

Face Mask Detection System for COVID_19 Pandemic Precautions using Deep Learning Method

Sneha Sen
M Tech Scholar

Department of Computer Science & Engineering
Lakshmi Narain College of Technology, Indore

Dr. Harish Patidar
Professor & HOD

Department of Computer Science & Engineering
Lakshmi Narain College of Technology, Indore

Abstract: The World Health Organization (WHO) has stated that there are two ways in which the spread of COVID 19 virus takes place that are respiratory droplets and physical contact. So, avoiding the spread of this virus need some precautionary steps to be taken that are social distancing and the wearing of masks. Among these two precautions the mask wearing is considered as the important factor for the spread of COVID 19 virus because these droplets can land on any surface. So, to keep track of the people that are wearing mask or not is more important. Here we have presented a mask detection system that is able to detect any type of mask and masks of different shapes from the video streams for following the rules that are applied by the government. Deep learning algorithm is used here and the PyTorch library of python is used for mask detection from the images/video streams. The proposed system is able to detect the mask wearing people and those one who are not wearing the masks.

Keywords: COVID 19 Pandemic, Mask Detection System, Deep Learning Algorithm, World Health Organization, PyTorch Library, Convolutional Neural Network.

I. INTRODUCTION

COVID-19 pandemic has come in existence in December 2019 and the first case of this virus is found in China. From there this virus started spreading in all over world and almost in every country of this world [1]. Staying home, travel by self driving, avoid the transportation, avoiding to travel to the infected cities all these measures can be taken for stopping the spread of covid 19 disease [2]. The two main reasons found behind the spread of this virus are stated by WHO as the respiratory droplets and the physical contact of peoples [3]. The respiratory droplets from peoples may reach to other persons that are in contact (within 1m) if the infected person sneezes or coughs, and this droplet may spread in air and may reach to various surfaces that are closer. This virus is remains on almost every surface and this may lead to the contact transmission.

The precautions stated by the governments of every country are of social distancing and the wearing of masks [4]. The masks that are used by peoples are of various types for example medical masks, surgical masks, procedure masks and also some designer masks of different shapes (cup shape) [5]. These masks are also used for the stoppage of the spreading of respiratory droplets of the infected person and this mask may also provide the adequate breathability, fluid

penetration resistance and high filtration. Social distancing is a simple phenomenon of maintaining distance between objects and peoples of minimum 2 meters.

It is to be noted that the role of Internet of things, artificial intelligence, blockchain, unmanned aerial vehicles etc is going to be very crucial in detection of covid19 disease [6]. So here in our proposed approach we have designed a mask detection method that is able to detect various types of masks and of different designs. Our approach is able to detect the peoples wearing mask or not from images and video streams with the help of computer vision, for this deep learning algorithm is applied and the PyTorch library of python is used for implementation. The approach first trains the deep learning model that is MobileNetV2 and then applying this for the detection of masks from images/video streams.

II. LITERATURE REVIEW

Face recognition system is helpful in various fields. One of the areas is presentation attack detection for the recognition of the face has been introduced in this paper by authors. For this they have used 3D silicon face masks for the real objects. They have used the new database with 8 custom 3D silicon masks along with the bona fide presentation. The effectiveness of the presentation attack detection for such 3D silicon mask has been evaluated and it is found to be good [7].

Here in this paper the authors have proposed a mask detection system for the health care personal inside the operation theatre. As the health care personal need to wear a mask in the operation theatre and the proposed system will alert for any personal not wearing the mask. There are two detection system used for face and medical mask wearing. Their system achieved almost 90% recall and less than 5% of false positive rate. They have worked for the medical mask detection from the images that are taken from 5m distance by cameras [8].

In this paper the authors have worked on the masked face detection from the video. The masked person is detected in this presented approach and mainly 4 steps are performed for the detection that are estimation of distance between camera and person, detection of eye line, detection of part of face and detection of eye. They have analyzed their algorithm on various video surveillance systems and achieved a fine accuracy [9].

