Face Mask Detection: An Application of Artificial Intelligence

Poonam Mittal[1], Ashlesha Gupta[1], Bhawani Sankar Panigrahi[2(✉)], Ruqqaiya Begum[3], and Sanjay Kumar Sen[4]

[1] Department of Computer Engineering, J C Bose University of Science and Technology, YMCA, Faridabad, India

[2] Department of CSE (AI & ML), Vardhaman College of Engineering (Autonomous), Hyderabad, India

bspanigrahi.cse@gmail.com

[3] Department of CSE, Vardhaman College of Engineering (Autonomous), Hyderabad, India

[4] Department of IT, Vardhaman College of Engineering (Autonomous), Hyderabad, India

Abstract. COVID-19 has been announced as a new pandemic which has affected almost all the countries of the world. Millions of people have become sick and thousands have died due to the respiratory illness caused by the virus. The virus is known to spread when small droplets from nose or mouth of an infected person gets dissolved in air when he or she coughs or exhales or when a person touches a surface infected with virus. The governments all over the world are working on ways to curb the spread of this virus. Multidisciplinary researchers are working to find the best solutions in their own way. Out of the many solutions wearing surgical facemasks is being one of the best preventive measures to limit the spread of corona virus. These masks support filtration of air and adequate breathability. But the problem is that few people don't use the masks regularly or occasionally due to various reasons like negligence and discomfort etc. This is one of the main causes of high spread of COVID. So, there is a strong need to detect people without mask at public places and to aware them. There are so many initiatives taken by government in this direction, but all have their limitation in one or the other way. So, there is a strong need of a digital solution to ensure that people comply with the government rules of wearing masks in public place sand to recognize unmasked faces on existing monitoring systems to maintain safety and security. Facial recognition systems were being used to identify faces using technology that includes hardware like video cameras. These systems work by combining AI based pattern recognition system along with biometrics to map facial features from an image and compare it with a database of known faces. This research content is also an initiative in this direction to optimize the results.

Keywords: COVID-19 · Face Mask · Detection Methods · Deep Learning · Image processing

S. Nandan Mohanty et al. (Eds.): ICISML 2022, LNICST 471, pp. 193–201, 2023.
https://doi.org/10.1007/978-3-031-35081-8_16

1 Introduction

Face Mask Detection is a technique which is used to classify that a person is wearing mask or not. Face mask detection can give us insights into the future of the current situation [1–5]. The Face mask detection system [6–8] can be used at Airports, Hospitals, Offices, Workspaces and Public places signal airport authorities to act against the travelers without masks. Face Mask detection techniques mainly follow the following steps:

i. Detection of people passing through camera.
ii. Identify mask on the face.
iii. Providing reliable statistics about whether the person wearing or not wearing the mask.

A variety of face detection systems have been proposed: A real time face mask detection technique was proposed by Harshil Patel which was implemented using Python and OpenCV. The technique executes by applying facial landmark feature to detect face images in the existing video images. Kera [13] and TensorFlow [14] based deep learning techniques are used to categorize the input images into mask or no-mask categories. Since the technique utilizes artificially generated images of the people wearing masks this can't be equivalent to the real images of people wearing masks. So, training model on the Real images can lead to the higher accuracy in detecting the people with the mask and without the mask.

This technique is based on PyTorch and OpenC, used to identify person without mask on image/video-stream. The data is augmented using PyTorch torch vision. The augmented data is then pre-processed using PyTorch transforms. The pre-processed data is then used to train the classifier (MobileNetV2) [7, 12, 13, 15, 18] using PyTorch. The trained classifier is then used to divide video stream images into mask and no-mask categories. The advantage of using PyTorch over TensorFlow is that it provides data parallelism.

Convolution Neural Networks based Image Classification technique has been preferred as it is lightweight and efficient mobile oriented model. The accuracy of the proposed model is 60–70%. It worksin2 phases:

i. **Object detection**–Neural network SSD issued here for object detection
ii. **Classification**-MobileNetV2 [12–14] issued here for classification purpose.

This technique also suffers from the limitation like small and blurred image, varying angle of face. To deal the issue of lack of clarity of the images in still frames this method was proposed, where different frames of the video are used for decision making. Additional techniques called as tracking is required to deal the same. All the frames related to a common person are grouped and classifier runs on each frame and then combines the results into a single decision.

