Covid-19 Face Mask Detection Using Python

Prof. Payal Kadam, Gautam Kumar, Ganesh Rajak, Devansh Bhatt
Electronics and Telecommunication, Bharati Vidyapeeth (Deemed to be University) College of Engineering, Pune

***Abstract*—** The episode of Coronavirus Disease 2019 (Covid-19) tremendously affected mankind. Till April 2022, around 512 million individuals have been impacted internationally because of the infectious spread of Covid-19. About more than 6.2 million have died until now due to the pandemic. According to WHO, wearing a mask can reduce the Covid-19 transmission significantly. The "no mask, no work" strategy has been proposed by several countries' governments. While some have levied huge fines on people without masks in public places. All these techniques have been very helpful in blocking the Covid-19 transmission. So, to stop this transmission, a system to detect masks on faces, also known as facemask recognition, is very important for ensuring public safety.

However, there is still a lot of need for future research to develop an effective facemask positioning system. As a result, the goal of this research is to draw attention to the researchers by developing a product that will aid in the containment of the covid infection on the college campus or in other public settings. It would help in the fight against the Covid-19 pandemic effectively and accurately.

Keywords—Covid-19; face mask detection; python; machine learning; TensorFlow; Keras

I. Introduction

A. Covid-19

Since the first patient infected by Corona Virus Disease 2019 (COVID-19) has been recognized in 2019, the infection spread the world exceptionally quickly. It is immediately pronounced a worldwide pandemic by the World Health Organization. Toward the finish of April 2022, a bigger number than 512 million a great many people were contaminated by the infection, and more than 6.2 million large number of individuals were dead by the infection, or the illness brought about by COVID-19 across the globe, with seriously being added consistently, as indicated by the COVID-19 dashboard delivered by the Johns Hopkins University of Medicine [1].

Covid-19 has wreaked havoc on humanity in the previous two years, regardless of age, gender, or location. Covid-19 caused not only physical pain but also a financial disaster in many rich and developing countries. As shown in Figure 1, in terms of cases the forefront is the US, and India is not far behind with a close second. [2] Therefore, this is the need for the hour to do something about this.

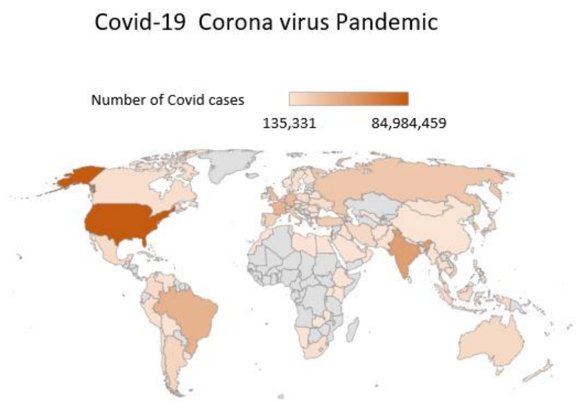

Figure 1: World Map according to covid cases

Within a brief timeframe, the infection has spread more infectiously than any others. There are a variety of reasons for this quick transmission, including a lack of attention, failure to keep up with social separation, failure to use facemasks in open gatherings, and so on. According to ongoing research, using a facemask can significantly reduce the contagious transmission of Covid and other respiratory illnesses. Even though Covid vaccinations have been developed and mass distribution has commenced as of December 2020, they just diminish the intricacies and dreariness of Covid-19 as opposed to annihilating the infection. As a result, wearing a facemask may be the most effective and secure approach to protecting a person against this infection. On open social gatherings and outside, the WHO strongly advises wearing a facemask since it prevents illness transfer through the nose or oral pit. [3]. If you have respiratory symptoms or are caring for someone who has symptoms, the World Health Organization (WHO) suggests using a face mask [4]. Facemasks (i.e., cotton textures, N-95) protect against the Covid-19 infection in the range of 50 to 95 percent [5]. To avoid being affected by Covid-19 in the situation, it is best to wear a facemask all the time. Wearing a veil is mandatory by state-run administrations in several countries. To promote mindfulness, labels such as "no mask, no work" have been proposed. In this case, determining whether someone at a public social event or organization is wearing a cover has been a big field of research.

If a person is wearing a facemask, traditional tactics such as human power or watches are not normally accessible to the screen. As a result, facemask recognition using AI or deep learning is critical. On the other hand, it is critical to ensure that a facemask is worn properly. In any scenario, while the mask will be identified, the infection's infectious transmission will not be examined. As of late, many investigations have been directed to decide whether an individual is wearing a facemask openly and keeping up with safety measures. In the battle against the pandemic Covid, many specialists and disease transmission experts have a point of view that the transmission of COVID-19 can be really limited assuming individuals wear a mask, maintain social separation, wash hands, and dynamic quarantine. It has been confirmed to be extremely compelling that wearing a cover is one of the super careful steps for people in general [6]. Subsequently, individuals are supported, even constrained by regulations and rules, to wear a mask when they need to enter public regions, like stores, emergency clinics, and air terminals.

