Face Mask Detection using Machine Learning

Aditya Chichghare[1], Nihar Kanse[2], Nikhil Pathak[3], Mayur Madhekar[4]
[1,2,3,4] *Department of Information Technology, Government College of Engineering, Amravati - 444601, Maharashtra, India*

Abstract - **A face mask detection software reduces many efforts in the tasks which can be automated. A visual input is taken from a particular area using camera. Then using the predefined dataset, the visuals are compared to identify whether face mask is present or not. Once we have the visual form of the input, it can be given as an input to the deep learned model to predict the action or output that is supposed to be served to the user. The model is needed to have patterns in which it can distinguish between masked and non-masked people. The model will also consist of the actions to be taken, once it determines the presence of face mask. Also, a basic thing that this software requires is visual. For that, any camera like CCTV, wireless webcams, etc. can be used to capture visuals. Various python libraries like tensor flow, numpy, sklearn are used for implementation of machine learning. UI will consist of a window which will pop up on user's command and it will display the visuals given by the camera to identify the face mask and then it will display the output accordingly whether face mask is present or not. Face mask detection software has many wide ranges of applications in this pandemic era.**

Index Terms - **OpenCV, Machine Learning, MobileNetV2, Keras, Haar Cascade.**

I.INTRODUCTION

Covid-19 has created a whole new frequency, and people realize that it is entering a new world. Although our society is currently changing rapidly, we must respond quickly to the new needs surrounding all of us. A risk-free environment will be everyone's top priority to make life more meaningful than before. We need to find ways to protect those who return to work and keep ourselves and our loved ones out of trouble. New plans are made every day according to guidelines and rules. As masks become a new standard in daily life, it is necessary to be vigilant at all times to create a safe environment conducive to public safety. For security reasons, many areas of society seem to be using some covid tracking tools. One of the most important tools is the mask detector. Using this system, you can check who does not have the mask you need.

These systems, in combination with existing surveillance systems, use innovative neural network algorithms to check whether a person is wearing a mask. In this chapter, we will briefly discuss artificial intelligence and its subgroups, namely machine learning and deep learning, and deep learning. Following the framework is a simple implementation of the mask detection system.

This project "Face mask detection using machine learning" which is based on machine learning concepts, aims at providing service of identifying whether a person is wearing mask or not. In this project we analyse the image by using CNN model.

II. RELATED WORK

So far, there are a lot of research papers related object detection and face recognition. For example, in A. Rosebrock, 2020. COVID-19: Face Mask Detector with OpenCV, Keras/TensorFlow, And Deep Learning – Pyimagesearch, it has been explained that how to implement face mask detection with the help of deep learning and open source machine learning libraries such as Keras and TensorFlow. Also, the article discusses about the dataset that has been used to train the face mask detector model. Python script has been used to train the the model with open-source libraries like Keras and Tensorflow.

Also, in D. Dwivedi, "Data Science, Artificial Intelligence, Deep Learning, Computer Vision, Machine Learning, Data Visualization and Coffee", it has been mentioned that face recognition would gain significant importance and promising applications in the upcoming years. Face detection is the fundamental part of face recognition process. In recent times, a lot of research and study work has done to make the face detection more advance enhance the accuracy of the prediction. It has wide applications across many fields such as law-enforcement, entertainment, safety, etc.

The Object Detection Algorithm uses edge or line detection features proposed by Viola and Jones in their research paper "Rapid Object Detection using a Boosted Cascade of Simple Features" published in 2001. The algorithm is given a lot of positive images consisting of faces, and a lot of negative images not consisting of any face to train on them.

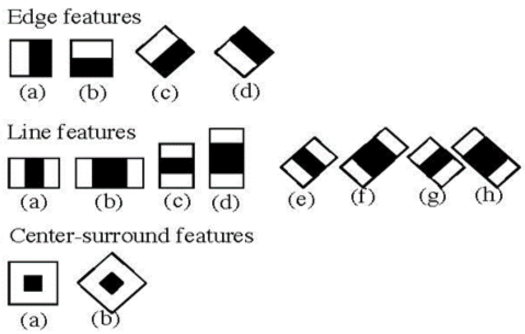

Fig. 1 Haar Features

In the Viola — Jones research, an algorithm is used for finding the location of the human faces in a frame or image. All human faces share some universal properties like the eyes region is darker than its neighbour pixels and nose region is brighter than eye region.

Haar algorithm is also used to select features or to extract features from objects in an image, with the support of edge detection, line detection, center detection for detecting eyes nose, mouth, etc. on the picture. It is used to select the essential features in an image and extract these features for face detection.

III. METHODOLOGY

Dataset

The data set used in the project consists of 3835 images, including 1916 faces with masks and 1919 faces without masks. All the images are actual images taken from Kaggle datasets and RMFD (Real-World Masked Face Dataset). The images in the dataset covers diverse races. The proportion of masked to unmasked faces determine that the dataset is balanced. In our approach, we have dedicated 80% of the dataset as the training data and the remaining 20% as the testing data, which makes the split ratio as 0.8:0.2 of train to test set.

