Face Mask Detection Using Several Deep Learning Methods

Tianhan Chen, Tianzi Luo, Xianghu Wang, Zhutian Liu
Department of Electrical Engineer & Computer Science,University of California, Riverside, CA
University of California, Riverside
<tchen298, tluo023, xwang473, zliu272>@ucr.edu
Student ID: 862325379, 862251424, 862318845, 862254970

Abstract

The COVID-19 pandemic has been a worldwide catastrophe. Its impact not only economically, but also socially safety and in terms of human health was unexpected. Wearing a medical mask is one of the prevention measures that can limit the spread of certain respiratory viral diseases, including COVID-19. Therefore, it is necessary to wear a mark properly at public places like supermarkets, shopping malls.

In this paper, we propose an intelligent face mask detection system with several methods to automatically detect when facemasks are being worn incorrectly in real-time scenarios. We want to use the end-to-end approach (YOLOV5 and Faster R-CNN) and split approach (face recognition and facemask detection) to detect not only if a mask is used or not, but also other errors that are usually not taken into account but that may contribute to the virus spreading.

Keywords: YOLOv5, Faster R-CNN, Facemask detection and recognition, Deep Learning, COVID19

1 Introduction

The coronavirus disease until June 2022 has already infected more than532 million people and caused over 6.3 million deaths globally according to the situation report of the World Health Organization(WHO). There are many serious large-scale respiratory diseases as well including Severe Acute Respiratory Syndrome(SARS) and Middle East Respiratory Syndrome(MERS). Therefore, people should be concerned about their health and respiratory diseases. There are studies conducted that wearing face masks can help to stop coronavirus from spreading [1] [2] [3] and the government made it necessary for all the people to wear face masks when going out in public places. Research studies show the effectiveness of N95 and surgical masks in preventing virus transmission are 91% and 68% respectively [4]. This is the reason that a facemask detection system is necessary to help people but there are very few researches related to face mask detection.

Deep neural networks (DNNs) as the main component of deep learning methods have everything it offers including object detection, image classification, and image segmentation [5] [6]. Convolutional neural networks (CNNs) is one of the principal models of DNN is generally used in computer vision tasks [7]. After training the model, CNNs can identify and classify facial images even with minor differences using their overwhelming feature extraction ability and store image pattern details.

Face mask detection is a system that detects whether a person is wearing a mask or not. It is the same as an object detection system in which a system detects a particular class of objects. Through building this system we are trying to help ensure people's health and safety in public places. To accomplish this task, we'll be fine-tuning the YOLOV5 [8] and Faster R-CNN architecture [9].

In this paper, we proposed another algorithm, the split approach. It is a two-phase system in which first we need to recognize the human faces in an image and cut out the face part. Then, for each human face image, we implement facemask detection. To achieve better results, we trained our model using a large dataset consisting of with, without and incorrectly-wear mask faces.

2 Related works

2.1 Object Detection

Object recognition is the inherent tasks that a computer vision (CV) technique. Object recognition encompasses both image classification and object detection [10]. The task of recognizing the mask over the face in the pubic area can be achieved by deploying an efficient object recognition algorithm through surveillance devices. The object recognition pipeline consists of generating the region proposals followed by classification of each proposal into related classes [11].

2.1.1 Single-stage detectors

Simple regression problem by taking the input image and learning the class probabilities and bounding box coordinates. OverFeat [12] and DeepMultiBox [13] were early examples. YOLO (You Only Look Once) popularized the single-stage approach by demonstrating real-time predictions and achieving remarkable detection speed but suffered from low localization accuracy when compared with two-stage detectors; especially when small objects are taken into consideration [8]. Besides, the RetinaNet [14] proposed by Lin is also a single-stage object detector that uses a featured image pyramid and focal loss to detect the dense objects in the image across multiple layers and achieves remarkable accuracy as well as speed comparable to two-stage detectors.