To beat COVID-19, the government needs to guide and screen individuals out in the open spots, for instance, noncontact calm estimation through observing instruments. Notwithstanding, checking countless individuals in many spots is a difficult errand. It includes the discovery of wearing masks. A large portion of the observing instruments misses the mark on work, which can be implemented by the integration between machine learning and monitoring devices.

As COVID-19 infections began to be reported around the world, many countries responded by shutting down places like schools, workplaces, and international borders to contain the spread of the virus. On 30 January 2020, the first cases of COVID-19 were discovered in three locations in Kerala (India). On March 23, Kerala proclaimed a state of emergency, followed by the rest of the country on March 25. Contamination rates began to drop in September. As shown in Figure 2, the daily cases were at their peak in mid-September with more than 80,000-90,000 cases per day. The second wave started around the start of March 2021. It was a lot worse than the first one, which results in shortages of clinic beds, vaccines, oxygen chambers, and other clinical supplies. Around April end, in terms of new and active cases, India had surpassed the United States. On April 30, 2021, it became the first nation to record more than 400,000 new cases in 24 hours. By March 2022, India had only 22,487 cases the nation over.

India started its vaccination program on 16 January 2021 with AstraZeneca immunization (Covishield) and the native Covaxin. Afterward, Sputnik V and the Moderna vaccines were approved for emergency use as well. On January 30, 2022, India declared that it controlled around 1.7 billion portions of immunizations, and more than 720 million individuals were completely immunized.

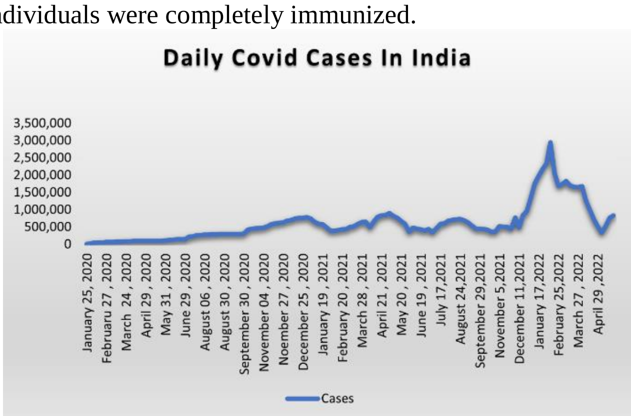

Figure 2: Covid cases in India (cases/day)

B. Motivation

The most usual motivation for making this report is to have an effective method to fight against the deadly pandemic, utilization of masks, and safeguard or regard others, including high-risk populations and individuals. Other necessary motivations were self-protection, obligation, longing for control, prerequisites, and expert advice. These infections, like COVID-19, infect the lungs by impeding the oxygen stream, which can compromise a person's life. This motivated us to develop systems and plans based on AI tools in the battle against this hazardous infection. [7].

C. Significance of Face Mask Detection

As per the centers for Disease Control and Prevention, the most common way for Covid contamination to spread is through respiratory droplets produced when people talk, cough, inhale, or sniffle with droplet size 5-10'm however transmission increments when people talk and yell loudly. Hence, to forestall quick COVID-19 contamination, numerous arrangements, like containment and lockdowns, are proposed by most of the world's state-run administrations [8]. In any case, this COVID-19 administration inefficacy can be moreover investigated with game-hypothetical situations past the public products game. Specifically, a few specialists have zeroed in on the reluctance of states to establish troublesome yet vital infection control measures (e.g., remain at-home requests and lockdowns), as well as noncooperation because of reasons other than free riding. For example, some argued that because severe stay-at-home measures can enormously influence individuals' jobs, the expense of remaining at home (combined with lockdown weakness) can outweigh the risk of disease from going out. As individual-level choices straightforwardly affect the public level adequacy of stay-at-home requests, state-run administrations might refrain from executing them due to expected low compliance, particularly from financially hindered people who don't have the advantage of remaining at home. Hence, there is a need for an effective system in place such that even low strata people will also not suffer and could go out to earn their daily bread. As discussed above, face masks can prevent the transmission of the virus, detecting the proper implementation of wearing face masks and having a proper algorithm in place is very important.

Facemask detection algorithms recognize and detect facemasks in a photo or video stream with bounding boxes. Image characterization and image localization are combined in object detection. An object's class is determined by its image characterization. For example, a photograph will be classified as either a "covered mask face" class or a "no-masked face class" based on the grouping of facemasks.

