



e-ISSN: 2319-8753 | p-ISSN: 2320-6710

IJRSET

International Journal of Innovative Research in
SCIENCE | ENGINEERING | TECHNOLOGY

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

Volume 10, Issue 7, July 2021

ISSN INTERNATIONAL
STANDARD
SERIAL
NUMBER
INDIA

Impact Factor: 7.569

Face Mask Identification in Real Time with Python

Oshi Sanjeev Jain¹, Shaikh Sadaf Siddiqua Mohsin², Dinesh Khairnar³, Abhijit P. Ingale⁴

U.G. Student, Department of Computer Science & Engineering, Shri Sant Gadge Baba College of Engineering and Technology, Bhusawal, Maharashtra, India^{1, 2, 3}

Guide, Assistant Professor, Department of Computer Science & Engineering, Shri Sant Gadge Baba College of Engineering and Technology, Bhusawal, Maharashtra, India⁴

ABSTRACT: The new Coronavirus disease (COVID-19) has impacted negatively around the globe. Wearing a protective face mask has become the new normal in recent year, and numerous public service providers may require customers to wear mask to use their services. These findings sparked research into automatic (computer vision-based) face-mask Identification tools that could help monitor public behaviour and assist limit the COVID-19 epidemic. As a resultant, we used deep learning frameworks such as Tensorflow, Keras, and OpenCV to construct a real-time facemask Identification model.

KEYWORD: Deep Learning, OpenCV, Tensorflow, Keras, Python, Dataset, COVID-19.

I. INTRODUCTION

Due to the worldwide COVID-19 corona virus outbreak, the utilization of face masks publically is changing into a lot of popular aspects. Face masks are well-tried to be effective by scientists. Masks worked by obstructing the flow of data. By 2020, the speed of modification can have accelerated significantly. According to, World Health Organization (WHO) COVID-19 is going to be declared a worldwide pandemic.

The coronavirus outbreak has sparked unprecedented international scientific collaboration. In many ways, Artificial Intelligence (AI) based on machine learning and deep learning can aid in the fight against Covid-19^[1]. This mask detection technique based on computer vision and deep learning. The proposed model can be used in conjunction with surveillance cameras to prevent COVID-19 transmission by detecting people who aren't wearing face masks. Facial detection is primarily accomplished using two methods: knowledge-based and image-based. Understanding based methods rely on human knowledge to use information like skin color and template matching. The other way, the image-based method, is based on statistical data analysis, which means we teach the system to be able to distinguish between facial and non-facial aspects^[3]. The model proposed here employs an image-based method, which necessitates the usage of two datasets to train the model, which is categorized as images with masks and images without masks. The system will recognize whether a person is wearing a facemask or not based on a real-time video taken with a camera. The approach for identifying facemasks is further divided into three subcategories as illustrated in Figure 1; they are Face Detection, Feature Extraction, and Face- mask Detection.



Figure 1: General steps of a facemask recognition structure.

The following is how the paper is laid out. The second section of the paper discusses a literature review on face mask identification. The nature of the employed dataset is discussed in Section III, as well as the details of the packages used to construct the suggested model. Our method is summarised in Section IV. The flow diagram depicts the algorithm's step. Section V demonstrates the benefits of our model. The experimental data and analysis are presented in Section VI. Finally, Section VII summarises our system's conclusion.



II. LITERATURE SURVEY

Many studies have been conducted on this major topic of Covid-19 illness, and a lot of machine learning classification models have been established. According to [6] face detection research necessitates expression recognition, face tracking, and pose estimation. Face identification is a tough task due to the fact that faces alter in size, shape, colour, and other characteristics and are not immutable. The authors [7] believe that occlusive face detection is hampered by the lack of a large dataset comprising both masked and unmasked faces. According to the study [8] Covid-19 spreads fast over the world and causing a global health problem. Deep and traditional machine learning were employed to detect the disguised face. Models are trained with the use of a dataset. The author of another paper [9] claims that COVID-19 is harmful to the environment and used Face-masknet and deep learning algorithms to recognise a masked face as well as the inappropriate manner to wear the mask. The authors [7] believe that facial expression should be excluded from 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 scope. According to the work reported [11], a convolutional neural network (CNN) in 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.

Researchers have developed a mask recognition framework that requires complex technology, a lot of time and a lot of space to load the data. We use deep learning to model the mask detector. Using MobileNetV2, we dealt with a computationally efficient system that makes it easier to call the data set. We use the CNN architecture to get better performance, where we only use pre-trained data sets for detection. We can use any camera to detect human faces. Avoiding the spread of the virus will be very beneficial to society and the people. Here, we use real-time video detection Using OpenCV (Python library). It shows the percentage of accuracy in identifying people who wear masks and those who don't. Therefore, we propose a method to identify COVID19 masks based on various deep learning algorithms and using the simplest open source programming language (i.e., Python).

