FACE MASK DETECTION SYSTEM Using Python and OpenCV

Divija Sisodia[1], Anukrati Rawal[2], Ketuman Vishwakarma[3], Bhavesh Rajani[4], Devashish Mandloi[5], Dr.Asif Ali[6]

[1,2,3,4,5]B.Tech Scholar, [6]Associate Professor, Department of Information Technology, Acropolis Institute of Technology & Research, Rajiv Gandhi Proudyogiki Vishwavidyalaya, Indore, India

Abstract— *Heretofore, face masks have been the first line of defense for protection since the outbreak of the Coronavirus Disease in 2019(COVID-19) around the world. Studies have demonstrated that wearing a face mask fundamentally diminishes the danger of viral transmission and gives a feeling of protection. As the world recuperates from the pandemic and plans to get back to a condition of routineness, there is a flood of tension among all people, particularly the individuals who mean to continue face-to-face activities in their daily lives. This project aims to identify the presence of face masks on live recordings or pictures of people. We'll utilize Python and OpenCV for this undertaking. We've worked on datasets accessible on the web from Kaggle to test and train our model. Moreover, we have utilized basic concepts of transfer learning and object detection algorithms in neural networks to finally detect the presence or absence of face masks in a picture or a video stream. The aftermath of the trial and test results shows that our model performs well on the test data with 99% on both precision and recalls.*

Keywords— Detection System, Coronavirus, COVID-19, Face Mask, Python, OpenCV, Keras.

1. INTRODUCTION

A new public health crisis threatens the world with the emergence and spread of the 2019 novel coronavirus (2019-nCoV) or the severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2). The virus originated in bats and was transmitted to humans through unknown intermediary animals in Wuhan, Hubei province, China, in December 2019. [1]. There have been 167,202,778 COVID-19 cases and 3,471,408 deaths around the world to date(Sunday, May 23, 2021). [2]. The disease is transmitted by inhalation or contact with infected droplets and the incubation period ranges from 2 to 14 d. The symptoms are usually fever, cough, sore throat, breathlessness, fatigue, and malaise. The disease is mild in most people; in some (usually the elderly and those with comorbidities), it may progress to pneumonia, acute respiratory distress syndrome (ARDS), and multi-organ dysfunction. Many people are asymptomatic. [3].

Numerous precautionary steps have been taken to battle against the Coronavirus. Amongst them, cleaning hands, keeping distance, wearing a mask, abstaining from contacting eyes, nose, and mouth are the primary, where wearing a mask is the easiest one. Coronavirus is a disease spread from one human to another, which can be constrained by guaranteeing the proper and correct way of wearing a mask. The spread of COVID-19 can be restricted if individuals rigorously stick to social distancing and wear a facial mask. Tragically, people are behaving poorly and not complying with these rules effectively, speeding the spread of the virus. Detecting individuals not adhering to the rules and informing the concerned authorities can diminish the spread of the Coronavirus. To screen that individuals follow this fundamental security standard of wearing a mask, a detection technique ought to be implemented. A Face mask detection system can be implemented for the same. Face mask detection implies distinguishing if an individual is wearing a mask or not. To screen that individuals are following this fundamental security

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 05 Issue: 06 | June - 2021 ISSN: 2582-3930

standard, a detection technique ought to be implemented. A Face mask detection system can be implemented for the same. Face mask detection implies distinguishing if an individual is wearing a mask or not. Face detection is one of the applications of object detection and can be used in many areas like security, biometrics, law enforcement, autonomous driving, education, surveillance, and more. This paper presents a simplified approach to serve the purpose of face mask detection using the basic Machine Learning (ML) packages/libraries such as TensorFlow, Keras, OpenCV, MobileNetV2, imutils, NumPy, Matplotlib, and Scipy.

The rest of the paper is organized as follows: Section 2 explores related work associated with face mask detection. Section 3 discusses the methodology, which includes the nature of the used dataset and the details of the packages incorporated to build the proposed model. Section 4 comprises experimental results and analysis. Section 5 concludes the paper and draws the line towards future works. Finally, Section 6 reports the acknowledgment, and Section 7 lists the references cited in this paper.