2.1.2 Two-stage detectors

In contrast to single-stage detectors, two-stage detectors follow a long line of reasoning in computer vision for the prediction and classification of region proposals. They first predict proposals in an image and then apply a classifier to these regions to classify potential detection. Spatial pyramid pooling SPPNet [15] (modifies R-CNN with an SPP layer) collects features from various region proposals and fed into a fully connected layer for classification. The capability of SPNN to compute feature maps of the entire image in a single-shot resulted in a significant improvement in object detection speed by the magnitude of nearly 20 folds greater than R-CNN. Next, Fast R-CNN is an extension over R-CNN [9] and SPPNet. It introduces a new layer named Region of Interest (RoI) pooling layer between shared convolutional layers to fine-tune the model. Moreover, it allows to simultaneously train a detector and regressor without altering the network configurations. Although Fast-R-CNN effectively integrates the benefits of R-CNN and SPPNet but still lacks detection speed compared to single-stage detectors [16].

2.2 Face recognition

In the recent work, there are three major approaches in Face recognition.

The first method is Anchor-base Face Detection. Anchor boxes are a set of predefined bounding boxes of a certain height and width. Using an Anchor box makes the evaluation of face prediction faster because we don't need to scan the whole image. Anchor box-based approach has a significant advantage at detect faces that vary in scales and various bounding box shapes.

The second method is Scale-invariant Face Detection. Those methods construct a different framework to detect the face in variant size, where the High-level feature is used to detect large faces and the low-level feature is used to detect small faces. Also, those methods help to integrate the high-level semantic feature into low-level layers with higher resolution.

The third method is Context-associated Face Detection. Those methods will using context information such as body contextual information to helps predicting the position of human faces. Using additional supervise labels for context information or anchor-level attention in the training process, those methods shows largely benefits the detection of finding small, blurred and occluded faces.

3 Methodology

3.1 Data setup

We used two individual datasets to train and test our neural network models. For YOLO and faster R-CNN, we used data which has 853 images belonging to the 3 classes, wear a mask, not wear a mask and wear the mask incorrectly, as well as their bounding boxes. We divided the training set and test set according to 4 to 1. For the split-step approach, we used another dataset which has 4559 images for the face recognition part.

In the Real-time detection, we captured face with a camera and output should be the bounding boxes and predicted labels.

3.2 Faster RCNN

Faster R-CNN is the state of art object detection framework developed by Microsoft researchers, based on the region-based CNN (R-CNN) variants. Here we mainly focus on the R-CNN [17]. It present itself as an end-to-end network and can identify the object and locate its border [9]. However, in a theory view it has three stages,

- Region proposal generator
- Feature extraction
- Classification

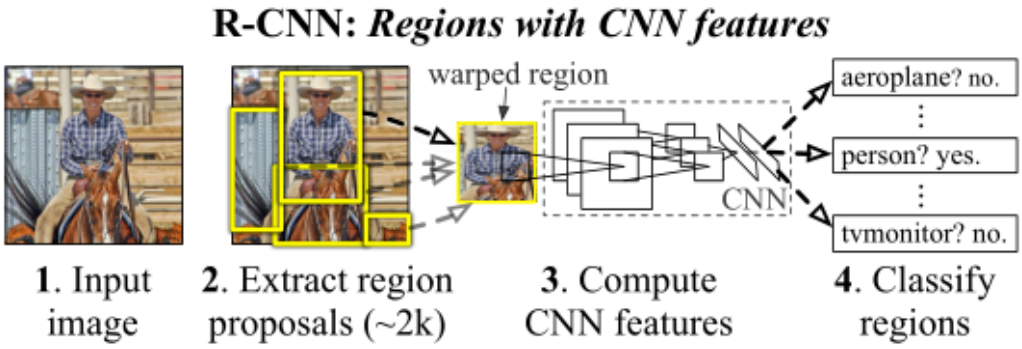

Figure 1: R-CNN architecture

Initially the R-CNN has strictly three stages pipeline as Fig. 1. The novelty is the usage of CNN for feature extraction on proposed regions. The multi-class classifier take the features and classify the region to either background or one of the object classes.