III. DATASET AND PACKAGES

In this section, the technical aspects of our method are discussed. The primary aim of this research is to explore prediction systems in order to design a facemask Identification model system that utilizes the collected data in the form of images i.e., Datasets have been used for test the current method. Data Set consists of about 1,376 images with 690 images containing people with face masks and 686 images containing people without face masks. Dataset from Kaggle consists of 853 images and its countenances are clarified either with a mask or without a mask. In figure some face collections are front face pose, head turn, tilt and slant with single and multiple faces in the frame and different types of masks having different colors as well [6].

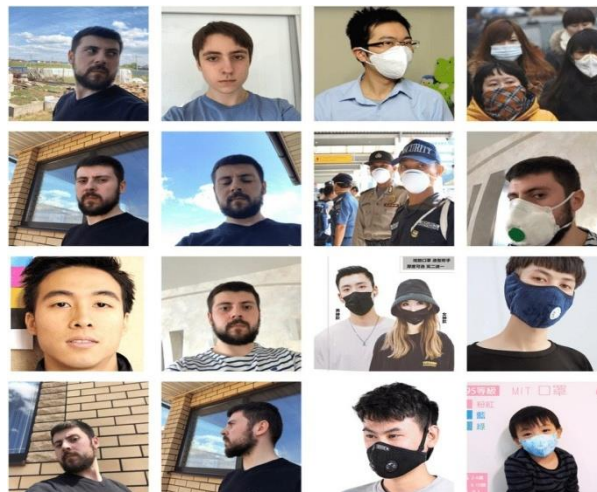




Figure.2: Samples from Dataset including faces without masks and with masks ^[10].

Packages we use in this system model are:

1.Keras

Keras offers simple reflections and building units for the creation and transportation of machine learning arrangements with high duplication of velocity. It takes full advantage of the quantifiability and cross-platform capabilities of TensorFlow. The core knowledge structures of Keras are layers and models. All the layers used in the CNN model are implemented using Keras. Along with the conversion of the class vector to the binary class matrix in data processing, it helps to compile the overall model. Keras is a human-centric API, not a machine-centric one. Keras adheres to best practices for lowering a cognitive load: it provides consistent and straightforward APIs, and it keeps the number of commands to a minimum. For most usage scenarios, it requires user activities, and it delivers Error warnings that are both clear and actionable. It also has a large number of features. Developer guidelines and documentation Keras is made up of a slew of different implementations of the commonly used neural network building blocks.

2. OpenCV

OpenCV can be used to create real-time computer vision application as a cross-platform library in machine learning. It focuses primarily on image processing, video recording, and analysis, with capabilities such as face detection and object detection. The library includes tools for recognizing items in digital photos (such as faces and figures of people, text, and so on), tracking the movement of objects, converting images, applying machine learning algorithms, and identifying common components in various photographs, among other things.

3.TensorFlow

The tensor flow is used for expressing Machine learning algorithms as a programming interface. It is used to build ML systems in a variety of fields, including sentiment analysis, voice recognition, geographic information extraction, computer vision, text summarization, information retrieval, computational drug discovery, and flaw detection. The suggested model's Sequential CNN architecture uses TensorFlow as its backend (which consists of numerous layers). It's also used to rearrange data in data processing (in form of pictures).

4. NumPy

Numpy is used for scientific computing as it is the most important Python package. It's a Python library that includes a multidimensional array object, various derived objects (such as masked arrays and matrices), and a variety of routines for performing fast array operations, such as mathematical, logical, shape manipulation, sorting, selecting, I/O, discrete Fourier transforms, basic linear algebra, and basic stabbing.

5. Convolutional neural network (CNN/ConvNet)

CNN / ConvNet is a type of deep neural network that is widely used for visual data analysis. Now, we think about matrix multiplications when we think about a neural network, however, ConvNet is not the case. It employs Convolution, a specific approach. The system uses a perceptron, a machine learning unit algorithm, to examine data for

supervised learning. Now the convolution of mathematics is a mathematical procedure using two functions that produces a third function that shows how one function is altered by the other. CNN's set to image processing, natural language processing, and different kinds of cognitive tasks.

