Real-Time Face Mask Object Detection: Project

Ahmed Mostafa : Assiut University, Egypt

July 25, 2021

Abstract

An ongoing global pandemic of coronavirus disease 2019 (COVID-19) is caused by severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2). The novel virus was identified in Wuhan, China, in December 2019; a lockdown in Wuhan and other cities in Hubei province failed to contain the outbreak, and it spread to other parts of mainland China and around the world.

1 Introduction

COVID-19 is mainly transmitted via the respiratory route when people inhale droplets and particles that infected people release as they **breathe**, **talk**, **cough**, **sneeze**, or **sing**. Infected people are more likely to transmit COVID-19 when they are physically close to others. However, infection can occur over longer distances, particularly indoors.

Masks should be used as part of a comprehensive strategy of measures to suppress transmission and save lives; the use of a mask alone is not sufficient to provide an adequate level of protection against COVID-19.

Masks should be used as part of a comprehensive 'Do it all!' approach including physical distancing, avoiding crowded, and more. Depending on the type, masks can be used for either protection of healthy persons or to prevent onward transmission. Make wearing a mask a normal part of being around other people. The appropriate use, storage and cleaning or disposal of masks are essential to make them **as effective as possible**.

In our project, we try to enforce the culture of wearing masks to further prevent the spread of the virus by monitoring vital facilities. It can be used to encourage and help people to wear the mask in important areas. It can be used to prevent people from entering some other areas without it as well.

Masks are a simple barrier to help prevent your respiratory droplets from reaching others. Studies show that masks reduce the spray of droplets when worn over the nose and mouth. You should wear a mask, even if you do not feel sick.

2 Related Work

Now in this related work [1] part I will discuss the some work that has been done in this field. I have selected the following projects: (1) COVID-19: Face Mask Detector with OpenCV, Keras/TensorFlow, and Deep Learning [2], (2) Face mask detection using deep learning: An approach to reduce risk of Coronavirus spread [3], (3) Covid-19 Face Mask Detection Using TensorFlow, Keras and OpenCV [4], (4) The Face Mask Detection For Preventing the Spread of COVID-19 at Politeknik Negeri Batam [5], (5) Real-time Face Mask Detection in Video Data [6].

2.1 COVID-19: Face Mask Detector with OpenCV, Keras/TensorFlow, and Deep Learning

To create our face mask detector, we trained a two-class model of people wearing masks and people not wearing masks. We fine-tuned MobileNetV2 on our **mask/no mask** dataset and obtained a classifier that is $\tilde{9}9\%$ accurate.

We then took this face mask classifier and applied it to both images and real-time video streams by:

1. Detecting faces in images/video.
2. Extracting each individual face.
3. Applying our face mask classifier.

Our face mask detector is accurate, and since we used the MobileNetV2 architecture, it's also *computationally efficient*, making it easier to deploy the model to embedded systems (Raspberry Pi, Google Coral, Jetosn, Nano, etc.)

2.2 Face mask detection using deep learning: An approach to reduce risk of Coronavirus spread

In this work, a deep learning based model for detecting masks over faces in public place to curtail community spread of Coronavirus is presented. The proposed model efficiently handles varying kinds of occlusions in dense situation by making use of **ensemble of single and two stage detectors**. The ensemble approach not only helps in achieving high accuracy but also improves detection speed considerably. The model is 98.2% accurate at mask detection with average **inference times of 0.05 seconds per image**.

The other factors that contributed towards achievement of highly efficient model include application of bounding box affine transformation and transfer learning. The bounding box transformation improves localization performance during mask detection. Transfer learning leads to good results by enabling use of powerful pre-trained model such as ResNet 50 being trained on large dataset like ImageNet.