Most post-Covid-19 facemask identification methods are fundamentally Deep learning (DL)-based calculations, which are a subset of AI (ML) algorithms [9] [10]. Recurrent Neural Networks (RNNs), Deep Neural Networks (DNNs), and Long Short-Term Memory (LSTM) are all part of a DL-based network (LSTM). DNN is a brain network that employs various secret layers, and the cycle is referred to as a DL-based approach. In comparison to traditional ML-based algorithms, the DL-based algorithms have unequaled component extraction capabilities. During the learning period, the highlights that allude to the edges, corners, and surfaces of a picture should be definitively separated using a computation. DNN removes those undeniable level highlights from a picture naturally, while ML-based calculations require human oversight and are generally hand-tailored. Therefore, DL-based algorithms have been used to lead to most of the facemask recognition calculations.

II. FACE MASK DETECTION TECHNIQUES

Divya Meena et al. proposed that a face is recognized from an image that has numerous attributes in the face detection method. Face detection requires expression recognition, face tracking, and position estimation. Face detection is done with the viola Jones method, and face recognition is done with principal component analysis [11].

The Viola-Jones detector [12] uses an algorithm that extracts features using a Haar feature descriptor with an integrated picture technique and a cascaded detector to enable real-time object detection. Even though it makes use of integral pictures to speed up the process, it is still computationally expensive. The histogram of oriented gradients (HOG), a useful feature extractor for detecting humans, computes the directions and magnitudes of oriented gradients over picture cells [13].

Shiming Ge et al. stated that several misplaced expressions can be restored, and the facial cues can be reduced to a large amount using the locally linear embedding (LLE) technique and dictionaries trained on a big pool of faces with mask, synthetic banal faces [14]. In recent years, face detectors based on CNN have received a lot of attention. Zhang et al. [15] used CNN to create a multitask cascaded architecture that extracted face and landmark locations from coarse to fine.
In [16], Karl Pearson et al. devised Principal Component Analysis to recognize faces in 1901. YOLO is an anchor-based detector proposed by Redmon et al. [17], which may directly choose feature maps to achieve real-time performance.

Object detection is one of the most important achievements of deep learning and image processing since it finds and recognizes objects in images. The object recognition process begins by generating region recommendations, which are then classified into related classes [18]. Bounding boxes are one of the most frequent methods for producing localizations for items. An object detection model may be trained to recognize and detect several objects, making it versatile. Because of their robustness and great ability to extract features, deep learning-based detectors can perform effectively [19]. There are two categories: a) Single-stage detectors and b) Two-stage detectors

The single-stage detectors approach region proposal detection as a simple regression problem, learning the class probabilities and bounding box coordinates from the input image. YOLO popularized the single-stage approach by displaying real-time predictions and reaching exceptional detection speed, but it suffered from poor localization accuracy when compared to two-stage detectors, particularly when small objects are included [20]. Further, the YOLO network was developed to YOLOv2 with batch normalization, a high-resolution classifier, and anchor boxes. Although YOLOv3 is faster than Single-Shot Detector (SSD), it falls short in terms of classification accuracy [21] [22].

Lin et al. [23] presented RetinaNet to increase detection accuracy by combining an SSD and an FPN architecture with a unique focal loss function to address the class imbalance problem.

The Two-stage detectors are based on a long line of computer vision reasoning for predicting and classifying region proposals. In the first stage, it generates region proposals, which are subsequently fine-tuned in the second stage. The two-stage detector has a high detection rate but is slow.
Region-based CNN (R-CNN) [24] proposes prospective regions that may contain objects using selective search. Fast R-CNN is an R-CNN and SPPNet extension [24] [25]. To fine-tune the model, it adds a new pooling layer called Region of Interest (ROI) between shared convolutional layers. It also allows you to train a detector and a regressor at the same time without having to change your network settings. Even though Fast-R-CNN effectively combines the features of R-CNN and SPPNet, it still falls short of single-stage detectors in terms of detection speed [26].

According to [27], face recognition is used to recognize faces by lowering the image's dimensionality, resulting in a smaller database and faster processing. An issue emerges when using high-dimensional data. On a training set of face photos, it computes an Eigen picture. To minimize the dimensionality of a large dataset, PCA is applied to the Eigenface technique. This strategy is effective due to its simplicity and the limited number of stages. It's an appearance-based approach to face recognition that uses the information to encode and holistically compare individual faces, reducing space complexity and processing. The encoded image and encoded dataset are compared.

III. PROPOSED METHODOLOGY

In this pandemic circumstance, this proposed model works to recognize masked faces, which play a key role in the transmission of coronavirus from person to person. In our project, we employ the CNN algorithm to detect the mask face with more accuracy. The model can quickly recognize the faces of the mask from any angle. When someone approaches the surveillance area without a mask on. The proposed approach should be able to detect a face and a mask in motion in order.