The paper presents a model for masked face detection for the security purpose. For this they have presented a special cascade CNN which works on three layers of CNN for the detection of masked face. Also, the authors have worked on the self created MASKED FACE dataset as the already present masked face dataset is not that much sufficient to evaluate algorithms more precisely. The proposed CNN model worked well and the detection of masked faces is done accurately [10].

Another work in this field is done for the detection of masked face and after that the detection of the original face. The proposed work has been carried out in two stages, first one is to detect the masks that are covering larger area of face than needed and the secondly to get the face that is not present in the training dataset. They have applied GAN based algorithm and used celebA dataset for training and achieved higher accuracy [11].

Here another work related to masks have been done by the authors, they have worked on the face detection from the faces that are covered with the masks. The authors have discovered the images that are covered with masks from the video and then original images are detected from this masked face. For this they have proposed Multi-task Cascaded CNN and SVM for classification. The proposed system is able to detect the masked faces and the original faces of peoples [12].

Authors here have proposed a face recognition system that is based on fully convolutional network. The authors have worked on the generation of face segmentation from any size of images. FCN here works as the training method for the extraction of features also the gradient descent is for the training with the loss function as of binomial cross entropy.

They have achieved 93.884% mean pixel accuracy from the proposed approach [13].

Here the authors have worked on the face detection that is very fast and reliable method presented by Viola. This algorithm uses skin mask for the detection of face which gives result 4 times faster. Here the training is done for modeling the system with 2 eyes or 1 eye and 1 nose for lowering the false detection rate. They got 2.4 percent false negative in place of 10% [14].

III. PROPOSED APPROACH

The proposed approach for the mask detection is stated here and the various implementation steps involved in this approach. The flowchart for this is presented below that states the overall flow of the approach. The model starts with the loading of the dataset for mask detection and the preprocessing of data is done with the help of PyTorch torchvision. Then after the generation and the training of model is done that is MobileNetV2. Then after the serialization of the face mask classifier takes place. The first phase of this model is as stated in above statements and the MobileNetV2 classifier training process is done by using the PyTorch framework of deep learning.

Now after getting the serialization of face mask we will load this face mask classifier, and then the faces are loaded from the available image / video stream. Now the next stage comes with the preprocessing of the PyTorch transforms and the OPENCV of python. Here at the end the face mask detector detects the people with the masks or without masks. And the results are stated at last in show figure 1.