In this technique a heuristic approach is used for posing estimation where key-points for many body parts, including the head are extracted. Using head key-point seems more intuitive and simpler. There are some weird glitches pose detections at some frames. These can be filtered out using a threshold for the detection scores. In the cases, where stable detection for all head key-points is not possible, chest key-points seems like the best

approach for street cameras in general. Here, pose estimation and extracting heads are used as object detection instead of using an SSD neural network. Now the classification to detect mask/no mask classification is done on the head images. Methods used to classify the images are:

i. **Threshold:** Images are classified on a scale of 100 where classifier outputs with score under20 (*no mask*) or above 80 (*mask*). Others are discarded because the existing solutions ignore the case of misplaced mask.
ii. **Multiple snapshots of video stream:** For each person, count the snapshot which achieves threshold of mask and similarly for non-mask as they walk their path.

Few refinements were also proposed to improve the reliability of the solution. Like margins may be used to refine the image and it darken the outside area. A configurable parameter to classify frontal face may be used to classify only faces so that both eyes can be identified.

2 Problem Identification

Need is the key for innovation. Innovation in real world video analytics is at its infant stage. Issues in real world analytics are constantly arising and researchers are trying to get the solution. There is a strong need to find the solutions and to raise the accuracy of existing solutions. Most of the existing solutions work very slow which is a big concern.

i. **Updating the model of pose estimation**: With the advancement of system this is an open area of research which model to use to get better pose estimation.
ii. **Classifier Improvement**: Mask and without mask are successfully implemented but it is very difficult to detect misplaced mask like nose and mouth are visible. Available datasets are ignorant towards this category.
iii. **Person re-identification**: Re-identification techniques are required if the screen freezes. Current video analysts have no solution, but continuous steps are being taken to improve the solution.

3 Working of Face Mask Detection

Face mask detection system combines AI techniques with deep learning algorithms along with image processing capabilities. It can be connected to existing surveillance systems to capture face images. The face images are then processed and fed as input to image classifier that uses deep learning models to categorize images into masked and no-mask images. The system can then issue alarm or send warning signals to the officials if someone tries to enter public places without mask.

Convolution Neural Networks are used here to implement Face Mask Detection System as CNN approach models the local understanding of the image properly. Very few parameters are used repeatedly in comparison to a fully connected network. While a fully connected network generates weights from each pixel on the image, a convolution neural network generates just enough weights to scan a small area of the image at any given time. A Convolution neural network (CNN) [8, 14, 15] is mainly used for image processing, classification, segmentation and for other auto correlated data. Each

convolution layer contains a series of filters known as convolution kernels. Pixel values are provided as input to the filters where it is a matrix of integers. Each pixel is multiplied by the corresponding value in the kernel, and then the result is summed up for a single value for simplicity representing a grid cell, like a pixel, in the output channel/feature map. To implement the model following steps are followed.

3.1 Gathering the Dataset

The first step before beginning implementation of the CNN model using MobileNetV2 is to gather the dataset for the face mask detection. This dataset of about *1,376* images will be used for training the model and also for testing its accuracy. The images with mask are generated artificially by using the facial landmarks.

3.2 Processing the Data

As Pandas [15, 18] library has its application for data manipulation and analysis, so here it is used to manipulate the images. Images are stored as large, multi-dimensional arrays (matrices) so here Numpy [17] library is used. These libraries are used for here for data pre-processing as desired for training of CNN model. The steps are as follows:

a. Image path variables extracted from dataset and stored separately in a image path list.
b. Pre-processing and Labelling–Pre-processing steps include resizing to 224 × 224 pixels, conversion to array format, and scaling the pixel intensities in the input image to the range [−1, 1] through pre-process input.
c. Splitting of training data and test data - We have taken the ratio of 80:20 for training and test data respectively. In machine learning, the "Features" are the descriptive attributes which are to be used to predict something and "Label" is the Output we are attempting to predict.
d. Images labelled as the Feature for training the model (X_train) and the class of mask and without mask as the Label (Y_train). Thes arrays X_train and Y_train are converted into Numpy arrays for better processing at later stages.

3.3 Classification Using MobileNetV2 Architecture

An improved version of MobileNetV1 is released by Google as MobileNetV2 (a neural network), which is optimized for mobile devices. MobileNetV2 delivers better results while keeping the mathematical overheads as low as possible. Mobile Nets are small, low-latency, low-power models parameterized to meet the resource on strain sofa variety of use cases. Mobile models with a spectrum of different model size can be successfully implemented through MobileNetV2. It is a very effective feature extractor for object detection and segmentation. MobileNetV2 is much faster and more accurate than MobileNetV1.