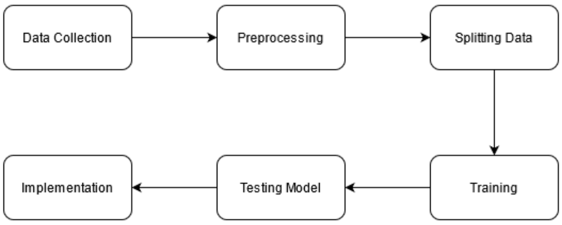

Figure 3: Flow Chart

The model's flow will be as indicated in Figure 3. We'll need some key libraries and datasets to work on this proposed architecture. The task's key problem is to correctly identify the face from the images and then check whether it has a mask on it. If not, then detect the name of the person who is not wearing the mask. And log their name.

A. Libraries

a) TensorFlow

TensorFlow, a programming interface for expressing machine learning algorithms, is used in a variety of computer science applications, including computer vision, sentiment analysis, information retrieval, geographic information extraction, text summarization, voice recognition, and computational drug discovery and flaw detection.[14] TensorFlow accepts inputs as a multi-dimensional tensor array and creates dataflow graphs and structures to define how data flows across the graph. It permits the creation of a flow chart for these inputs, which is carried out on one end and executed on the other.

b) Face Recognition

It is a python library written by a GitHub user Ageitgey. It also supports the command line [29]. This program is used in this project to identify and log the name of the individual who is not wearing the face mask.

c) Keras

Keras provides essential reflections and building units for the generation and transmission of ML arrangements at high iteration rates. It fully utilizes TensorFlow's scalability and cross-platform features. Keras' primary data structures are layers and models [30]. Keras is utilized to implement all the layers in the CNN model. It aids in the compilation of the overall model in the data processing in conjunction with the conversion of the class vector to the binary class matrix.

d) MobileNetV2

MobileNetV2 is extremely like the original MobileNet, with the exception that it employs inverted residual blocks with bottlenecking functionality. In comparison to the original MobileNet, it has a much smaller parameter count. Any image size larger than 32 × 32 is supported by MobileNets, with larger image sizes providing better performance. [31]

e) OpenCV

OpenCV (Open-Source Computer Vision Library), an open-source computer vision and machine learning software library, is used to distinguish and recognize objects, recognize faces, track camera actions, group movements in recordings, remove red eyes from photos taken with flash, follow eye gestures, obtain comparable images from a database of images, perceive the landscape, and overlay it with reality, and many more [17]. To resize and color convert data images, the proposed technique makes use of OpenCV's characteristics.

B. Dataset

The dataset is the most important component of the CNN model. The more photos in the collection, the more accurate the model will be.

The data for this study was obtained from Kaggle [33]. These are actual people's faces, both with and without masks. In this project, one dataset has been used for training the CNN model. The datasets consist of a total of 4,095 images. It has two sections:

a) Without Mask: 1930 Images

Figure 4: Dataset without mask

b) With Mask: 2165 Images

Figure 5: Dataset with mask

IV. METHODOLOGY

The suggested architecture comprises two phases: training the model and applying it to mask detection. In Figure 5, we can see the flow of the training of the model.

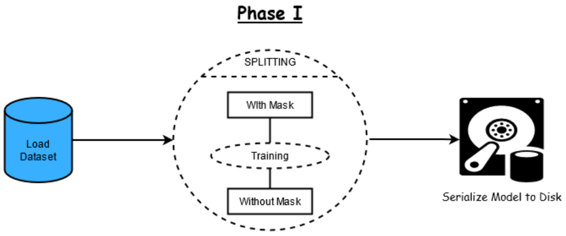

Figure 6: Training of Model

Aside from the traditional face detectors, convolutional neural network (CNN)-based models have made significant development in recent years. The suggested face detector in [34] used the feature aggregation approach [35], with CNN extracting the features. The authors of [36] proposed using attribute-related CNN to estimate candidate window confidences. A region-based CNN face detector was recently proposed in [37], which additionally considered contextual information. [38] proposed a novel grid loss to address occlusion concerns in face identification tasks. [39] presented a locally linear embedding module to obtain a similarity-based descriptor for the same goal. It performed well on the occlusion face problem when combined with the dictionary mechanism. CNNs are traditional multi-layer neural networks in which the previous layer feeds one layer and outcomes can be measured and evaluated from both layers [40]. CNN is used for classification and image processing. One or more convolution layers make up a CNN. Rather than dealing with the complete image, CNN seeks to discover characteristics that are useful within it. In addition to an input layer and an output layer, CNN has numerous secret layers. In this study, we used a deep CNN with three convolution layers. Convolution is a technique for merging two mathematical functions to create a new function. Max pooling is a sample-based discretization approach. The goal is to minimize the complexity of an input representation such that decisions about characteristics identified in binned sub-regions can be made [41].

Keras and TensorFlow are used to train the Mask detection model. The steps of the algorithm are listed below.