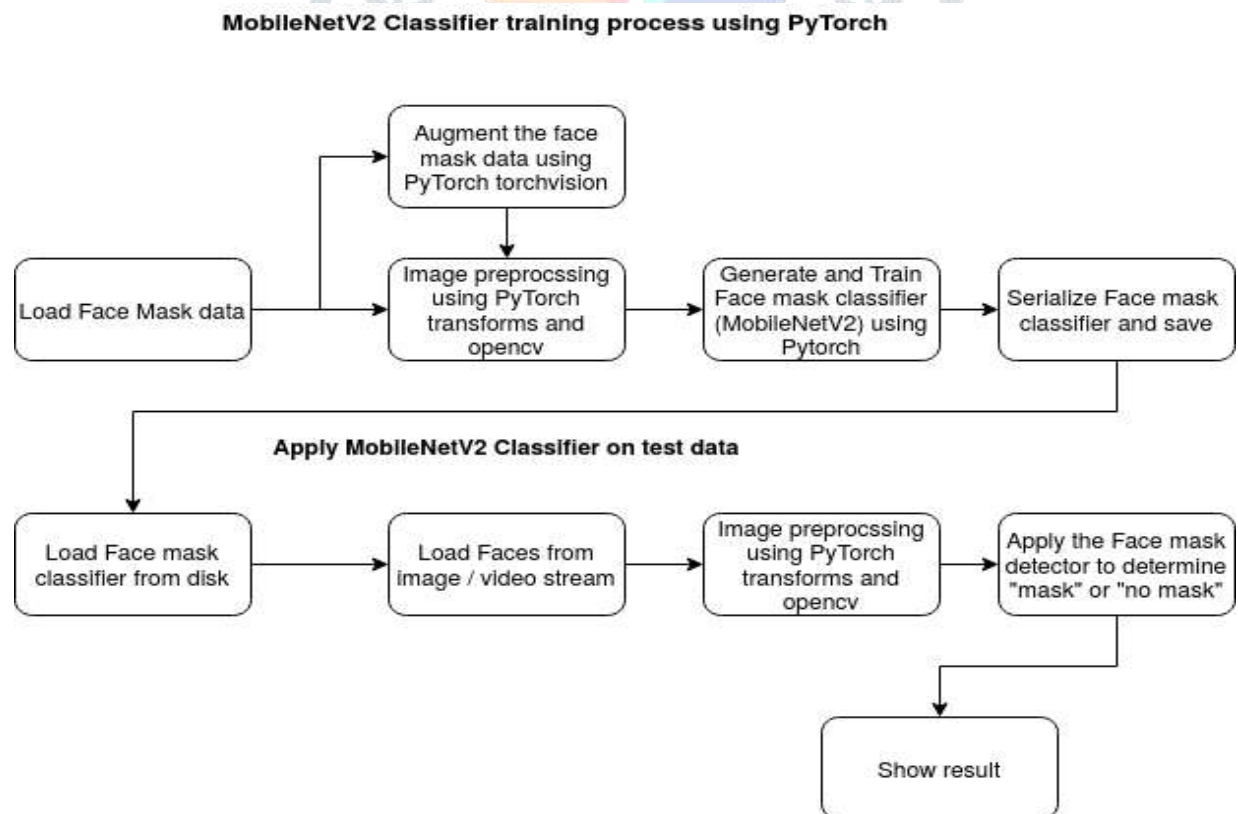


Figure 1. Flowchart for the Proposed Mask Detection.

IV. IMPLEMENTATION

The implementation of the proposed approach is presented here and the various steps followed and the data used for implementation is stated. Also, the selection of each and every library, framework, algorithm and the dataset are stated in this section.

Data at Source

Here for our experiment we have downloaded the raw images from the article on PyImageSearch and the task of augmentation on this image is done by the OpenCV. “Mask” and “NoMask” are the tags given to the images at the start itself. The size and resolutions of the taken images are varied because they are taken from various devices having different configurations.

Data Preprocessing

Here the preprocessing steps have been stated which are performed for making the images noise free and make them clear for detection, and this preprocessed image could be given to the neural networks model as input.

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

For the implementation of such deep learning network the following options can be seen as per availability.

1. TensorFlow
2. Keras
3. PyTorch
4. Caffe
5. MxNet
6. Microsoft Cognitive ToolKit

We have made use of the PyTorch as it can be easily implemented on the python platform, so one with a basic knowledge of python can make use of this to build their own deep learning model along with this some of the advantages can be seen as:

1. Data Parallelism
2. Framework like architecture.

Algorithm development using PyTorch is done by making use of the following module,

- PyTorch DataLoader – Loading of data is done from the folder of images
- PyTorch DataSets ImageFolder – image sources are located by using this also it provides already developed module for labeling the targeted variable.
- Pytorch Transforms – It helps in the application of the steps of preprocessing on the given images when it is being read from the source folder.
- PyTorch Device – This is used for the identification of the capability of the system such as CPU, GPU power

for training of the model. The usage of the system can be switched with the help of this.