The later developed fast R-CNN approach (Fig. 2) builds a one staged network. It proposed ROI pooling, which takes all the region proposals and compute their feature vectors. Then they are fed to softmax function for object probability and a bounding box regressor.

The main benefit of such pooling layer is the speed. The ROI pooling layer computes all the features in a convolution network rather than computed them one by one.

The next generation model, faster R-CNN, improves the region generation method. Changed the time-consuming Selective Search algorithm into a fully convolutional network called region proposal network (RPN). Essentially, it implements the terminology of neural network with attention to tell the Fast R-CNN where to look.

3.3 YOLO V5

The YOLO [18] model follows a different path to object detection. Instead of building up from region proposal to classifier, it processes the whole image at the same time, which is where the name you

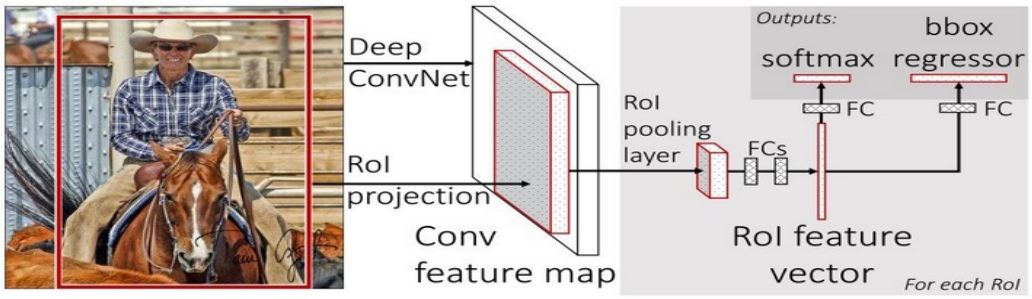

Figure 2: Faster RCNN architecture

only look once comes from. The 3 shows the basic architecture of yolo [8]. It consists basically the convolution network and 2 fully connected layers.

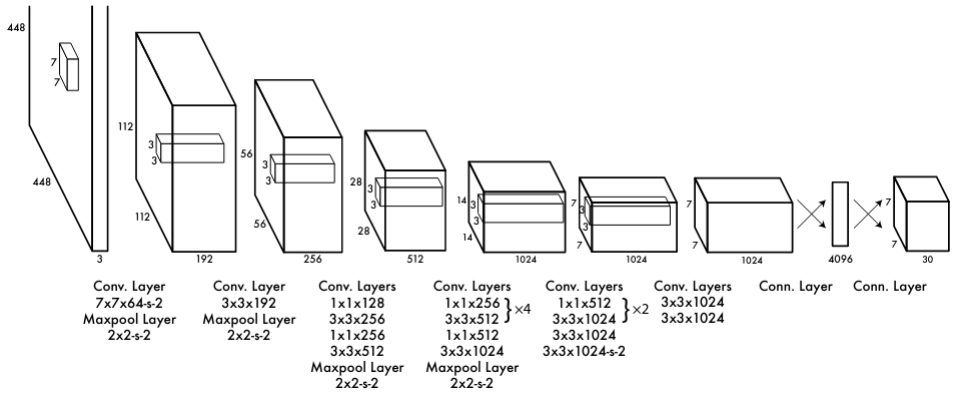

Figure 3: YOLOv3 architecture

The image is split into grids (fig 4) and each grid has confidence scores for bounding box and object class. Without the process of region proposal generation, the process is extremely fast. We implemented the real-time face mask classification on webcam and the speed performance is reasonable for practical use.