A. Dataset Collection

The collection of data is the first step in developing the Face Mask Recognition model. The dataset tracks those who use masks and those who don't. The model will distinguish between persons who are wearing masks and those who are not. This study uses 2165 data with a mask and 1930 data without a mask to develop the model. The image is cropped at this point until the only visible thing is the object's face.

B. Pre-Processing

For pre-processing to remove noisy disturbances, improve some relevant features, and for further analysis of the trained model, the input picture dataset must be loaded as Python data structures. Before applying face detection and matching techniques, the input image must be pre-processed.

Pre-processing techniques include noise removal, eye and mask identification, and hole filling. Noise reduction and hole filling help to eliminate erroneous face/face detection. The face image is cropped and re-localized after pre-processing. To increase the quality of the pre-processed image, histogram normalization is used [42].

C. Splitting Data

Following the pre-processing phase, the data is divided into two batches: training data (80%) and testing data (the remaining 20%). Images with and without masks are included in each batch.

D. Training Model

Building the model is the next step. The model is built in six steps: creating the augmented training picture generator, the base model with MobileNetV2, adding model parameters, compiling the model, training the model, and finally saving the model for future prediction.

Epoch	Loss	Accuracy	Val loss	Val acc
1/20	0.5124	0.7356	0.3948	0.8270
2/20	0.2756	0.8813	0.3120	0.8695
3/20	0.2346	0.9129	0.3081	0.8589
4/20	0.2245	0.9035	0.1897	0.9194
5/20	0.1713	0.9323	0.2494	0.8912
6/20	0.1672	0.9378	0.1521	0.9225
7/20	0.1325	0.9434	0.2599	0.8934
8/20	0.1296	0.9541	0.2456	0.8922
9/20	0.1530	0.9445	0.3226	0.8789
10/20	0.1363	0.9456	0.2605	0.8964
11/20	0.1180	0.9545	0.2142	0.9054
12/20	0.1324	0.9534	0.3545	0.8968
13/20	0.1045	0.9587	0.1782	0.9194
14/20	0.1279	0.9478	0.3604	0.8727
15/20	0.1235	0.9689	0.3104	0.8721
16/20	0.1070	0.9634	0.2824	0.8943
17/20	0.1099	0.9568	0.2191	0.9045
18/20	0.1089	0.9570	0.2588	0.8997
19/20	0.1065	0.9580	0.2163	0.9145
20/20	0.0915	0.9689	0.2505	0.9076

Figure 7: Log of loss and accuracy while training the model

E. Testing/Evaluation

There are procedures in testing the model to ensure that it can predict well. Predictions on the testing set are the initial step. There will be 20 iterations for checking the loss and accuracy. In figure 7, we can see how accuracy is increasing on time. When the accuracy line remains stable, it indicates that more iteration is not required to improve the model's accuracy.

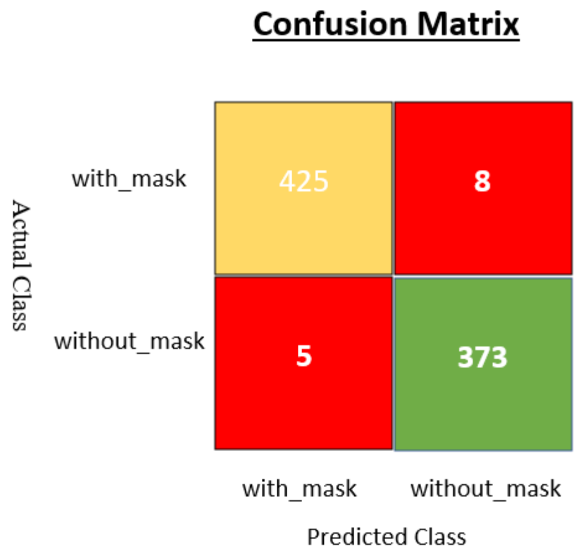

Figure 8: Confusion Matrix

There were 433 images with masks and 378 images without masks used to evaluate/test the model. The confusion matrix is given in figure 8.

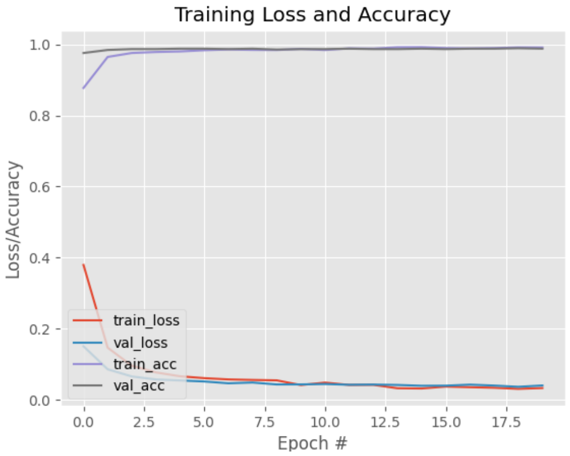

Figure 9: Training loss and accuracy graph

F. Implementation

The model used in the video. After reading the video frame by frame, the face detection algorithm is applied. If a face is detected, the process moves on to the next step. Reprocessing will be done on detected frames with faces, including shrinking the image size, converting to an array, and preprocessing input with MobileNetV2.