2. RELATED WORK

As cited in [4]. Images containing faces are essential to intelligent vision-based human-computer interaction, and research efforts in face processing include face recognition, face tracking, pose estimation, and expression recognition. However, many reported methods assume that the faces in an image or an image sequence have been identified and localized. To build fully automated systems that analyze the information contained in face images, robust and efficient face detection algorithms are required. Given a single image, the goal of face detection is to identify all image regions which contain a face, regardless of its 3D position, orientation, and lighting conditions. Such a problem is challenging because faces are non-rigid and have a high degree of variability in size, shape, color, and texture. Numerous techniques have been developed to detect faces in a single image. So two significant problems pose a challenge:

1) Unavailability of enormous datasets consisting of both masked and unmasked faces. And,

Existing System	Features	Benefits	Limitations
Manual mask checking	Personnels are appointed at public places to manually check if people are wearing masks or not.	It is pretty accurate. The personnels are easily able to check whether the rules are being followed or not.	Requires human resources, which is very costly and time-consuming. Checking on regular intervals at Traffic stops

2) Exclusion of facial expressions in the masked/hidden areas of the face.

Different scientists and experts chiefly centered around gray-scale face images. [5]. While some were based on pattern identification models, possessing initial information of the face model while others were using AdaBoost [6]., which was an incredible classifier for training purposes. At that point came the Viola-Jones Detector, which gave a forward leap in face recognition innovation, and continuous face identification got conceivable. It dealt with different issues like the orientation and brightness of the face; it failed to work in dull, dark, and dim environments. Furthermore, Zhang, Zhang, Li, and Qiao [7]., one of the renowned researchers, made a DPM-based face mask detector using around 30000 faces divided into masks and without masks category. His work achieved an exceptional accuracy of 97.14 %, and DPM-based face mask detection models can achieve majestic precisions. Still, the cost of computation due to the use of DPM was colossal and staggering, which isn't very feasible for local use.

The current scenario in India for detecting face masks is dependent on security personnel manually checking the masks.

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 05 Issue: 06 | June - 2021 ISSN: 2582-3930

3. METHODOLOGY

Block diagram:

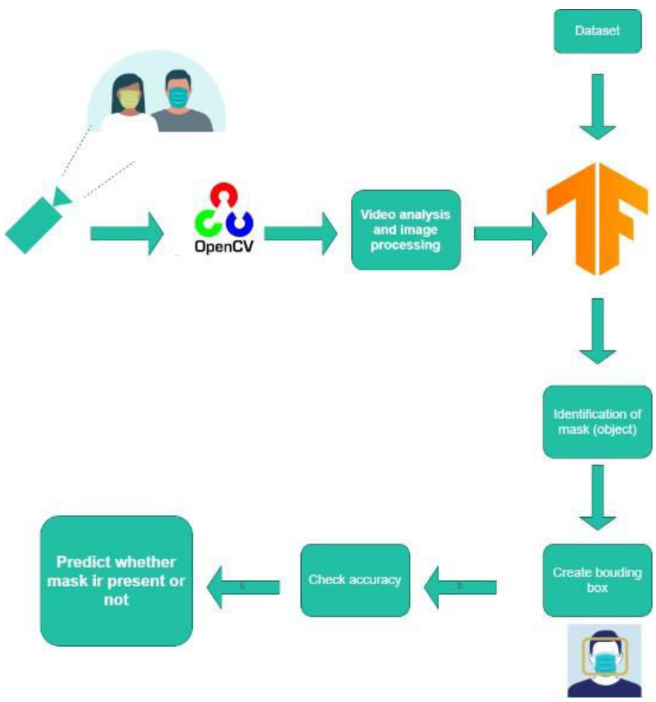

Figure 1. Block diagram of the proposed face mask detection system.

Data-Flow Diagram:

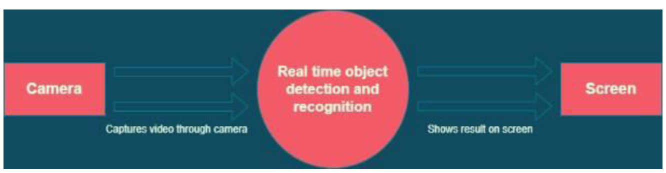

Figure 2. Data-Flow Diagram(DFD) Level 0.