Architecture

The algorithm consists of two parts: first it detects the presence of multiple faces in a certain image or video sequence, and then in the second part it detects the presence or absence of a face mask on the face. To detect face we have used OpenCV library. The latest OpenCV includes a Deep Neural Network (DNN) module equipped with a pre-trained Convolutional Neural Network (CNN) for facial recognition. The new model improves face recognition performance compared to traditional models.

The faces, we need to evaluate the bounding box around it and send it to the second part of the model to check whether the face has a mask or not.

The second part of the model is trained with the labelled dataset. We used Keras in conjunction with TensorFlow to train our model. The first part of the training includes storing all image labels in a NumPy array and the corresponding images are also reshaped for the model base (224, 244, 3) Before inputting, we perform the following random image augmentation, Rotations up to 20 degrees , zoom in and zoom out up to 15%, offset width or height up to 20%, cutting angle up to 15 degrees counter clockwise, flip inputs horizontally, and points outside the boundaries of the inputs are padded from the closest available pixel of the input. The basic model used here is MobileNetV2 with the specified 'ImageNet' weights. ImageNet is an image database that has been trained on hundreds of thousands of images and therefore helps a lot with image classification. For the base model, we cut off the head and used a number of our self-defined layers. We used an average grouping layer, a flat layer, a dense layer with exit shape (None, 128) and ReLU activation, a 50% demolition layer for optimization, finally another dense layer with exit shape (None, 2) and Softmax activation is used. The flow diagram of the algorithm process is shown below.

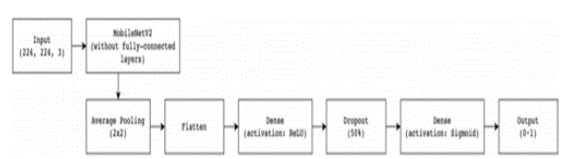

Fig. 2 Process Flow Diagram of the Model

C. Loss Function

The loss of the general detection problem can be broken down into two main subgroups: localization and confidential loss. The localization is the difference between the default predicted bounding box and the terrain truth bounding box (g). For a given center (cx, cy) we try to change the width and height of the box to reduce the loss. Confidence loss is the just measure of how high the probability of the presence of an object is, when there exists an object. Loss is the measure of how much a predicted box differs in its dimensions from the ground truth box.

Respectively, the equations for the localization and confidence losses can be defined as:

$$L_{loc}(x,l,g) = \sum_{i \in Pos}^{N} \sum_{m \in \{cx,cy,w,h\}} x_{ij}^{k} \text{smooth}_{L1}(l_i^m - \hat{g}_j^m)$$

$$\hat{g}_j^{cx} = (g_j^{cx} - d_i^{cx})/d_i^w \qquad \hat{g}_j^{cy} = (g_j^{cy} - d_i^{cy})/d_i^h$$

$$\hat{g}_j^{w} = \log\left(\frac{g_j^w}{d_i^w}\right) \qquad \hat{g}_j^{h} = \log\left(\frac{g_j^h}{d_i^h}\right)$$

and

$$L_{locf}(x,c) = -\sum_{i \in Pos}^{N} x_{ij}^{p} log(\hat{c}_i^p) - \sum_{i \in Neg} log(\hat{c}_i^0) \quad \text{where} \quad \hat{c}_i^p = \frac{\exp(c_i^p)}{\sum_p \exp(c_i^p)}$$

The notations used in the equations are as follows:

- g: ground truth bounding box
- l: predicted box
- d: default bounding box
- x_ij^p : matching i^th predicted box with j^th default box with “p” category.
- cx, cy: distance from centre of box in both x and y direction
- w, h = width and height with respect to image size
- c: confidence of presence of object or not.

D. Methods

Convolutional Neural Network (CNN):

Convolutional neural network (CNN or ConvNet) is a deep neural network most commonly used to analyse visual images. They are also known as change-invariant or spatially invariant artificial neural networks (SIANN) based on weight-sharing architecture of convolution kernels that slides on the input features and provide an equivalent translation response called a feature maps. Counter-intuitively, most convolutional neural networks are only equivariant, as opposed to invariant, to translation. They are used in image and video recognition, recommender systems, image classification, image segmentation, medical image analysis, natural language processing, brain-computer, financial time series and interfaces

MobileNetV2:

MobileNetV2 is a convolutional neural network architecture designed to work well on mobile devices. It is based on the inverted residual structure, in which residual keys are found between the bottleneck layers. The middle expansion layer uses the light depth wish convolutions to filter out features that are non-linear. Overall, the MobileNetV2 architecture contains an initial common convolutional layer with 32 filters, followed by 19 remaining bottleneck layers.

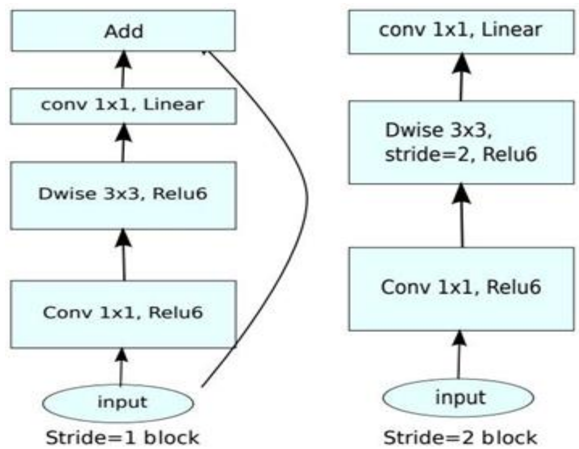

Fig. 3 Convolutional Blocks of MobileNetV2.

