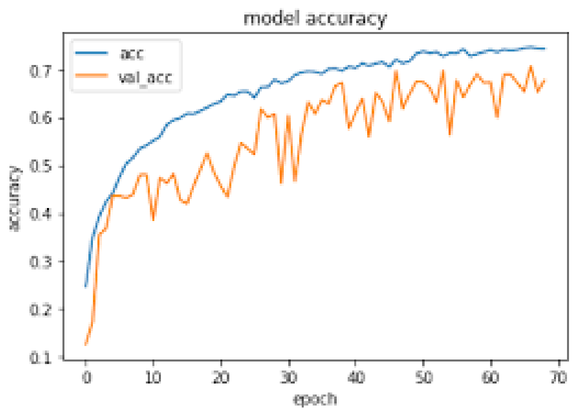


Fig. 3

The PR curve shown in Fig. 3 is consistent with the trends associated with the confusion matrices and other performance indicators. It can be noticed that the NWM class (red line) of the YOLOv4 classifier reached the highest precision values among the classes. This signifies the integrity of the experimental results conducted in this study that all models have consistent generalizations based on multiple different performance metrics.

## 3. RESULT

We have built the face recognition system for masked faces. Initially, our trained model showed almost 99% accuracy on the LFW dataset, confirming that the model is accurately trained on the VGGFACE2 dataset. Later, this trained model was deployed for the masked faces recognition pipeline. The masked dataset contained 800 images of 200 individuals. These 800 images comprise of 200 unmasked and 600 masked images (frontal and size pose for classifier building purposes). Once the system is trained with unmasked classifiers only and tested with masked faces, its accuracy was 79%. However, when the side poses masked faces are included with unmasked face classifiers, later training is carried out with the same classifier. System performance increased remarkably and accuracy improved up to 98% on the local dataset.

## 4. CONCLUSION AND FUTURE WORK

In this work, a deep learning-based approach for detecting masks over faces in public places to curtail the community spread of Coronavirus is presented. The proposed technique efficiently handles occlusions in dense situations by making use of an ensemble of single and two-stage detectors at the pre-processing level. The ensemble approach not only helps in achieving high accuracy but also improves detection speed considerably. Furthermore, the application of transfer learning on pre-trained models with extensive experimentation over an unbiased dataset resulted in a highly robust and low-cost system. The identity detection of faces, violating the mask norms further, increases the utility of the system for public benefits. Finally, the work opens interesting future directions for researchers. Firstly, the proposed technique can be integrated into any high-resolution video surveillance devices and not limited to mask detection only. Secondly, the model can be extended to detect facial landmarks with a facemask for biometric purposes.

## REFERENCES

1. M. Loey, G. Manogaran, M. H. N. Taha, and N. E. M. Khalifa, "A hybrid deep transfer learning model with machine learning methods for face mask detection in the era of the COVID-19 pandemic," Measurement, vol. 167, Article ID 108288, 2021.
2. B. Qin and D. Li, "Identifying facemask-wearing condition using image super-resolution with classification network to prevent COVID-19," Sensors, vol. 20, no. 18, p. 5236, 2020.
3. Z. Wang, P. Wang, P. C. Louis, L. E. Wheless, and Y. Huo, "Wearmask: fast In-browser face mask detection with serverless edge computing for COVID-19," 2021, https://arxiv.org/abs/2101.00784.
4. X. Zhang, H. Saleh, E. M. Younis, R. Sahal, and A. A. Ali, "Predicting coronavirus pandemic in real-time using machine learning and big data streaming

  

system," Complexity, vol. 2020, Article ID 6688912, 10 pages, 2020.

5. M. Razavi, H. Alikhani, V. Janfaza, B. Sadeghi, and E. Alikhani, "An automatic system to monitor the physical distance and face mask wearing of construction workers in COVID-19 pandemic," 2021, https://arxiv.org/abs/2101.01373.
6. S. K. Dey, A. Howlader, and C. Deb, "MobileNet mask: a multi-phase face mask detection model to prevent person-to-person transmission of SARS-CoV-2," in Proceedings of International Conference on Trends in Computational and Cognitive Engineering, pp. 603–613, Springer, Dhaka, Bangladesh, December 2021.

  