- Pytorch TorchVision – loading of already created libraries can be done by this. Like the pre trained models, image sources and many more. In PyTorch it comes as a core
- PyTorch nn – it is seen as the core module. This is mainly used for the creation of Deep neural network model. It provides the libraries that are needed for the model building. Like Linear layer, Convolution layer with 1D, conv2d, conv3d, sequence, CrossEntropy Loss (loss function), Softmax, ReLu and so on.
- PyTorch Optim – model optimizer can be defined by this. Proper data learning is provided by this module. Ex. ADAM, SDg, etc.
- Pytorch PIL – image loading from the source is done by this.
- PyTorch AutoGrad – automatic differentiation on the given operations of tensors can be carried out in this module. Like one line of code. Gradient calculation can be done by backward (). It is seen very useful while performing back propagation in DNN.

Algorithm for image classification using PyTorch

CNN provides various architectures such as ResNet, Inception, AlexNet, MobileNet, etc. We have implemented our approach on the MobileNetV2 architecture as it is light weight and seen very efficient.

MobileNetV2

This model is inspired from the previous version as MobileNetV1, and uses depth wise separable convolution as per the building blocks. MobileNetV2 provides us some additional features as: a) linear bottlenecks in between the layers, b) in between bottlenecks it provides shortcut connections. Below figure shows the structure of MobileNetV2.

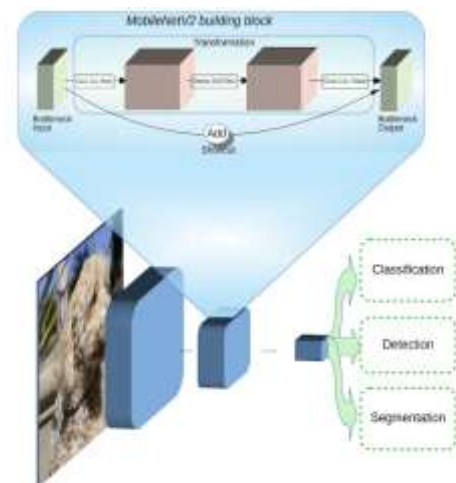


Figure 2. MobileNetV2 Architecture

MobileNetV2 model consists of various layers that can be seen in the table given below. The PyTorch provides a model in library in TorchVision for creating MobileNetV2, instead of creating own model.

ImageNet dataset gives the basis for defining the weights for every layer in the model. This weight states the padding, strides, kernel size, input and output channels.

Table 1. MobileNetV2 Inputs and Operations

Input	Operator	<i>t</i>	<i>c</i>	<i>n</i>	<i>s</i>
$224^2 \times 3$	conv2d	-	32	1	2
$112^2 \times 32$	bottleneck	1	16	1	1
$112^2 \times 16$	bottleneck	6	24	2	2
$56^2 \times 24$	bottleneck	6	32	3	2
$28^2 \times 32$	bottleneck	6	64	4	2
$14^2 \times 64$	bottleneck	6	96	3	1
$14^2 \times 96$	bottleneck	6	160	3	2
$7^2 \times 160$	bottleneck	6	320	1	1
$7^2 \times 320$	conv2d 1x1	-	1280	1	1
$7^2 \times 1280$	avgpool 7x7	-	-	1	-
$1 \times 1 \times 1280$	conv2d 1x1	-	k	-	-

Why to use MobileNetV2?

For the ImageNet dataset MobileNetV2 gives better results as compared to MobileNetV1 and ShuffleNet (1.5) in terms like size and cost for computation. When working on smaller size dataset this model can perform better.

Table 2. Results for ImageNet dataset

Network	Top 1	Params	MAdds	CPU
MobileNetV1	70.6	4.2M	575M	113ms
ShuffleNet (1.5)	71.5	3.4M	292M	-
ShuffleNet (x2)	73.7	5.4M	524M	-
NasNet-A	74.0	5.3M	564M	183ms
MobileNetV2	72.0	3.4M	300M	75ms
MobileNetV2 (1.4)	74.7	6.9M	585M	143ms

Face Mask Detector: Training with Face Mask As we have stated earlier here, we made use of smaller dataset that is augmented data. Using PyTorch transforms, PIL and datasets the preprocessing is performed on the source images at very first. Face mask training files look alike this as shown below.