2.3 Covid-19 Face Mask Detection Using TensorFlow, Keras and OpenCV

In this paper, we briefly explained the motivation of the work at first. Then, we illustrated the learning and performance task of the model. Using basic ML tools and simplified techniques the method has achieved reasonably high accuracy. It can be used for a variety of applications. Wearing a mask may be obligatory in the near future, considering the Covid-19 crisis. Many public service providers will ask the customers to wear masks correctly to avail of their services. The deployed model will contribute immensely to the public health care system. In future it can be extended to detect if a person is wearing the mask properly or not. The model can be further improved to detect if the mask is virus prone or not i.e. the type of the mask is surgical, N95 or not.

2.4 The Face Mask Detection For Preventing the Spread of COVID-19 at Politeknik Negeri Batam

This work developed the face mask detection by using YOLO V4 algorithm. The YOLO V4 algorithm consists of deep learning method which is able to detect the object properly. This device has already been installed at Politeknik Negeri Batam in real-time application to avoid the spread of COVID-19 in campus area. From the experiment results, the algorithm is able to detect and distinguish a **non-wearing and a wearing-mask precisely** with any condition of surrounding environment. In the future, we will add the thermal detection on this device to help the guard's work easier. Furthermore, this device is hopped to be installed in other crowd area which need face mask detector.

2.5 Real-time Face Mask Detection in Video Data

In response to the ongoing COVID-19 pandemic, we present a robust deep learning pipeline that is capable of identifying **correct and incorrect** mask-wearing **from real-time video streams**.

To accomplish this goal, we devised two separate approaches and evaluated their performance and run-time efficiency. The first approach leverages a pre-trained face detector in combination with a mask-wearing image classifier trained on a large-scale synthetic dataset.

The second approach utilizes a state-of-the-art object detection network to perform localization and classification of faces in one shot, fine-tuned on a small set of labeled real-world images. The first pipeline achieved a test accuracy of 99.97% on the synthetic dataset and maintained **6 FPS running on video data**. The second pipeline achieved a mAP(0.5) of 89% on real-world images while **sustaining 52 FPS on video data**.

We have concluded that if a larger dataset with bounding-box labels can be curated, this task is best suited using object detection architectures such as YOLO and SSD due to their superior inference speed and satisfactory performance on key evaluation metrics.

3 Methodology

3.1 YOLOv3

What is YOLO?

You only look once (YOLO) is a state-of-the-art, real-time object detection system. This is an algorithm that detects and recognizes various objects in a picture (in real-time). The YOLO algorithm consists of various variants. Some of the common ones include tiny YOLO and YOLOv3.

YOLO algorithm employs convolutional neural networks (CNN) to detect objects in real-time. The feature extractor of the YOLOv3 contains 53 convolutional layers, and thus it is named Darknet-53. The full YOLOv3 consists of 106 layers, including convolutional, residual, and up-sampling layers. Trainable parameters = 65, 252, 682