Model Architecture-The basic building block of MobileNetV2 architecture is a bottleneck depth separable convolution with residuals. The architecture of MobileNetV2 in Fig. 1 is comprised of the initial fully convolution layer having 32 filters and having 19 residual bottleneck layers. The architecture of MobileNet is tailored by various researchers as per different performance points. Input image resolution and width

multiplier as tunable hyper parameters are tailored to reach the desired precision and performance trade-off. The primary network (width multiplier 1, 224 × 224), has a computation cost of 300 million multiply-adds and uses 3.4 million parameters. The network computational cost ranges from 7 multiply-adds to 585 M MAdds, while the model size varies between 1.7 M and 6.9 M parameters.

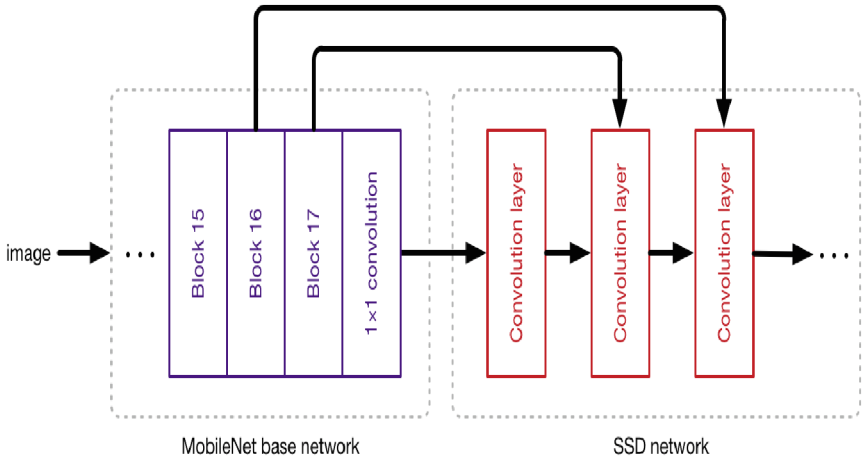

Fig. 1. Architecture of MobileNetV2

In comparison to MobileNetV1, MobileNetV2 is more accurate with reduced latency (Fig. 2).

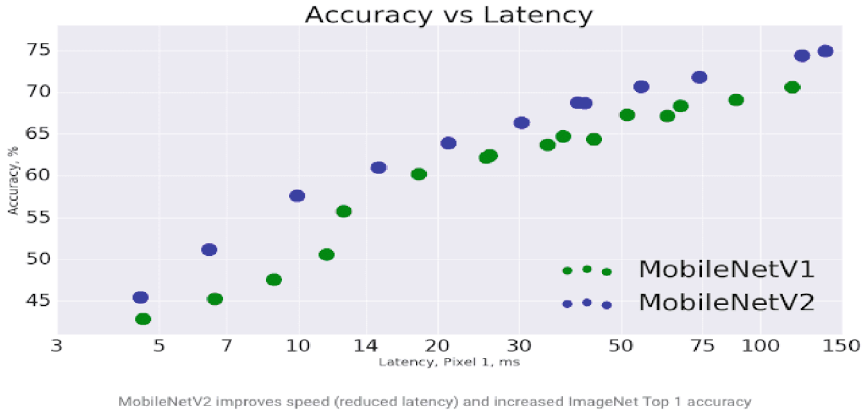

Fig. 2. Theoretical comparative view of MobileNetV1 andMobileNetV2 on Accuracy Vs Latency

V2 worked like feature extractor as MobileNetV1 but it requires 5 times fewer parameters and operations in terms of multiply-adds (Fig. 3).

Model	Params	Multiply-Adds	mIOU
MobileNetV1 + DeepLabV3	11.15M	14.25B	75.29%
MobileNetV2 + DeepLabV3	2.11M	2.75B	75.32%

Fig. 3. Comparison parameters of MobileNetV1 and MobileNetV2

Proposed model operations are best suited for mobile applications and allows memory-efficient inference. For the ImageNet dataset, MobileNetV2 works like a state of art for a wide range of performance points in ImageNet dataset. It works like an efficient mobile-oriented model that can be used as a base for many visual recognition tasks as in COCO datasets.