Figure 3: Example for Training Set

Model Overwritten

We have used MobileNetV2 for developing the model so that it can be implemented on any mobile device. 4 fully connected layers were built on the top of MobileNetV2 as 4 sequential layers for developing the model. The layers are
 1. Average Pooling layer with 7x7 weights
 2. Linear layer with ReLu activation function
 3. Dropout Layer
 4. Linear layer with Softmax activation function with the result of 2 values.

The last layer as the softmax layer provides the result at last in following two classifications as “mask” or “no mask”.

Final Network Model Architecture / Flow

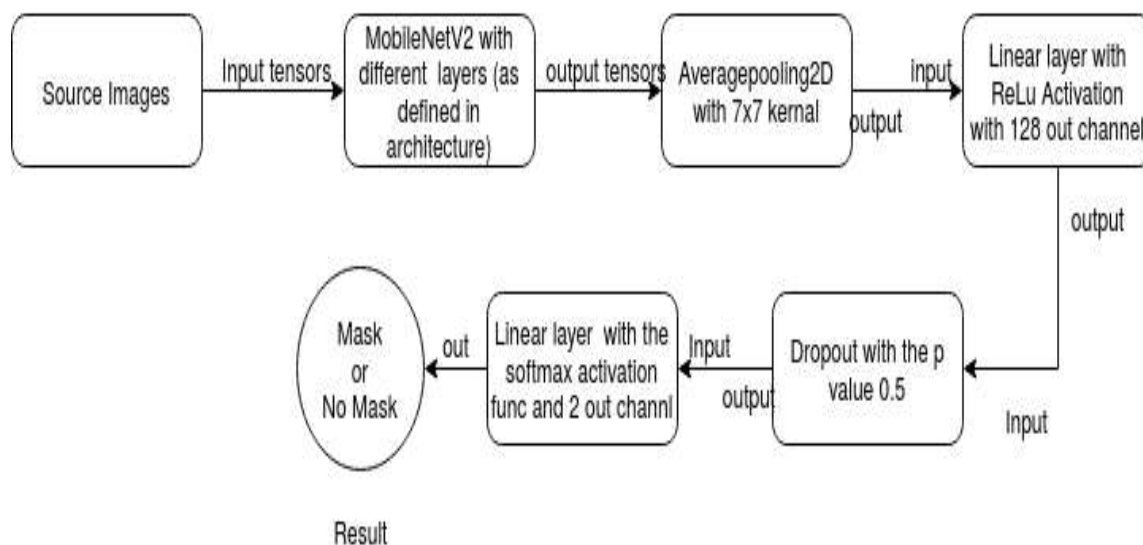


Figure 4. Final NetworkModel Architecture/Flow

V. RESULTS

The result section discusses the accuracy of the proposed mask detection approach. The dataset has been divided in two sets that are training and the validation set. The below graph states the accuracy obtained for the image classification for the training and the validation set. Here the training set contains 5000 images with mask and 4000 images without mask.

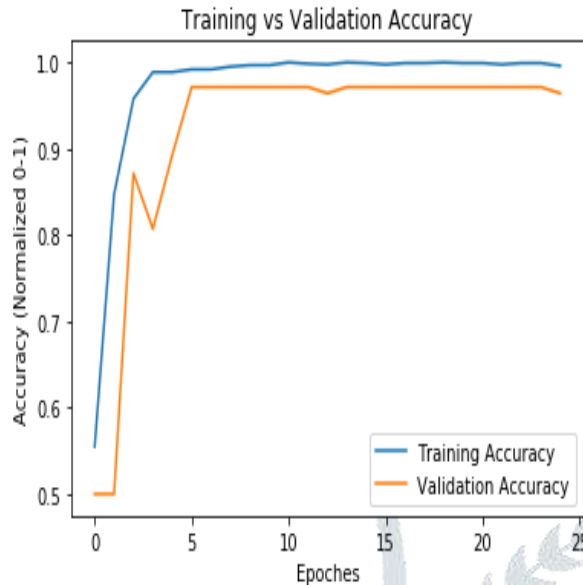


Figure 5: Training vs Validation Accuracy

Below we have presented some of the running examples of the detection of masks from both the images and video streams.

