Accurate Detection Of Face Masks Using Deep Learning

Shilpa Hariraj, Mohammed Furqaan Khan, Akshay Mohan Revankar, Vishwanath M Shantpurmath, Mohit vats

Abstract— The COVID-19 chaos has raised a range of challenges, there arises a severe want of protection mechanisms, mask being the first one one. The primary aim of the project is to find the presence of a mask on faces on a video stream like a webcam as well as on images. We use deep learning to work with our model. The design used for the object detection purpose is Single Shot Detector (SSD)owing to its sensible performance accuracy and high Speed images. Since the detection had to be real time, SSD with Mobilenet was used to train the model. The SSD-Mobilenet model provides the simplest accuracy and speed compensation which are applicable to all the object detection models. The mask detector uses opencv that uses a digital camera to show the detection and accuracy of the mask. The project presently detects face masks and shows it's accuracy in real time.

Index Terms— convolutional, detection, face, labelimg, masks, object

1 Introduction

2020 and 2021 have shown human kind some mind fixing series of events and covid nineteen has been somewhat life changing. It affected the health and lives of masses, covid 19 incorporated measures that required to be taken strictly. From basic hygiene standards to the treatments within the hospitals, individuals are attempting to be as safe as possible. Face masks are a really essential protecting equipment. individuals wear masks after they exit their homes. a technique ought to be developed to see if someone is wearing a face mask or not. The target of the project is to make a detector that acknowledges each single form of mask and shows the output on a screen. During this project, we are going to be developing a face mask detector that's able to distinguish between faces with masks and faces with no masks. During this report, we've planned a detector that employs SSD for face recognition and a neural network to detect presence of a face mask. The implementation of the algorithmic rule is on images, videos and live video streams.The start to acknowledge the presence of a mask on the face is to detect the face and also the second step is to point out the output on a screen. Face detection is one in all the applications of object detection and may be used in several areas like security, biometrics, enforcement and more

2 Data Preparation

Detailed The dataset was collected by downloading various photos of masks from the net, our priority was to download masks that looked different in both shape and color so that we could get the best results. Labelling is a fairly simple tool we used. We drew bounding boxes over the individual images and labelled each mask. LabelImg is a notation tool. It uses QT and Python to function. Once the boxes have been drawn over the images, they are saved as XML files.

The labeled image and the xml file is as follow:

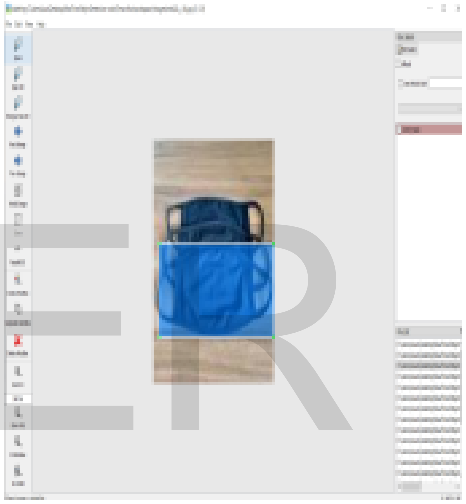

Figure 1 - Labelling of masks using Labelmg

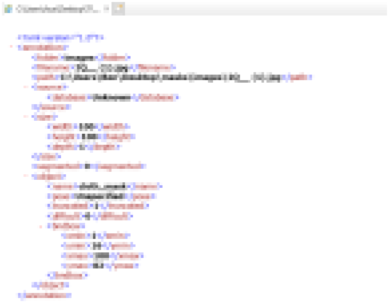

Figure 2 - Xml file of masks

3 Software Requirement

In the following, we present a model of face mask recognition based on computer-aided vision and deep learning. The proposed model can be incorporated into surveillance cameras to prevent the transmission of COVID19 by allowing the detection of people wearing a mask without a mask. between deep learning and classic machine learning techniques with Opencv, TensorFlow and Keras. We achieve maximum precision and spend the least time in the training and recognition process. We can run this project through the Anaconda application. The main requirement to implement this The project uses the Python programming language together with deep learning, machine learning, computer vision and also Python libraries, the architecture consists of Mobile Net as the backbone, it can be used for high and low computing scenarios on the CNN used algorithm in our proposed system. The system must have an undistorted data record 'with_mask'. The data record must contain more than 1500 images in the classes "with_mask". The data set should not reuse the same images in the training and testing phases. The system should be able to load the mask classifier model correctly. The system must be able to recognize faces in images or video sequences. The system must be able to extract each of them.Face Region of Interest (ROI) - For face recognition, and therefore face mask recognition, to be successful, no object should be between the system and the user's face. The final position of the face should fit within the frame of the webcam and should be closer to the camera. Correctly recognize skins in the formats 'png', 'jpg', 'jpeg' and 'gif'. The system must be able to recognize face masks on human faces in every live video frame. the probability should be displayed together with the output of "Mask".

4 Literature Review

In the field of image processing and computer vision the most trending topic is object detection. The use of object detection ranges from large scale applications to small scale industrial applications. The technique of object detection is used in a lot of industries. Some of these examples include the object detection in your phone's front camera for face recognition, used in security systems, medical imaging, self-driving cars, tracking and counting people and a plethora of tasks.

We have decided to detect masks because of the current pandemic hoping it will contribute to a system that helps people and potentially saves lives.

In this project we will be using tensorflow which is an open source platform for machine learning, we will then find multiple images of masks and download them, we. We then install label-img to create our labels, we generate csv files and we create tf record files. The categorization process begins after reading the image. The location of these objects is also specified by a boundary called a bounding box. Deep learning helps us to learn about features in a manner that’s end to end.

In this project we will be applying deep learning techniques and training CNN based models to detect the masks. The labelled images were given to a SSD-mobilnet model for training. SSD with mobilnet was used for training as it provides the best accuracy and speed trade off within the fastest detectors. SSD: Single shot multibox detector SSD only needs an input image and ground boxes for each object during training. We then evaluate the default boxes in a convoluted method, the evaluation happens at different locations using different scales. For each of these default boxes we predict the shape, the offsets and the confidences for the truth boxes.

By using the free GPU’s on google we will be training the model. Since the detection had to be real time, the SSD MobileNet model provides the best accuracy and speed trade off with respect to all object detection models.

The ratio of the samples in train/validation while splitting the dataset was kept equal using the train test split function of sklearn. Moreover, to deal with unbalanced data, they passed this information to the loss function to avoid unproportioned step sizes of the optimizer. They did this by assigning a weight to each class, according to its representability in the dataset. They assigned more weight to classes with a small number of samples so that the network will be penalized more if it makes mistakes predicting the label of these classes. While classes with large numbers of samples, they assigned to them a smaller weight.

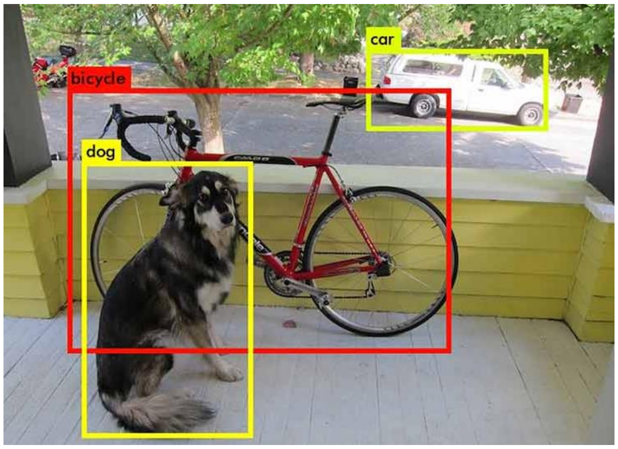

Figure 3 - Example of bounding boxes on objects

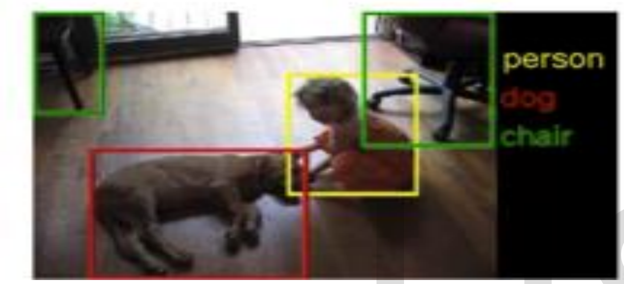

Figure 4 - Natural object classification

.