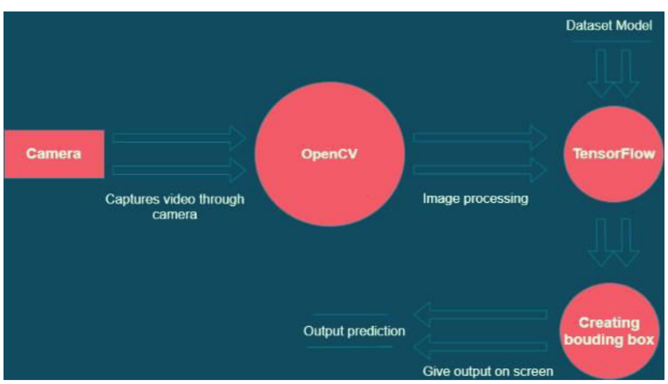

Figure 3. Data-Flow Diagram(DFD) Level 1.

Masked dataset:

Figure 4. Dataset consisting of masked faces.

Unmasked dataset:

Figure 5. Dataset consists of unmasked faces.

The dataset which we have utilized comprises 3833 total images, out of which 1915 are of masked faces, and 1918 are of unmasked faces. Every one of the pictures is real pictures separated from Kaggle. The pictures cover diverse races, i.e., Asian, Caucasian, and so forth. The extent of masked faces to unmasked faces confirms that the dataset is balanced. We need to divide our dataset into two sections: training dataset and testing dataset. The motivation behind dividing the dataset is to stay away from overfitting, focusing on minor details/noise that is unnecessary, and optimizes the training dataset accuracy. A model that performs well on a dataset that it has never seen (test information), which is called generalization is needed. The training set is the real subset of the dataset that we use to prepare the model. The model observes and learns from this data and afterward optimizes its boundaries. The test set is the remaining subset of data used to give an unbiased assessment of a final model fitted on the training dataset.

Packages incorporated

A. Keras

Keras is the high-level API of TensorFlow 2: an approachable, highly-productive interface for solving machine learning problems, focusing on modern deep learning. It provides essential abstractions and building blocks for developing and shipping machine learning solutions with high iteration velocity. Keras empowers engineers and researchers to take full advantage of the scalability and cross-platform capabilities of TensorFlow 2. [7].

B. TensorFlow

TensorFlow is an interface for expressing machine learning algorithms and an implementation for executing such algorithms. It has been used to c machine learning systems into production across more than a dozen areas of computer science and other fields, including speech recognition, computer vision, robotics, information retrieval, natural language processing, geographic information extraction, and computational drug discovery. [8].

C. OpenCV

OpenCV (Open Source Computer Vision Library), an open-source computer vision and ML software library, is utilized to differentiate and recognize faces, recognize objects, group movements in recordings, trace progressive modules, follow eye gesture, track camera actions, expel red eyes from pictures taken utilizing flash, find comparative pictures from an image database, perceive the landscape and set up markers to overlay it with increased reality and so forth. [9].

D. MobileNetV2

MobileNetV2 is a convolutional neural network architecture that seeks to perform well on mobile devices. It is based on an inverted residual structure where the residual connections are between the bottleneck layers. The intermediate expansion layer uses lightweight depthwise convolutions to filter features as a source of non-linearity. As a whole, the architecture of MobileNetV2 contains the initial fully convolution layer with 32 filters, followed by 19 residual bottleneck layers. [10].

4. RESULTS AND ANALYSIS

The dataset is partitioned into a training and testing set. The dataset comprises 3833 samples in total, where 80% is used in the training phase, and 20% is used in the testing phase. The training and testing dataset contains 3066 and 767 images, respectively. The developed architecture is trained for 4000 epochs since different training results cause overfitting on the training data. Overfitting occurs when a model learns the unwanted patterns of the training data. Hence, training accuracy increases, but testing accuracy decreases substantially.

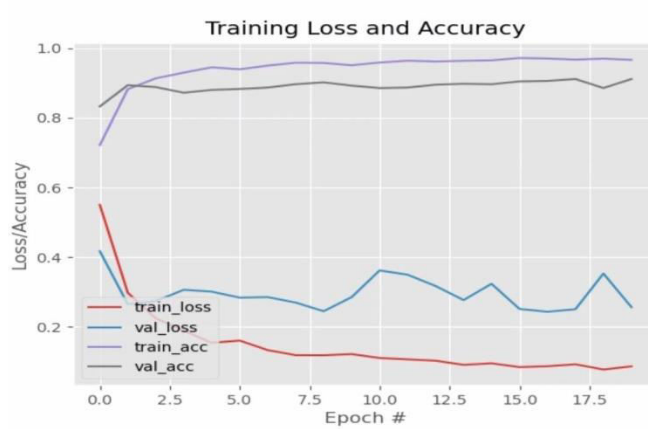

Figure 5. Training loss and accuracy graph for epochs vs. loss.

	precision	recall	f1-score	support
with_mask	0.99	0.99	0.99	383
without_mask	0.99	0.99	0.99	384
accuracy			0.99	767
macro avg	0.99	0.99	0.99	767
weighted avg	0.99	0.99	0.99	767

Figure 6. Model evaluation classification report.