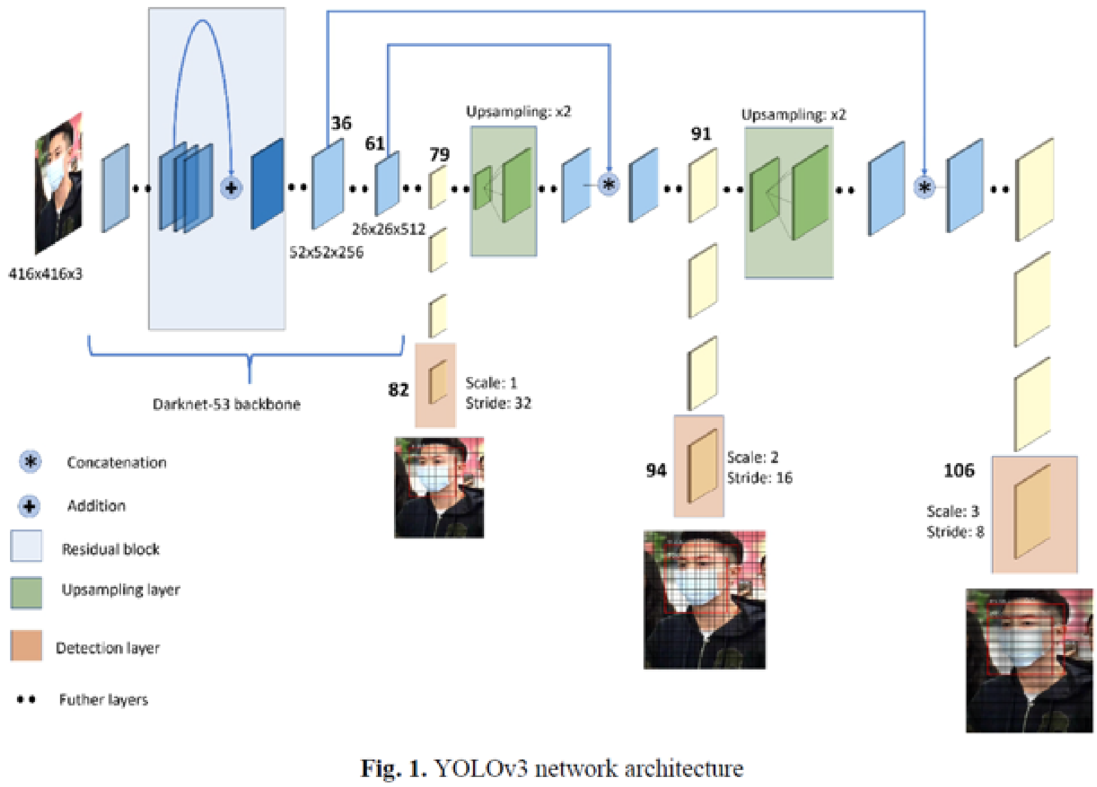
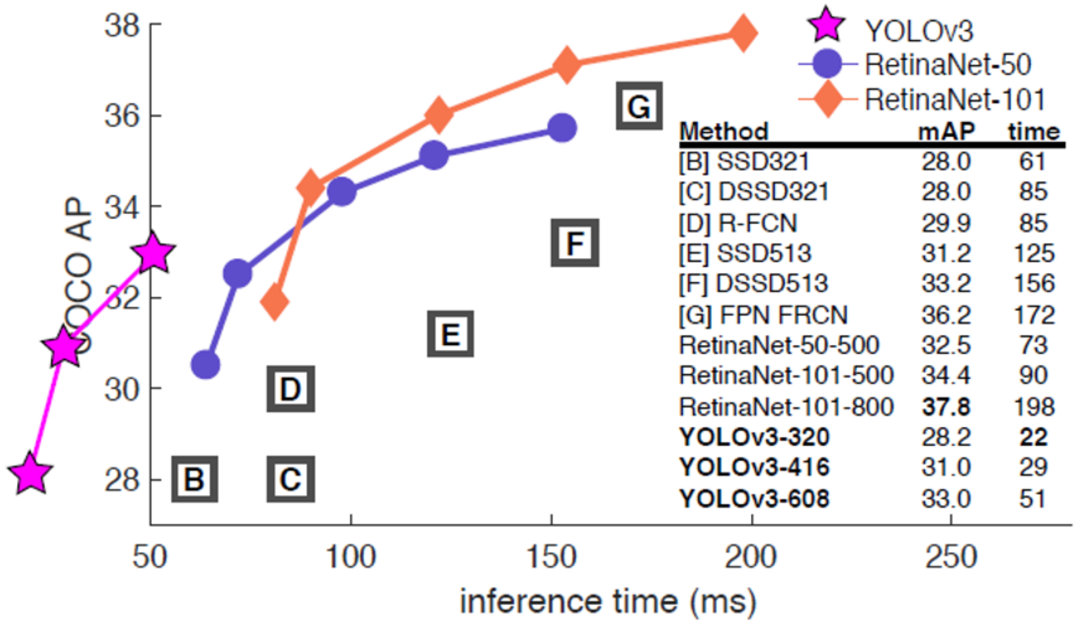


Figure 1: YOLOv3 Architecture

Why YOLO?