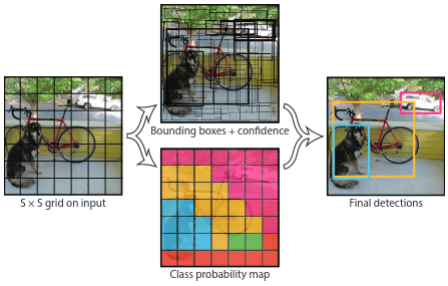

Figure 4: YOLO Grid approach

3.4 2-Step approach

Two step approach basically using face recognition and object classification algorithm together to get face location and musk wear condition classification. PyramidBox[19] and DenseNet[20] will be our choice for those 2 algorithm. The Fig. 5 shows the architecture of two step approach.

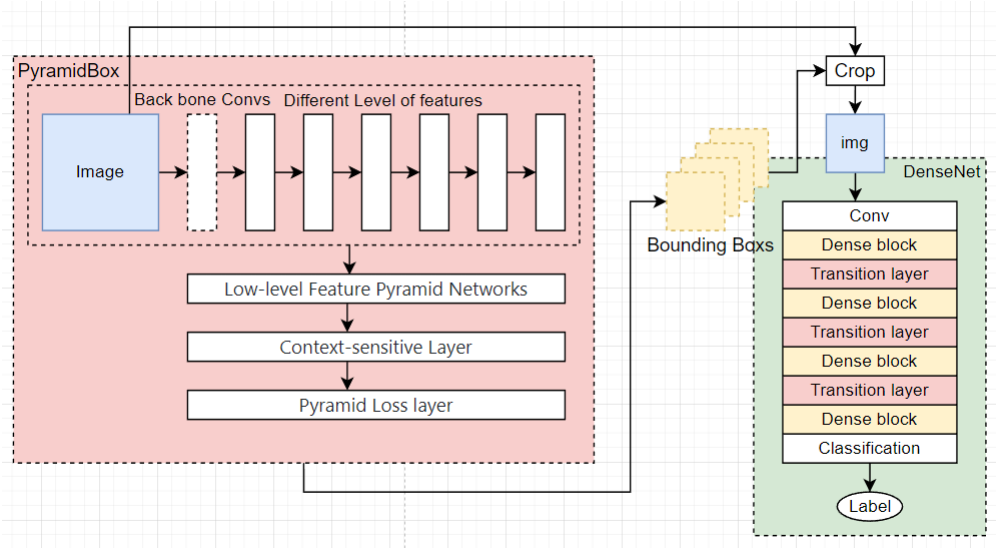

Figure 5: 2-step architecture

PyramidBox using VGG16 as its backbone structure, but changed its middle/Top layers in to more deeper convolution layers so that they can use its feature map to detect human face at different feature level. In the middle/top feature layer, its receptive field is big enough to capture the smallest face in a picture.

In the Low-level Feature Pyramid Network layer, they used Feature Pyramid network(FPN) to merge the high level feature map into low level layers. FPN is a popular network that used in a lot of Object detection methods. As we can see in Fig. 6, FPN takes both low level and high level feature as input, and up samples the high level feature to have the same size as the low level feature and finally do a element wise product. By doing that, low level feature can have higher resolution. PyramidBox did not apply FPN on top 3 layer because the reception field of those feature is too large to cover enough face texture and bring a lot of noise.

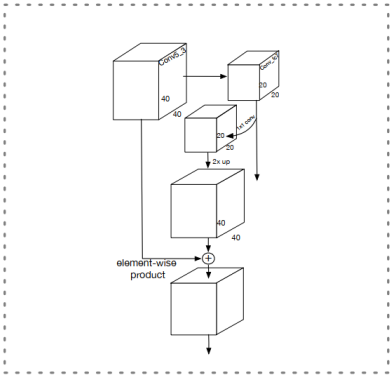

Figure 6: FPN architecture

In the context sensitive layer, for every anchor in the feature map, it will also predict not only the region of face but also head and body. This layer will compares the Iou of an anchor with its target region against a threshold to assign a label. The output of this layer will be the label and location for face, head and body. Based on the context information, PyramidBox can handle small, blurred, and partially occluded object better.