The next step is to use the saved model to predict input data. Predict the input image using a previously created model. In addition, the video frame will be labeled with mask or no mask as well as the predicted percentage. Figure 12-14 shows an example of how to use the model.

The final phase is facial recognition; in this scenario, anytime a face appears without a mask, the model will attempt to infer the person's name based on the existing database and log in to take further action. This is the final execution part of the process. The library itself provides the best way to recognize the face itself. In Figures 12 and 14, we can see the name of the person without a mask.

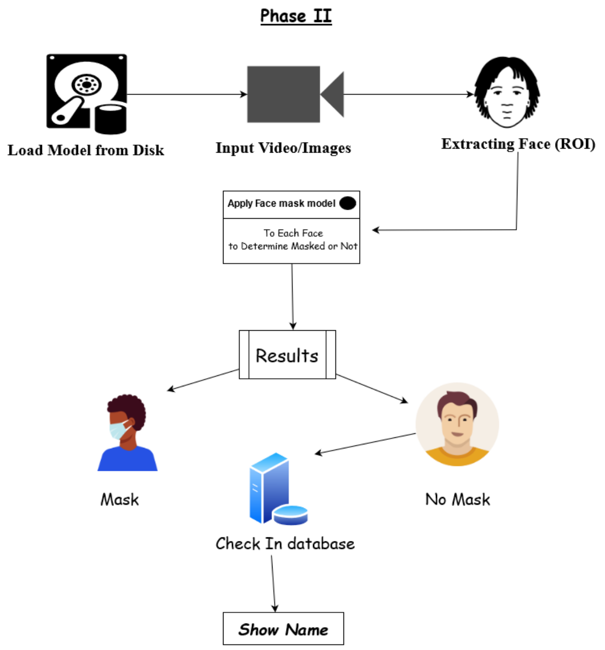

Figure 10: Working block diagram

V. PERFORMANCE METRICS

The testing of the model includes processes to check that the model is predicting accurately. Making predictions based on the testing data set is the first step. While training the model, loss and accuracy are recorded after each defined iteration. The results of training this model show that accuracy continues to improve as loss decreases. There is no need for extra iterations after a certain point when the accuracy is stable.

The next step is to analyze the overall performance of the MobileNetV2 model using the performance metrics listed below.

$$Precision = \frac{True\ Positive\ (TP)}{True\ Positive\ (TP) + False\ Positive\ (FP)}$$

$$Recall = \frac{True\ Positive\ (TP)}{True\ Positive\ (TP) + False\ Negative\ (FN)}$$

$$F1\ Score = 2 * \frac{Precision * Recall}{Precision + Recall}$$

The following are the performance metrics parameters:

- True Positive: These are successfully predicted positive values, indicating that the value of the real class is yes, as well as the value of the predicted class.
- True Negative: These are accurately predicted negative values, indicating that the value of the real class is no, and the value of the projected class is not as well.

- False Positive: When the predicted class is yes but the actual class is no.
- False Negative: When the predicted class is no but the actual class is yes.
- Precision: Precision is a measure of how well a model recognizes relevant data points like faces.
- Recall: Recall is a metric that determines whether a classifier model properly identifies True Positives.
- F1 Score: The f1 score is a better indicator of how well Precision and Recall are balanced.

	Precision	Recall	F1-Score	Support
With mask	0.99	0.98	0.98	433
Without mask	0.97	0.98	0.91	386
Accuracy			0.98	819
Macro avg	0.98	0.98	0.98	819
Weighted avg	0.98	0.98	0.98	819

Figure 11: Classification Report

Figure 11 shows the performance metrics of the MobileNetV2 model after evaluation.

VI. RESULTS AND DISCUSSION

The outcomes are more in line with what the model predicted. The mask identification is done with the help of a camera and produces accurate results. When a person's face is detected in the camera frame, the model will display a green or red frame over the face (Figure 12-14)

In the camera, a person who is not wearing a mask will have a red frame over his face, while someone who is wearing a mask will have a green frame. When the system detects a face without a mask, it will attempt to retrieve the person's name from the available database and display the name if exists (Figures 12 and 14).