4 Implementation and Result Discussion

Training and testing of the data set is performed in the initial phase of implementation and then results are visualized and compared with the existing model stop predict the accuracy.

4.1 Training and Testing the CNN Model

After the model is implemented, it is trained using the pre-processed trained dataset and generated the predictions for the tested dataset. The CNN model is trained over 20 epochs and batch size of 32 inputs at a time. The Training takes around 40 min.

4.2 Making Predictions and Visualizing Results

After training and testing the model, results are visualized in the form of a graph which has set the comparison between the real and predicted results. The following sections include the screenshots of the code of the model and the Dataset visualization, followed by the results. The training process takes around so much time to complete, the time is based on how many Epochs (Forward Pass and Backward Pass) are to be done and for how many days at a time the weights in the model are to be updated (i.e. the Batch Size).

4.3 Training Loss and Accuracy on Dataset

Data available in dataset is trained and analyzed for both MobileNet V1 and MobileNetV2 and its comparison is shown in Fig. 4 and model accuracy and comparison is shown in Fig. 5.

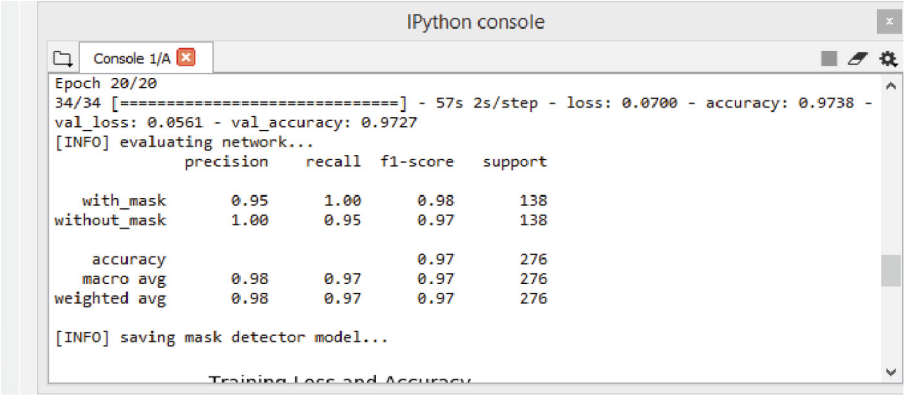

Fig. 4. Accuracy comparison of MobileNet V1 and MobileNet V2

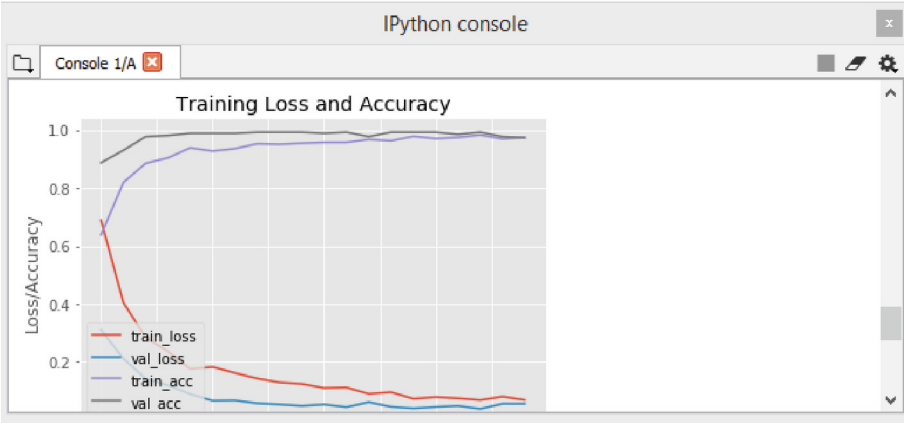

Fig. 5. Training Loss and accuracy of MobileNetV1 and MobileNet V2

4.4 Predictions and Analyzing Accuracy

Once the Model has been trained, we make the predictions by passing the input as frames of the video to the model. The Test Set contains images for classifying the class. The Predictions are made for images. These are then compared with the Actual class (Fig. 6).