5 TRAINING

SSD with Mobilenet was used for training because it offers the best accuracy and speed compensation within the fastest detectors. SSD: Single Shot Multiple Box Detector The SSD only needs one input image and ground truth frames for each object during training. small set of standard boxes with different aspect ratios at each location on different feature maps with different scales (e.g. (b) and (c)). For each standard frame, we predict both form offsets and confidences for all object categories ((c1, c2, ···, cp)). At training time, we first match these standard squares with the truth squares on the ground. For example, we paired two standard boxes with the cat and one with the dog that are treated as positive and the rest as negative. The loss of the model is a weighted sum between the loss of location and the loss of confidence.

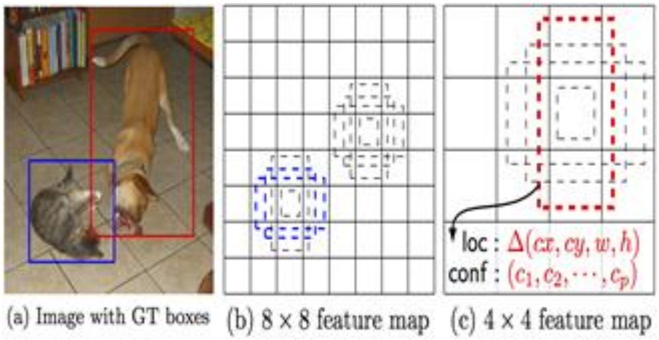

Figure 5 - Single shot detector

6 ARCHITECTURE

The overall objective loss function is a weighted sum of the localization loss and the confidence loss:

$$L(x,c,l,g) = \frac{1}{N}(L_{conf}(x,c) + \alpha L_{loc}(x,l,g))$$

The training and localization loss is defined as :

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

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

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

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

The MobileNet model is based on depth wise separable convolutions which is a form of factorized convolutions which factorizes a standard convolution into a depthwise convolution and a 1×1 convolution called a pointwise convolution. Standard convolutions have the computational cost of:

$$\mathbf{G}_{k,l,n} = \sum_{i,j,m} \mathbf{K}_{i,j,m,n} \cdot \mathbf{F}_{k+i-1,l+j-1,m}$$

$$D_K \cdot D_K \cdot M \cdot N \cdot D_F \cdot D_F$$

The computational effort of the number of input channels M, the size of the core Dk × Dk and the size of the feature map DF × DF is multiplied by turns in the depth have a computational effort of :

$$D_K \cdot D_K \cdot M \cdot D_F \cdot D_F$$

The deep fold is extremely efficient compared to the standard fold. However, it only filters the input channels, it does not combine them to create new functions. Therefore, an additional layer is needed that is a linear combination of the output of the convolution in depth by 1. computes × 1 convolution into it to generate these new properties.

$$\hat{\mathbf{G}}_{k,l,m} = \sum_{i,j} \hat{\mathbf{K}}_{i,j,m} \cdot \mathbf{F}_{k+i-1,l+j-1,m}$$

$$D_K \cdot D_K \cdot M \cdot D_F \cdot D_F + M \cdot N \cdot D_F \cdot D_F$$

By expressing convolution as a two step process of filtering and combining we get a reduction in computation of:

$$\frac{D_K \cdot D_K \cdot M \cdot D_F \cdot D_F + M \cdot N \cdot D_F \cdot D_F}{D_K \cdot D_K \cdot M \cdot N \cdot D_F \cdot D_F}$$
$$= \frac{1}{N} + \frac{1}{D_K^2}$$

The Architecture of the SSD as follows :

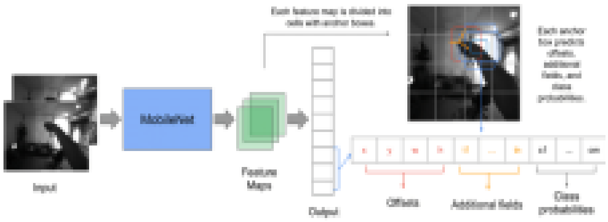
Input
MobileNet
Feature Maps
Each feature map is divided into cells with anchor boxes.
Output
Each anchor box predicts offsets, additional fields, and class probabilities.
Offsets
Additional fields
Class probabilities