In our project, we used the pre-trained model for inferring the bounding box of the human faces. For the output of this network, we crop the original images into small images of faces using the prediction results.

DenseNet is a deep object classification network, which using dense connection in most of its convolution block. Dense connection performs channel concatenation for input of a convolution layer and its output. Using the dense connection the gradient can be easily propagated to low level layers more directly. This can prevent the network from overfitting.

In our project, we Using the DenseNet121 model as our second step network. We used the data sets that contains 4500 images of human face with a musk in 3 wear condition. We split the dataset into training, validation, and testing with the ratio 4:1:1. During the training process, we used Adam optimization with weight decay. As we can see in the Fig. 7, our model finally can reach 0.96 percent of accuracy in testing set.

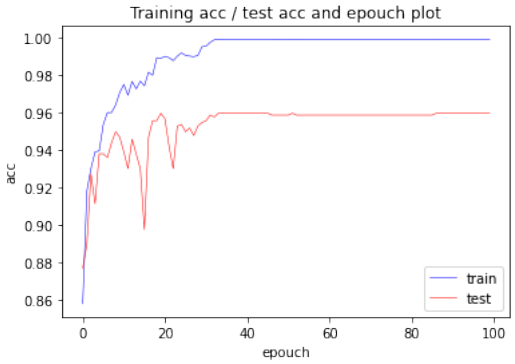

Figure 7: train/test accuracy

Combine with the bounding box and its label, we can output the full musk recognition result. It supposed to be more accurate on its prediction result because it must exits a human face before we can make a prediction on its label. Single musk photo without human face will not being recognized as a human with musk.

4 Experiments

4.1 Evaluation

Evaluating the object detection model is not simple, because each image can have many objects, and each object can belong to different categories. In this project it means that each image can have many face, and each face is belong to 3 different classes. This means that we need to measure whether the model finds all objects and verify whether the objects found belong to the correct class.This means that an object detection model needs to accomplish two things:

- Find all the objects in an image
- Check if the found objects belong to the correct class

In this project we use a single metric called mean Average Precision (mAP), which can combine these things together.

4.1.1 IoU

Before we calculate the final mAP, the first thing we should known is whether our face object was located. To evaluate if an object was located we use the Intersection over Union (IoU) as a similarity measure. IoU computes intersection over the union of the two bounding boxes; the bounding box for the ground truth and the predicted bounding box.
Fig. 8 shows how the Intersection over Union is calculated. An IoU value 1 implies that predicted and the ground-truth bounding boxes are the same. Later we will see that we use different IoU value as threshold to calculate the mAP.

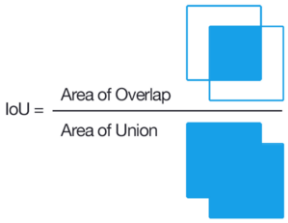

Figure 8: IoU calculation

4.1.2 mean Average precision

Mean Average Precision is the average of AP (Average Precision) of each class. Average Precision is calculated as the weighted mean of precision at each threshold, the weight is the increase in recall from the prior threshold. To compute all the stuff we run the following steps:

- Calculate the IoU for all bbox.
- Set the IoU threshold to 0.5.
- Calculate the confusion matrix-TP, FP, TN, FN.
- Based on the IoU threshold, calculate the precision and recall metrics.
- Measure the average precision.
- Calculate the mAP.
- Iterate the IoU threshold from 0.5-0.95 by step 0.1, repeat previous step to calculate all the corresponding mAP.

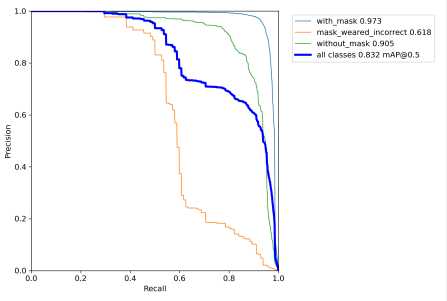

Figure 9: P-R curve of yolo model at IoU threshold 0.5

Fig. 9 shows an example of precision-recall curve of our yolo model at IoU threshold set to 0.5. The way we calculate the AP is just to measure the area below the P-R curve.