Convolutional Neural Network Architecture

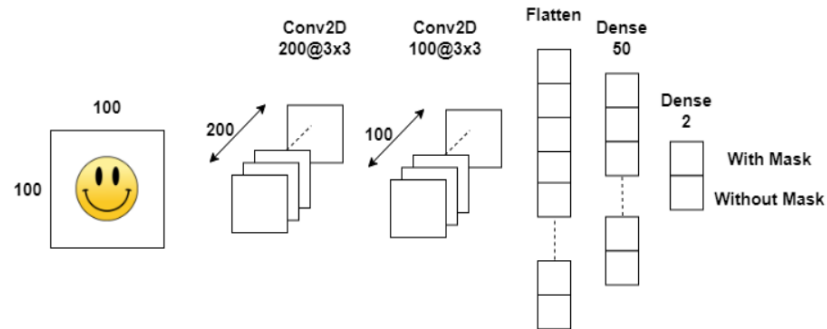


Figure.3: Convolutional Neural Network.

IV. PROPOSED METHODOLOGY

The main objective of this project is to get a system which identifies the person on image/video stream wearing face mask with the help of computer vision and deep learning algorithm by using the OpenCV, Tensor flow, Keras and Python library.

Algorithm of Face Identification Model

To detect faces, we will employ the Haar Cascade algorithm, also known as the Viola-Jones algorithm. It is essentially a machine learning object detection algorithm that is used to identify objects in an image or video. As we all know, videos are composed of frames, which are still images in which we detect faces in each frame of a video. So there isn't much difference between detecting a face in a still image and detecting a face in a real-time video.

Steps to perform the process:

Step 1: Data Visualization

Let's start by visualizing the total number of photographs in both categories in our dataset. We can observe that the 'yes' class has 690 photographs while the 'no' class has 686 photos.

Step 2: Data Augmentation

The next step is to expand our dataset so that we have a larger number of images to train with. We rotate and flip each image in our dataset in this round of data augmentation. We have a total of 2751 photographs after data augmentation, with 1380 images in the 'yes' class and 1371 photos in the 'no' class.

Step 3: Splitting the data

In this phase, we divide our data into two sets: the training set, which contains the images that will be used to train the CNN model, and the test set, which contains the images that will be used to test the model.

We use split size =0.8 in this case, which means that 80 percent of the total photos will go to the training set, while the remaining 20.

Step 4: Building the Model

The Sequential CNN model is then built using multiple layers such as Conv2D, MaxPooling2D, Flatten, Dropout, and Dense. We utilize the 'softmax' function in the final dense layer to generate a vector that represents the likelihood of each of the two classes.

Because there are just two classes, we utilize the 'adam' optimizer and 'binary cross entropy' as our loss function. Furthermore, we can use the MobileNetV2 for improved accuracy.



Step 5: Pre-Formation the CNN model

After we've built our model, let's make the 'train generator' and 'validation generator' to fit them to our model in the next step. There are 2200 images in the training set and 551 images in the test set.

Step 6: Training the CNN

This is the main step where we will match our images of the training and test sets to our sequential model that we created with the Keras library. We trained the model for 20 epochs (iterations). We can train for a greater number of epochs to achieve greater precision and avoid over fitting.

After the 20th epoch, our model is 98.86 percent accurate with the training set and 96.19 percent accurate with the test set. This model is well trained and free of any over-fitting.

Step 7: Label the Data

We'll label two probabilities for our outcomes after we've built the model ["0" for "without mask" and "1" for "with mask"]. I also use RGB values to change the color of the surrounding rectangle ["RED" for "without mask" and "GREEN" for "with mask"].

Step 8: Import Face Identification Program

Following that, we design it to use it detect whether people wearing a face mask or not by using our personal computer webcam. First, we need to implement face identification. In this case, we use the Haar Cascade classifier algorithm to identify facial features.

Step 9: Identify the Faces with and without Masks

Finally, we utilize the OpenCV package to execute an endless loop with our web camera, where we use the Cascade Classifier to detect the face. `webcam = cv2` is the code. The use of a webcam is indicated by `VideoCapture(0)`.

Each of the two classes ([without the mask, with mask]) will be predicted by the model. The label will be chosen and shown around our faces based on which probability is higher.

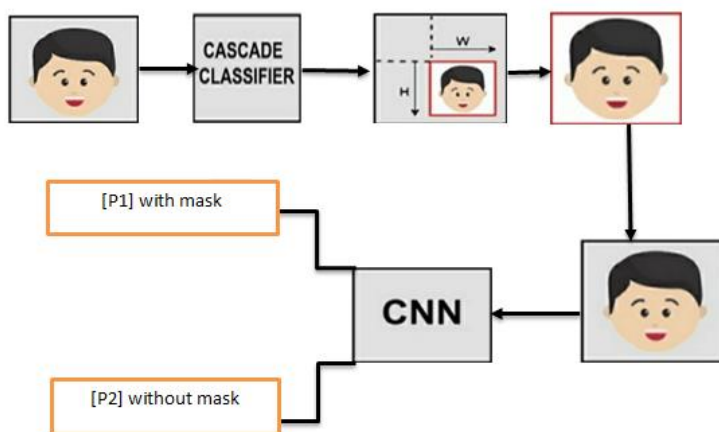


Figure.4: Overview of the System

V. ADVANTAGES OF OUR MODEL

- It is small, shared, and easily updatable.
- This type detects quickly and easily.
- It also takes little time to train the dataset.
- This face mask detection model offers a high level of precision.