Figure 5 - Architecture of SSD

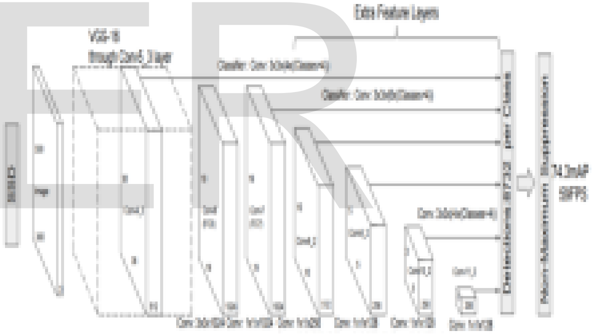

Figure 6 - SSD

Network structure of standard convolutions and Depthwise convolutions

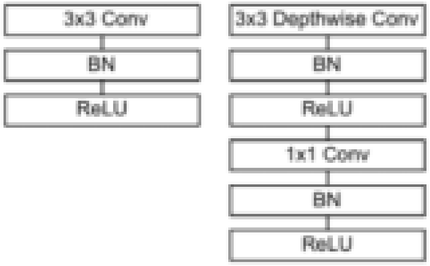

Figure 7 - Network Banter

Width Multipliers and Resolution Multipliers: The Role of the Width Multiplier & agr; consists in thinning a network evenly on each layer. The second hyperparameter for reducing the computational effort of a neural network is a resolution multiplier ρ

$$D_K \cdot D_K \cdot \alpha M \cdot \rho D_F \cdot \rho D_F + \alpha M \cdot \alpha N \cdot \rho D_F \cdot \rho D_F$$

7 Results

The model was able to detect the bounding box real time, it showed the percentage of the accuracy and worked seamlessly :