Speed. As the name suggests, the algorithm requires only a single forward propagation through a neural network to detect objects. This means that prediction in the entire image is done in a single algorithm run. The CNN is used to predict various class probabilities and bounding boxes simultaneously.

Accuracy. YOLO is a predictive technique that provides accurate results with minimal background errors.



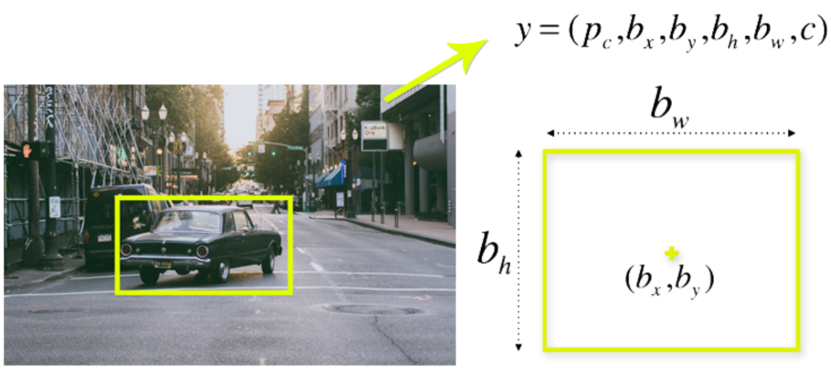
3.2 YOLO Algorithm [7][8]

Algorithms based on regression – instead of selecting interesting parts of an image, they predict classes and bounding boxes for the whole image in one run of the algorithm. The two best known examples from this group are the YOLO (You Only Look Once) family algorithms and SSD (Single Shot Multibox Detector). They are commonly used for real-time object detection as, in general, they trade a bit of accuracy for large improvements in speed.

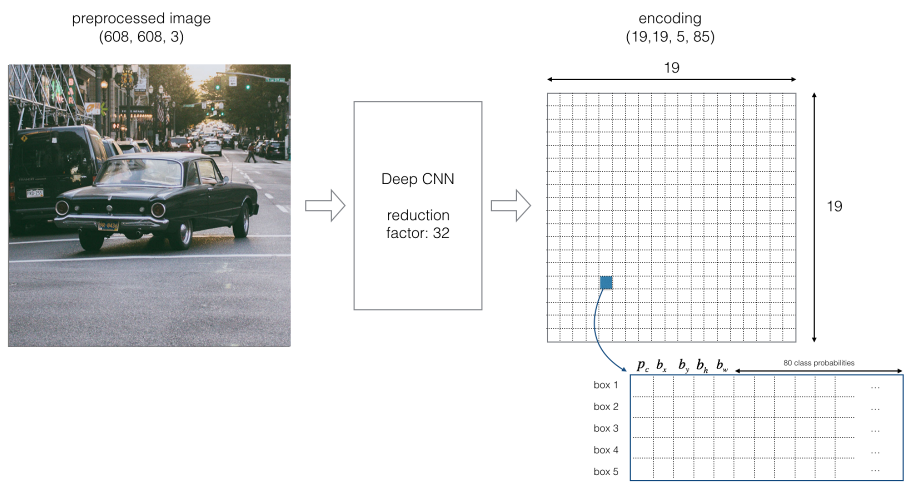
To understand the YOLO algorithm, it is necessary to establish what is actually being predicted. Ultimately, we aim to predict a class of an object and the bounding box specifying object location.

Each bounding box can be described using four descriptors:
1. center of a bounding box (bxby)
2. width (bw)
3. height (bh)
4. value c is corresponding to a class of an object (such as: car, traffic lights, etc.).