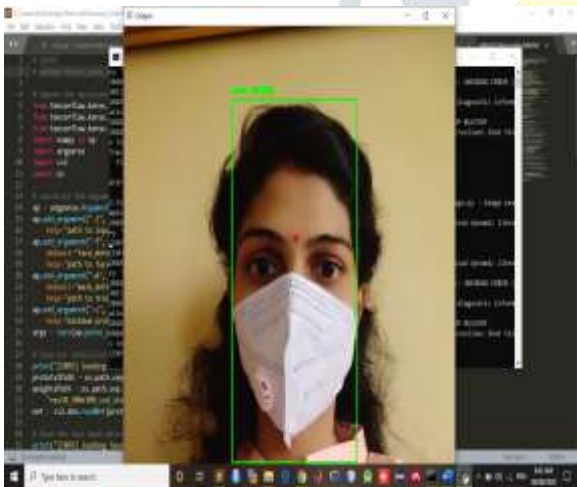


Figure 6. Mask Detection from Image

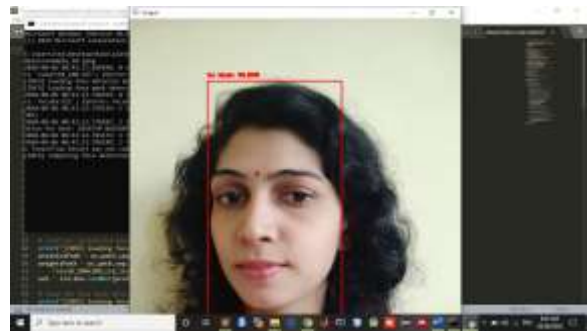
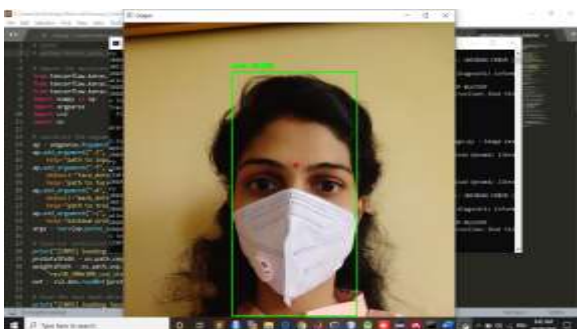


Figure 7. Mask detection from live Video Stream

VI. CONCLUSION

COVID-19 pandemic has come with various challenges to the world and the spread of this virus should be controlled as this virus has affected more than one crore peoples all over the world and the counting is still going on. One of the major precautions is to wear mask for the stopping of the spread of the respiratory droplets of infected peoples through cough or sneeze as well the healthy people should be covered with mask. So here we have presented an approach that uses deep learning algorithm and the framework of MOBILENETV2 is used for implementation along with the PyTorch and OPENCV of python. The results state that the proposed model is capable of detecting the peoples with or without masks from the images as well from the video streams. The accuracy for the training and validation set is compared and found to be of 79.24 %.

Reference:

- [1] A. Waheed, M. Goyal, D. Gupta, A. Khanna, F. Al-Turjman, and P. R. Pinheiro, "CovidGAN: Data Augmentation Using Auxiliary Classifier GAN for Improved Covid-19 Detection," *IEEE Access*, vol. 8, pp. 91916–91923, 2020, doi: 10.1109/ACCESS.2020.2994762.
- [2] K. Frp *et al.*, "6WXG \ RQ (SLGHPLF 3UHYHQWLRQ DQG & RQWURO 6WUDWHJ \ RI & 29 ; % DVHG RQ 3HUVRQQHO) ORZ 3UHGLFWLRQ," pp. 688–691, 2020, doi: 10.1109/ICUEMS50872.2020.00150.
- [3] A. A. R. Alsaedy and E. Chong, "Detecting Regions At Risk for Spreading COVID-19 Using Existing Cellular Wireless Network Functionalities," *IEEE Open J. Eng. Med. Biol.*, pp. 1–1, 2020, doi: 10.1109/ojemb.2020.3002447.
- [4] R. F. Sear *et al.*, "Quantifying COVID-19 Content in the Online Health Opinion War Using Machine Learning," *IEEE Access*, vol. 8, pp. 91886–91893, 2020, doi: 10.1109/ACCESS.2020.2993967.
- [5] J. Barabas, R. Zalman, and M. Kochlan, "Automated evaluation of COVID-19 risk factors coupled with real-time , indoor , personal localization data for potential disease identification , prevention and smart quarantining," pp. 645–648, 2020.
- [6] V. Chamola, V. Hassija, V. Gupta and M. Guizani, "A Comprehensive Review of the COVID-19 Pandemic and the Role of IoT, Drones, AI, Blockchain, and 5G in Managing its Impact," in *IEEE Access*, vol. 8, pp. 90225-90265, 2020, doi: 10.1109/ACCESS.2020.2992341.
- [7] R. Ramachandra *et al.*, "Custom silicone face masks: Vulnerability of commercial face recognition systems & presentation attack detection," *2019 7th Int. Work. Biometrics Forensics, IWBF 2019*, pp. 1–6, 2019, doi: 10.1109/IWBF.2019.8739236.
- [8] R. Paredes, J. S. Cardoso, and X. M. Pardo, "Pattern recognition and image analysis: 7th Iberian conference, IbPRIA 2015 Santiago de Compostela, Spain, June 17–19, 2015 proceedings," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 9117, pp. 138–145, 2015, doi: 10.1007/978-3-319-19390-8.
- [9] G. Deore, R. Bodhula, V. Udpikar, and V. More, "Study of masked face detection approach in video analytics," *Conf. Adv. Signal Process. CASP 2016*, pp. 196–200, 2016, doi: 10.1109/CASP.2016.7800000.

- 10.1109/CASP.2016.7746164.
- [10] W. Bu, J. Xiao, C. Zhou, M. Yang, and C. Peng, "A cascade framework for masked face detection," *2017 IEEE Int. Conf. Cybern. Intell. Syst. CIS 2017 IEEE Conf. Robot. Autom. Mechatronics, RAM 2017 - Proc.*, vol. 2018-January, pp. 458–462, 2017, doi: 10.1109/ICCIS.2017.8274819.
- [11] N. Ud Din, K. Javed, S. Bae, and J. Yi, "A Novel GAN-Based Network for Unmasking of Masked Face," *IEEE Access*, vol. 8, pp. 44276–44287, 2020, doi: 10.1109/ACCESS.2020.2977386.
- [12] M. S. Ejaz and M. R. Islam, "Masked face recognition using convolutional neural network," *2019 Int. Conf. Sustain. Technol. Ind. 4.0, STI 2019*, vol. 0, pp. 1–6, 2019, doi: 10.1109/STI47673.2019.9068044.
- [13] T. Meenpal, A. Balakrishnan, and A. Verma, "Facial Mask Detection using Semantic Segmentation," *2019 4th Int. Conf. Comput. Commun. Secur. ICCCS 2019*, pp. 1–5, 2019, doi: 10.1109/CCCS.2019.8888092.
- [14] Y. Heydarzadeh, A. T. Haghghat, and N. Fazeli, "Utilizing skin mask and face organs detection for improving the Viola face detection method," *Proc. - UKSim 4th Eur. Model. Symp. Comput. Model. Simulation, EMS2010*, pp. 174–178, 2010, doi: 10.1109/EMS.2010.38.