- It emits a buzzer sound when the mask is worn.
- This model requires less time for training and detection.
- This technology can be used in ATMs, banks, airports, train stations, and many other public venues where two or more people interact to guarantee that public safety norms are followed. We can protect them from our technique.
- This model is far safer because it uses a touch-free approach to eliminate physical contact.
- Manual monitoring makes it difficult for security guards to determine whether or not people are wearing masks. So, with our technique, we use a webcam to recognise people's faces and protect everyone from virus spreading.

VI. EXPERIMENTAL RESULTS

The following outputs have been collected onscreen and are displayed ahead after successful implementation and testing. Figure.5 shows the result obtained: when a person is wearing a facemask, the system informs that user he or she is wearing a facemask, and when a person is not wearing a facemask, the system informs that user he or she is not wearing a facemask, along with a percentage of how well he or she covered his or her face with the mask. If a person is spotted without a mask on, a beep alarm will sound until the mask is put on. Also, if everyone wears a mask, they will be immune to the illness. Our method would substantially assist in limiting the spread of COVID-19 in this way.



Figure.5: Output Screen.

VII. CONCLUSION

We have implemented this model that has been built is relevant to the recognition of people wearing face masks in images as well as live video streams. The accuracy of the model was 98.9 percent after training it with Python, OpenCV, Keras, and Tensorflow. As technology advances and new trends emerge, we now have a revolutionary face mask detector that may be useful in public healthcare. The backbone of the system is Mobile Net, which may be used for both high and low computation scenarios.



Figure.6: Plot of Training Loss and Accuracy Graph

We used Python, OpenCV, tensor flow, keras, and CNN to detect whether or not persons were wearing face masks. The models were evaluated using image and real-time video streams. Furthermore, on a publicly available face mask dataset, the proposed technique produces state-of-the-art results. Face mask detection technology that can detect whethersomeone is wearing a mask and allow them in would be highly valuable to society

REFERENCES

- [1] Y.Fang, Y.Nie, and M. Penny, "Transmission dynamics of the covid-19 outbreak and effectiveness of government interventions: A datadriven analysis," *Journal of medical virology*, vol. 92, no. 6, pp. 645–659, 2020.
- [2] T. Karras, S. Laine and T. Aila, "A Style Based Generator Architecture for Generative Adversarial Networks", *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4396-4405, 2019.
- [3] P. Khandelwal, A. Khandelwal and S. Agarwal, "Using computer vision to enhance safety of workforce in manufacturing in a post covid world", 2020.
- [4] S. Feng, C. Shen, N. Xia, W. Song, M. Fan, and B. J. Cowling, "Rational use of face masks in the covid-19pandemic," *The Lancet Respiratory Medicine*, 2020.
<https://www.investopedia.com/terms/d/deep-learning.asp>
- [5] Z. Wang, G. Wang, B. Huang, Z. Xiong, Q. Hong, H. Wu, P. Yi, K. Jiang, N. Wang, Y. Pei et al., "Masked face recognition dataset and application", 2020.
- [6] D. Meena and R. Sharan, "An approach to face detection and recognition", 2016 *International Conference on Recent Advances and Innovations in Engineering (ICRAIE)*, Jaipur, 2016.
- [7] S. Ge, J. Li, Q. Ye and Z. Luo, "Detecting Masked Faces in the Wild with LLE- CNNs", 2017 *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, 2017, pp.
- [8] Vinitha.V1, Velantina.V2. COVID-19 FACEMASK DETECTION WITH DEEP LEARNING AND COMPUTER VISION.
- [9] Inamdar, M., & Mehendale, N. (2020). Real-Time Face Mask Identification Using Facemasknet Deep Learning Network. Available at SSRN 3663305.
- [10] "Face Mask Detection", Kaggle.com, 2020. [Online].
Available: <https://www.kaggle.com/andrewmvd/face-mask-detection.2020>.
- [11] S. Ghosh, N. Das and M. Nasipuri, "Reshaping inputs for convolutional neural network: Some common and uncommon methods", *Pattern Recognition*, vol. 93, pp. 79-94, 2019. Available:10.1016/j.patcog.2019.04.009.



**Impact Factor:
7.569**



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

 9940 572 462  6381 907 438  ijirset@gmail.com



www.ijirset.com

Scan to save the contact details