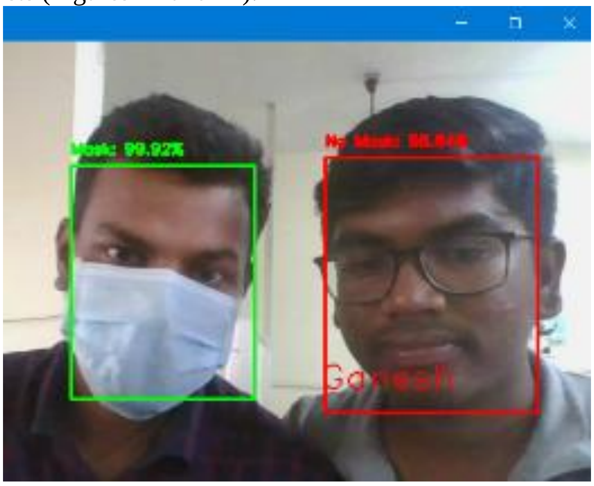

Figure 12: Live Video implementation

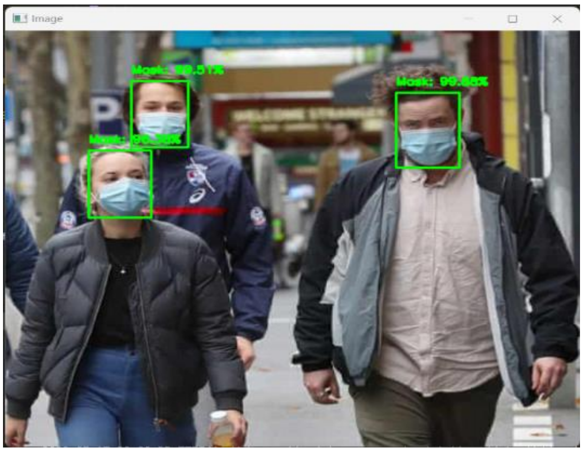

Figure 13: Crowded image detection

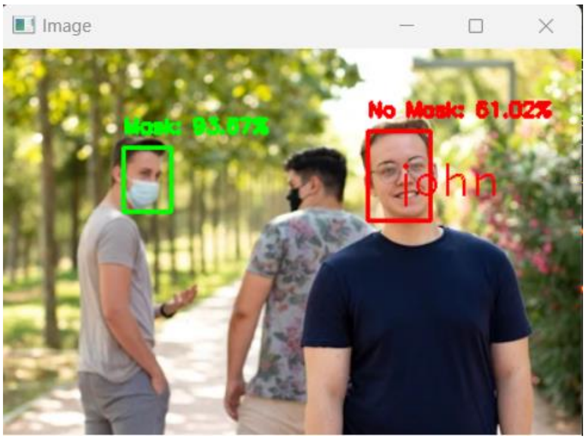

Figure 14: Detecting mask and without mask along with name

VII. CONCLUSION

This paper describes a method for a smart city that can help decrease the transmission of coronavirus by alerting authorities when someone is not wearing a COVID-19 protective mask. The effort is motivated by those who break the rules that are in place to prevent the transmission of coronavirus. A deep learning technique is utilized to detect the mask on the face in the system's face mask detection architecture. Labeled picture data were collected to train the model, with the images being facial images with and without masks. A face mask is detected with 98% accuracy by the suggested technique. The technique presented in this study will be a useful tool for requiring everyone to wear a facial mask in public places.

REFERENCES

[1] COVID-19 Dashboard by the Center for Systems Science and Engineering, April. 2022 https://coronavirus.jhu.edu/

[2] Cumulative number of cases (by number of days since 10,000 cases) https://www.worldometers.info/coronavirus/worldwide-graphs/

[3] N. Leung, D. Chu, E. Shiu, K.-H. Chan, J. Mcdevitt, B. Hau, et al., "Respiratory virus shedding in exhaled breath and efficacy of face masks", Nature Med., vol. 26, pp. 676-680, May 2020.

[4] S. Feng, C. Shen, N. Xia, W. Song, M. Fan, and B. J. Cowling, "Rational use of face masks in the COVID-19 pandemic", Lancet Respiratory Med., vol. 8, no. 5, pp. 434-436, May 2020.

[5] How Mask Antiviral Coatings May Limit COVID-19 Transmission, Jun. 2020 https://www.optometrytimes.com/view/how-mask-antiviral-coatings-may-limit-covid-19-transmission

[6] COVID-19 pandemic https://en.wikipedia.org/wiki/COVID-19_pandemic

[7] OECD Policy Responses to Coronavirus (COVID-19) https://www.oecd.org/coronavirus/policy-responses/using-artificial-intelligence-to-help-combat-covid-19-ae4c5c21/

[8] Centers for Disease Control and Prevention https://www.cdc.gov/coronavirus/2019-nCoV/

[9] M. Inamdar and N. Mehendale, Real-time face mask identification using facemask net deep learning network, India, Jul. 2020

[10] A. Chavda, J. Dsouza, S. Badgujar and A. Damani, "Multi-stage CNN architecture for face mask detection", arXiv:2009.07627, 2020

[11] D. Meena and R. Sharan, "An approach to face detection and recognition," 2016 International Conference on Recent Advances and Innovations in Engineering (ICRAIE), 2016, pp. 1-6, doi: 10.1109/ICRAIE.2016.7939462.