IV.RESULTS

- The accuracy of the model is 96%. It can also be observed in the accuracy loss graph obtained below.
- The loss value is below 0.1.
- It can be observed that with every iteration of the model training, the loss is getting gradually decreased and the accuracy is getting increased.

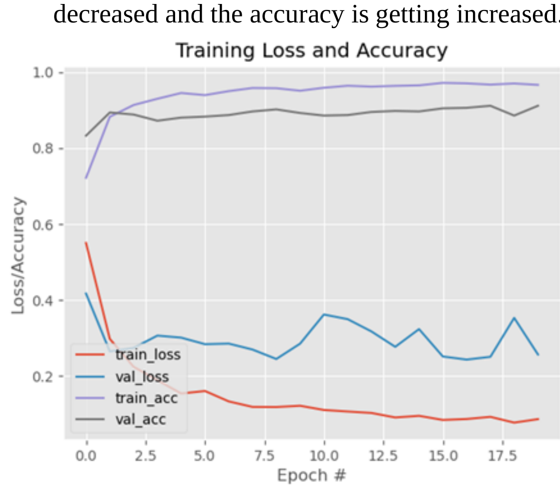

Fig. 4 Training Loss and Accuracy Plot

The output is tested on a video stream. A face with mask on is indicated in yellow color rectangle drawn across the face with the probability of it being correct. While a face without a mask is also indicated with a rectangle drawn across it in red color, along with the probability of it being correct.

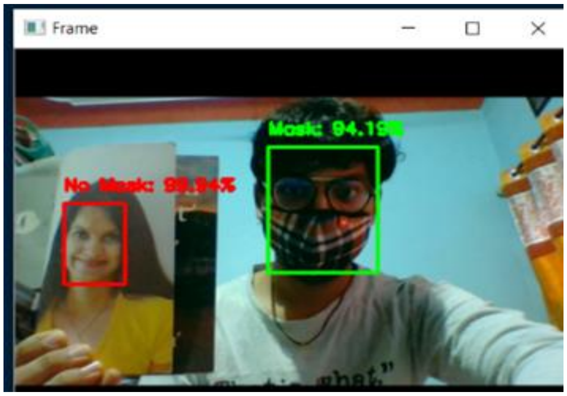

Fig. 5 Output Obtained

V.CONCLUSION

To mitigate the spread of COVID-19 pandemic, measures must be taken. We have modelled a face mask detector using a convolutional neural network. To train, validate and test the model, we used the data set, which consists of 1916 images of masked faces and exposed in 1919. These images come from various sources such as Kaggle and RMFD datasets. The model was derived from images and live video transmissions. To select a base model, we evaluated metrics such as accuracy, precision, and recovery and selected the MobileNetV2 architecture with the best performance with 96% accuracy and 99% recall. In addition, it is computationally efficient with MobileNetV2, which makes it easier to install the model in embedded systems. This face mask detector can be used in many areas such as shopping malls, airports and other high-traffic locations to keep an eye on the public and prevent the spread of disease by checking who is and who is not following the basic rules.

VI.ACKNOWLEDGMENT

Prof. D. R. Uike, Dept. of Information Technology (I.T.), Government College of Engineering, Amravati, has been a source of support and guidance to the authors throughout this research.

We are also obliged to our Head of Department (Dept. of Information Technology), Prof. A. W. Bhade for giving us this great opportunity. We are grateful for their cooperation during the period of our assignment.

REFERENCES

[1] V. Bhat, A. Shambavi, Komal Mainalli, K M Manushree, Shraddha V Lakamapur, “Review on Literature Survey on Human Recognition with face Mask”, 2021.

[2] D. Dwivedi, “Data Science, Artificial Intelligence, Deep Learning, Computer Vision, Machine Learning, Data Visualization and Coffee”.

[3] R. Suganthalakshmi, A. Hafeeza, P. Abinaya, A.Ganga Devi, “Covid-19 Facemask Detection with Deep Learning and Computer Vision”, 2021.

[4] Rosebrock A., 2020. “COVID-19: Face Mask Detector with OpenCV, Keras/TensorFlow, And Deep Learning” - Pyimagesearch.

[5] C. Lippert, B. Bergner, A. Ahmed, R. Ali, S. Adel, H. Shahriar, S. Mojumder, 2020. “Face Mask Recognition” – a technical report.

[6] R. Sugantha lakshmi, A. Hafeeza, P. Abinaya, A. Ganga Devi, “Covid-19 Facemask Detection with Deep Learning and Computer Vision” in IJERT, 2021.

[7] P. Viola and M. Jones, “Rapid object detection using a boosted cascade of simple features,” in Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition. CVPR 2001, vol. 1. IEEE, 2001, pp. I–I.