4.1 Visualization of the results

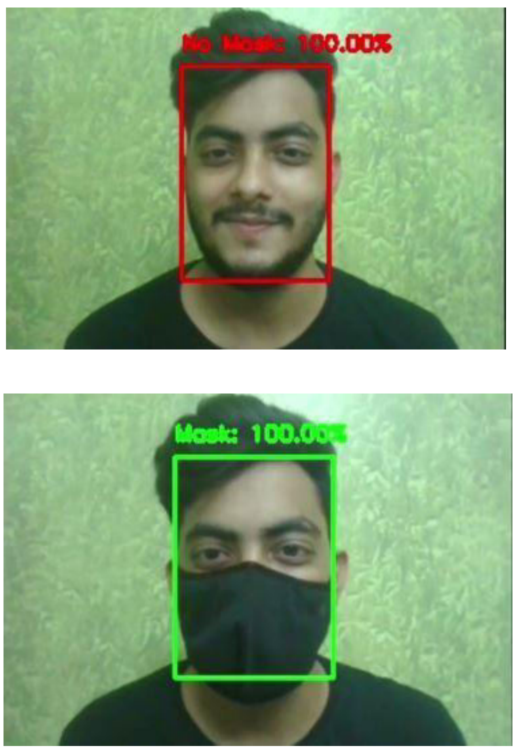

Figure 6. Prediction on live video.

The rectangular green box depicts the correct way of wearing a mask with an accuracy score on the bottom, while the red rectangular box represents the incorrect way of wearing a mask on the top. The model learns from the pattern of the training dataset and labels the dataset into two categories, "with_mask" and "without_mask," and then makes predictions.

5. CONCLUSION AND FUTURE SCOPE

In this paper, we momentarily clarified the driving force of the work and the problem of manual mask checking we wished to resolve. Furthermore, we represented the learning, implementation, and evaluation of the model. Utilizing basic Machine Learning algorithms and techniques, the model has achieved reasonably high accuracy and precision. Wearing a mask might be compulsory sooner rather than later, considering the Covid-19 situation around the world. The proposed model will contribute tremendously to the public health care system. In the future, it tends to be reached out to distinguish if an individual is wearing the mask correctly or not. The model can be additionally improved to recognize if the mask is prone to infection or not by detecting the type of mask. For example, the sort of mask is safe against the virus like N95 or not.

6. ACKNOWLEDGMENT

We would like to acknowledge Prof. Asif Ali for his constant support and guidance throughout our project, making it a seamless fusion of applied knowledge and diligence.
Also, we would like to thank the Information Technology Department for giving us this massive opportunity to contribute our bit towards the tech community.

7. REFERENCES

[1] "A Review of Coronavirus Disease-2019......" 13 Mar. 2020, https://link.springer.com/article/10.1007/s12098-020-03263-6.

[2] "COVID Live Update: 167,362,130 Cases and 3,475,053 Deaths from.................." https://www.worldometers.info/coronavirus/.

[3] "A Review of Coronavirus Disease-2019 (COVID-19) - NCBI - NIH." 13 Mar. 2020, https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7090728/.

[4] "Detecting faces in images: a survey | IEEE Journals & Magazine..................." https://ieeexplore.ieee.org/document/982883.

[5] "Multiresolution gray-scale and rotation invariant texture classification..............." http://ieeexplore.ieee.org/document/1017623.

[6] "Multi-class Classifier-Based Adaboost Algorithm | SpringerLink." https://link.springer.com/chapter/10.1007/978-3-642-31919-8_16.

[7] "About Keras." https://keras.io/about/.

[8] "Citing TensorFlow." https://www.tensorflow.org/about/bib.

[9] "OpenCV." https://opencv.org/.

[10] "MobileNetV2 Explained | Papers With Code." https://paperswithcode.com/method/mobilenetv2.