[12] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features", Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR), pp. 1-9, Dec. 2001.

[13] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection", Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR), pp. 886-893, Jun. 2005.

[14] 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), 2017, pp. 426-434, doi: 10.1109/CVPR.2017.53.

[15] X. Wang, K. C. K. Chan, K. Yu, C. Dong and C. C. Loy, "EDVR: Video Restoration With Enhanced Deformable Convolutional Networks," 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2019, pp. 1954-1963, doi: 10.1109/CVPRW.2019.00247.

[16] Y. Lecun, L. Bottou, Y. Bengio and P. Haffner, "Gradient-based learning applied to document recognition," in Proceedings of the IEEE, vol. 86, no. 11, pp. 2278-2324, Nov. 1998, doi: 10.1109/5.726791.

[17] S. Zhang, X. Zhu, Z. Lei, H. Shi, X. Wang and S. Z. Li, "FaceBoxes: A CPU real-time face detector with high accuracy," 2017 IEEE International Joint Conference on Biometrics (IJCB), 2017, pp. 1-9, doi: 10.1109/BTAS.2017.8272675.

[18] Qiao S., Liu C., Shen W., Yuille A. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition. 2018. Few-Shot Image Recognition by Predicting Parameters from Activations.

[19] Z. Zou, Z. Shi, Y. Guo and J. Ye, "Object detection in 20 years: A survey", arXiv:1905.05055, 2019, [online] Available: http://arxiv.org/abs/1905.05055.

[20] J. Redmon, S. Divvala, R. Girshick, A. Farhadi, You only look once: Unified, real-time object detection, in: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2016, vol. 2016-Decem, pp. 779–788, doi: 10.1109/CVPR.2016.91.

[21] Kumar A., Zhang Z.J., Lyu H. Object detection in real time based on improved single shot multi-box detector algorithm. J. Wireless Com. Netw. 2020;2020:204. doi: 10.1186/s13638-020-01826-x.

[22] Morera Á., Sánchez Á., Moreno A.B., Sappa Á.D., Vélez J.F. SSD vs. YOLO for detection of outdoor urban advertising panels under multiple variabilities. Sensors (Switzerland) 2020 doi: 10.3390/s20164587.

[23] T.-Y. Lin, P. Goyal, R. Girshick, K. He and P. Dollar, "Focal loss for dense object detection", Proc. IEEE Int. Conf. Comput. Vis. (ICCV), pp. 2980-2988, Oct. 2017.

[24] R. Girshick, J. Donahue, T. Darrell and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation", Proc. IEEE Conf. Comput. Vis. Pattern Recognit., pp. 580-587, Jun. 2014.

[25] Inamdar M., Mehendale N. Real-Time Face Mask Identification Using Facemasknet Deep Learning Network. SSRN Electron. J. 2020 doi: 10.2139/ssrn.3663305.

[26] Nguyen N.D., Do T., Ngo T.D., Le D.D. An Evaluation of Deep Learning Methods for Small Object Detection. J. Electr. Comput. Eng. 2020;2020 doi: 10.1155/2020/3189691.

[27] Turk, M., & Pentland, A. (1991). Eigenfaces for recognition. Journal of cognitive neuroscience, 3(1), 71-86.

[28] "TensorFlw White Papers", TensorFlow, 2020, [online] Via: https://www.tensorflow.org/about/bib

[29] https://github.com/ageitgey/face_recognition

[30] "Keras documentation: About Keras", 2020, [online] Via: Keras.io.

[31] https://keras.io/api/applications/mobilenet/

[32] "OpenCV", 2020, [online] Via: Opencv.org.

[33] https://www.kaggle.com/

[34] B. Yang, J. Yan, Z. Lei and S. Li, "Convolutional channel features", Proc. Int. Conf. Comput. Vis., pp. 82-90, Dec. 2015.

[35] P. Dollar, R. Appel, S. Belongie and P. Perona, "Fast feature pyramids for object detection", IEEE Trans. Pattern Anal. Mach. Intell., vol. 36, no. 8, pp. 1532-1545, Aug. 2014.

[36] S. Yang, P. Luo, C. C. Loy and X. Tang, "From facial parts responses to face detection: A deep learning approach", Proc. IEEE Int. Conf. Comput. Vis., pp. 3676-3684, Dec. 2015.

[37] C. Zhu, Y. Zheng, K. Luu and M. Savvides, "CMS-RCNN: Contextual multi-scale region-based CNN for unconstrained face detection", Proc. Comput. Vis. Pattern Recognit., pp. 57-79, Aug. 2017.

[38] M. Opitz, G. Waltner, G. Poier, H. Possegger and H. Bischof, "Grid loss: Detecting occluded faces", Proc. Eur. Conf. Comput. Vis., vol. 9907, pp. 386-402, Oct. 2016.

[39

[40

[41

[42