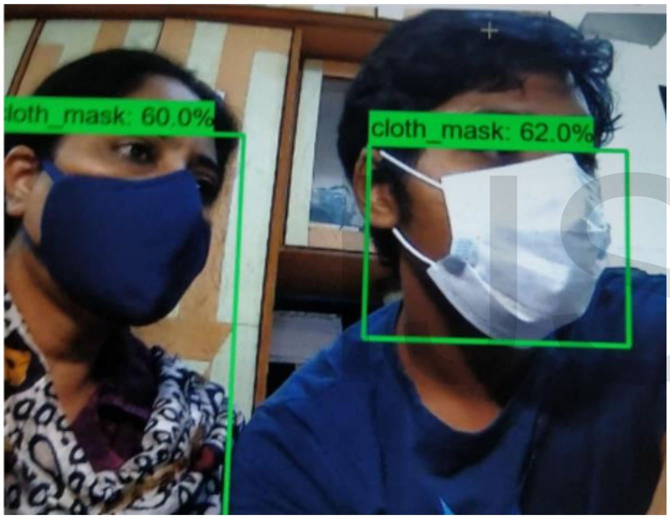

Figure 8 - Detection of masks on two faces

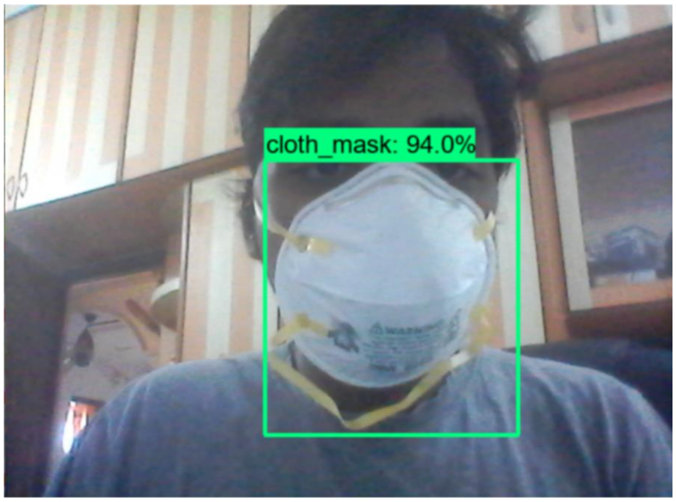

Figure 9 - Percentage shown above the box

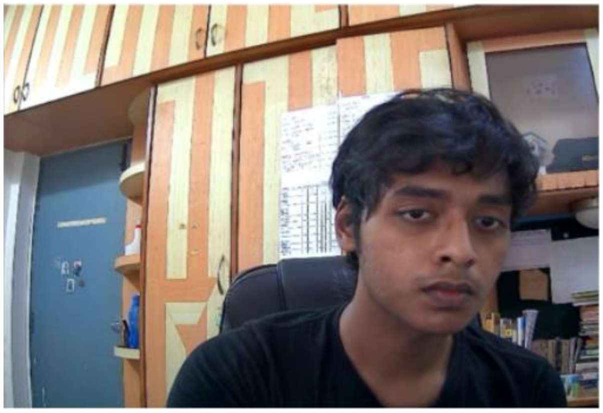

Figure 10 - No detection when subject is not wearing mask

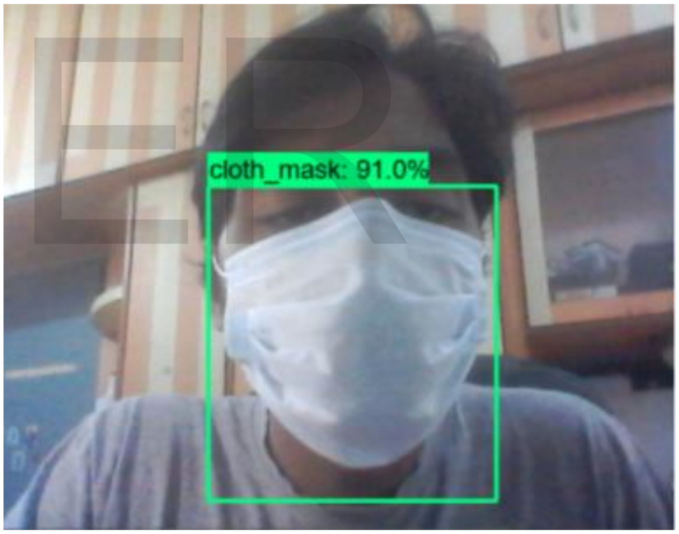

Figure 11 - Detection of masks at different face angles

8 Conclusion

In this paper we discuss the role of using CNN, deep learning and object detection to recognize face masks, the project required a system with a high gpu for training purposes. The major requirement for implementing this project is using python programming language along with Deep learning, Machine learning, Computer vision and also with python libraries. The architecture consists of Mobile Net as the backbone. The project ran smoothly and the detection of masks happened without any errors. The percentage above the bounding box signifies the accuracy of the detection of the mask, we plan to devise a detection technology that detects masks and no masks with a sound and a notification in the future.

REFERENCES

[1] T. Labelme.
[2] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, A. C. Berg, SSD: Single Shot MultiBox Detector, Lecture Notes in Computer Science 1 (2016) 21–37. doi:10.1007/978-3-319-46448-0_2
[3] A. A.-S. Selami, A. A. & Fadhil, A Study of the Effects of Gaussian Noise on Image Features, Kirkuk University Journal / Scientific Studies (2016)
[4] Stojnev, A. Ilić (2020).
[5] H.-T. & Choi, Lee, & Ho-Jun, Kang, & Hoon, Yu, & Sungwook, Park, Ho-Hyun (2021).
[6] Magalhães, Rafael & Peixoto, Helton. (2019). Object Recognition Using Convolutional Neural Networks. 10.5772/intechopen.89726.
[7] Richardson Santiago Teles de Menezes, Rafael Marrocos Magalhaes and Helton Maia (November 7th 2019). Object Recognition Using Convolutional Neural Networks, Recent Trends in Artificial Neural Networks - from Training to Prediction, Ali Sadollah and Carlos M. Travieso-Gonzalez, IntechOpen, DOI: 10.5772/intechopen.89726.