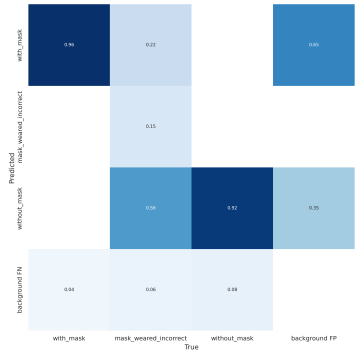

Figure 10: Confusion matrix of yolo model at IoU threshold 0.5

Fig. 10 shows the confusion matrix of yolo model at IoU threshold set to 0.5.Above all shows more clearly why we choose mAP as the evaluation method is that the mAP incorporates the trade-off between precision and recall and considers both false positives (FP) and false negatives (FN). This property makes mAP a suitable metric for most detection applications.

4.2 Experiments results

4.2.1 Final results

We followed the instructions which described in section 4.1.2 and evaluated all three models using the same testing set and same metric. The final result is shown in Fig. 11. It reveals that no matter under what IOU configurations, the Yolo model can get the best of the three results. This is also in line with expectations. Because the yolo model is considered as the most optimized model among them.

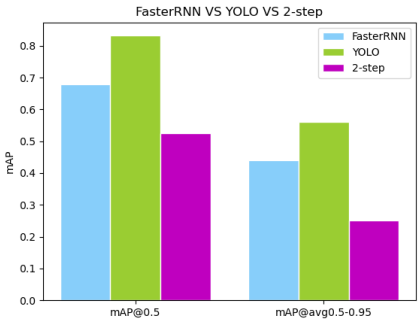

Figure 11: Final results comparison

4.2.2 Results visualization

We visualized all the results on both training and testing sets. Fig. 12 shows three examples of visualized results on the testing set via the yolo model. We can clearly see that our model performs well not only in simple scenarios, but also has relatively decent performance in complex scenarios with small bounding box.

Figure 12: Examples of visualized results on the testing set

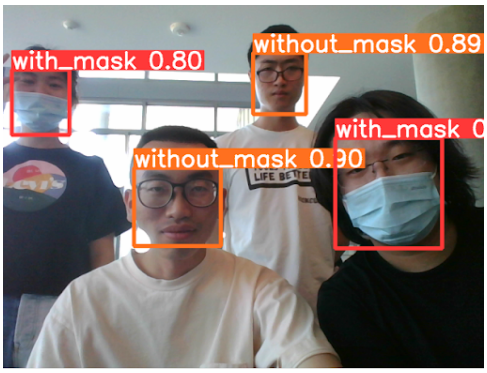

Figure 13: Testing on live video stream

We also wrote a program which can use the camera to processing our trained model on the live video. Fig. 13 is a screenshot of all our project members using the camera to test on the live stream. It shows that our model is also suitable for real time mask detection.

5 Conclusion

In our project, we implemented 3 approach to achieve human face musk detection. Using the same evaluation methods we compared the performance of each other. Although we showed that all three methods can predict musk location and label pretty well, we still found few problems in our training. The first issue is that, for the dataset that used by Object detection methods, it has limited number of images, also its labels is biased for example the number faces with musk is far more larger than faces that without musk or incorrect wearing. This will cause inaccuracy when testing its performance. Second problem is The dataset used by DenseNet in 2-step method, the variety of the musk color and shape is very small, which result to low accuracy when detecting people wear different musk. In the future we are considering find or construct dataset that has various musk type and non biased wearing condition, and redo the training for all three methods.

6 Contribution of Each Member

Tianhan Chen: Implementing and training 2-step human face musk detection network
Xianghu Wang: Constructing part of the Faster-RNN network and evaluating all the models
Tianzi Luo Responsible for project related work and background study and YOLOv5 model.
Tianzhu LiuResponsible for real-time detection front-end design and part of the Faster R-CNN architecture.