Fig. 6. Accuracy of mask detection

5 Conclusion and Future Enhancements

Face Mask Detection System is successfully implemented using the Convolutional Neural Network as it automatically detects the important features without any human supervision. An embedded device with limited computational capacity finds the best tuned model asMobileNetV2. This technique can also solve the problem of detecting the people without mask in the public places to control COVID 19 spread. System is trained with approximate 1376 AI generated images. This can also be increased for better accuracy. As artificially generated images cannot be equivalent to the real images of people so training model on the real images can lead to the higher accuracy. As the proposed method suffers from few limitations which is the future direction in the same field as: CNN based classifiers are slower but more accurate than other machine learning classifier algorithms. Also, for further improvement, adding more layers to the model can improve accuracy.

References

1. Mangla, M., Sharma, N.: Fuzzy modelling of clinical and epidemiological factors for COVID-19 (2020)
2. Du, R.-H., et al.: Predictors of mortality for patients with COVID-19 pneumonia caused by SARS-CoV-2: a prospective cohort study. Eur. Respir. J. **55**(5) (2020)
3. Vincent, J.-L., Taccone, F.S.: Understanding pathways to death in patients with COVID-19. Lancet Respir. Med. **8**(5), 430–432 (2020)
4. Ignatius, T.S.Y., et al.: Evidence of airborne transmission of the severe acute respiratory syndrome virus. New England J. Med. **350**(17), 1731–1739 (2004)
5. Tellier, R.: Review of aerosol transmission of influenza a virus. Emerg. Infect. Dis. **12**(11), 1657 (2006)
6. Kazemi, V., Sullivan, J.: One millisecond face alignment with an ensemble of regression trees. IN: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2014)
7. Fernández-Delgado, M., et al.: Do we need hundreds of classifiers to solve real world classification problems? J. Mach. Learn. Res. **15**(1), 3133–3181 (2014)
8. Karpathy, A., et al.: Large-scale video classification with convolutional neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2014)

9. He, K., et al.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2016)
10. An Introduction to Convolutional Neural Networks. https://towardsdatascience.com/an-introduction-to-convolutional-neural-networks-eb0b60b58fd7
11. Simple Introduction to Convolutional Neural Networks. https://towardsdatascience.com/simple-introduction-to-convolutional-neural-networks-cdf8d3077bac
12. Dey, S.K., Howlader, A., Deb, C.: MobileNet mask: a multi-phase face mask detection model to prevent person-to-person transmission of SARS-CoV-2. In: Kaiser, M.S., Bandyopadhyay, A., Mahmud, M., Ray, K. (eds.) Proceedings of International Conference on Trends in Computational and Cognitive Engineering. AISC, vol. 1309, pp. 603–613. Springer, Singapore (2021). https://doi.org/10.1007/978-981-33-4673-4_49
13. Venkateswarlu, I.B., Kakarla, J., Prakash, S.: Face mask detection using MobileNet and global pooling block. In: 2020 IEEE 4th Conference on Information & Communication Technology (CICT). IEEE (2020)
14. Vu, H.N., Nguyen, M.H., Pham, C.: Masked face recognition with convolutional neural networks and local binary patterns. Appl. Intell. **52**(5), 5497–5512 (2021). https://doi.org/10.1007/s10489-021-02728-1
15. Li, C., Cao, J., Zhang, X.: Robust deep learning method to detect face masks. In: Proceedings of the 2nd International Conference on Artificial Intelligence and Advanced Manufacture (2020)
16. Rahman, M.M., et al.: An automated system to limit COVID-19 using facial mask detection in smart city network. In: 2020 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS). IEEE (2020)
17. Kumar, A., Kaur, A., Kumar, M.: Face detection techniques: a review. Artif. Intell. Rev. **52**(2), 927–948 (2018). https://doi.org/10.1007/s10462-018-9650-2
18. Lee, D.-H., Chen, K.-L., Liou, K.-H., Liu, C.-L., Liu, J.-L.: Deep learning and control algorithms of direct perception for autonomous driving. Appl. Intell. **51**(1), 237–247 (2020). https://doi.org/10.1007/s10489-020-01827-9
19. Helaly, R., et al.: Deep convolution neural network implementation for emotion recognition system. In: 2020 20th International Conference on Sciences and Techniques of Automatic Control and Computer Engineering (STA). IEEE (2020)
20. Mangla, M., Sayyad, A., Mohanty, S.N.: An AI and computer vision-based face mask recognition and detection system. In: 2021 Second International Conference on Secure Cyber Computing and Communication (ICSCCC). Organized by NIT Jalandhar, Punjab, India, 21–23 May 2021. https://ieeexplore.ieee.org/document/9478175