References

[1] Benjamin J Cowling, Kwok-Hung Chan, Vicky J Fang, Calvin KY Cheng, Rita OP Fung, Winnie Wai, Joey Sin, Wing Hong Seto, Raymond Yung, Daniel WS Chu, et al. Facemasks and hand hygiene to prevent influenza transmission in households: a cluster randomized trial. *Annals of internal medicine*, 151(7):437–446, 2009.

[2] Samantha M Tracht, Sara Y Del Valle, and James M Hyman. Mathematical modeling of the effectiveness of facemasks in reducing the spread of novel influenza a (h1n1). *PloS one*, 5(2): e9018, 2010.

[3] Tom Jefferson, Chris B Del Mar, Liz Dooley, Eliana Ferroni, Lubna A Al-Ansary, Ghada A Bawazeer, Mieke L Van Driel, Mark A Jones, Sarah Thorning, Elaine M Beller, et al. Physical interventions to interrupt or reduce the spread of respiratory viruses. *Cochrane database of systematic reviews*, (11), 2020.

[4] Shuo Feng, Chen Shen, Nan Xia, Wei Song, Mengzhen Fan, and Benjamin J Cowling. Rational use of face masks in the covid-19 pandemic. *The Lancet Respiratory Medicine*, 8(5):434–436, 2020.

[5] Kaipeng Zhang, Zhanpeng Zhang, Zhifeng Li, and Yu Qiao. Joint face detection and alignment using multitask cascaded convolutional networks. *IEEE signal processing letters*, 23(10): 1499–1503, 2016.

[6] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25, 2012.

[7] Gabriel Cifuentes-Alcobendas and Manuel Domínguez-Rodrigo. Deep learning and taphonomy: high accuracy in the classification of cut marks made on fleshed and defleshed bones using convolutional neural networks. *Scientific reports*, 9(1):1–12, 2019.

[8] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018.

[9] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2014.

[10] Madhura Inamdar and Ninad Mehendale. Real-time face mask identification using facemasknet deep learning network. *Available at SSRN 3663305*, 2020.

[11] Siyuan Qiao, Chenxi Liu, Wei Shen, and Alan L Yuille. Few-shot image recognition by predicting parameters from activations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7229–7238, 2018.

[12] Pierre Sermanet, David Eigen, Xiang Zhang, Michaël Mathieu, Rob Fergus, and Yann LeCun. Overfeat: Integrated recognition, localization and detection using convolutional networks. *arXiv preprint arXiv:1312.6229*, 2013.

[13] Dumitru Erhan, Christian Szegedy, Alexander Toshev, and Dragomir Anguelov. Scalable object detection using deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2147–2154, 2014.

[14] Mingjie Jiang, Xinqi Fan, and Hong Yan. Retinamask: A face mask detector. *arXiv preprint arXiv:2005.03950*, 2020.

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE transactions on pattern analysis and machine intelligence*, 37(9):1904–1916, 2015.

[16] Nhat-Duy Nguyen, Tien Do, Thanh Duc Ngo, and Duy-Dinh Le. An evaluation of deep learning methods for small object detection. *Journal of Electrical and Computer Engineering*, 2020, 2020.

[17] https://blog.paperspace.com/faster-r-cnn-explained-object-detection.

[18] https://pjreddie.com/darknet/yolo/.

[19] Xu Tang, Daniel K Du, Zeqiang He, and Jingtuo Liu. Pyramidbox: A context-assisted single shot face detector. pages 797–813, 2018.

[20] Forrest Iandola, Matt Moskewicz, Sergey Karayev, Ross Girshick, Trevor Darrell, and Kurt Keutzer. Densenet: Implementing efficient convnet descriptor pyramids. *arXiv preprint arXiv:1404.1869*, 2014.