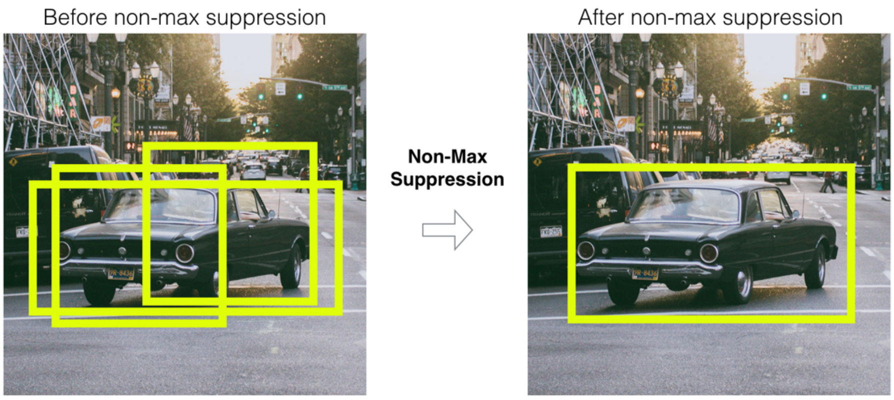
In addition, we have to predict the **pc** value, which is the probability that there is an object in the bounding box.



As we mentioned above, when working with the YOLO algorithm we are not searching for interesting regions in our image that could potentially contain an object. Instead, we are splitting our image into cells, typically using a 19×19 grid. Each cell is responsible for predicting 5 bounding boxes (in case there is more than one object in this cell). Therefore, we arrive at a large number of 1805 bounding boxes for one image.



Most of these cells and bounding boxes will not contain an object. Therefore, we predict the value pc, which serves to remove boxes with low object probability and bounding boxes with the highest shared area in a process called non-max suppression.



3.3 Dataset [9]

Masks play a crucial role in protecting the health of individuals against respiratory diseases, as is one of the few precautions available for COVID-19 in the absence of immunization. With this dataset, it is possible to create a model to detect people wearing masks, not wearing them, or wearing masks improperly. This dataset contains 853 images belonging to the 3 classes, as well as their bounding boxes in the PASCAL VOC format.

The classes are:

- With mask;
- Without mask;
- Mask worn incorrectly.

We split the data to 80% training and 20% testing examples.

3.4 Error Metrics

Mean average precision is an extension of Average precision. In Average precision, we only calculate individual objects but in mAP, it gives the precision for the entire model. To find the percentage correct predictions in the model we are using mAP.

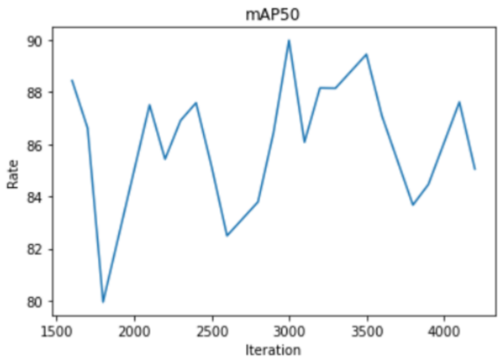
In most of the research papers, these metrics will have extensions like mAP iou = 0.5, mAP iou = 0.75, mAP small, medium, large. In this, we will clearly demonstrate what it actually means.

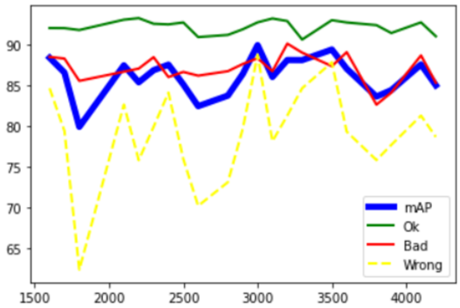
- **mAP iou=0.5** represents the model has used 0.5 threshold value to remove unnecessary bounding boxes, it is the standard threshold value for most of the models.
- **mAP iou=0.75** represents the model has used 0.75 threshold value, By using this we can get accurate results by removing bounding boxes with less than 25% of the intersection with ground truth image.
- **mAP small** represents the model has given mAP score based on smaller objects in the data.
- **mAP** large represents the model has given mAP score based on larger objects in the data.

In this project we were using **mAP iou=0.5**.

3.5 Results

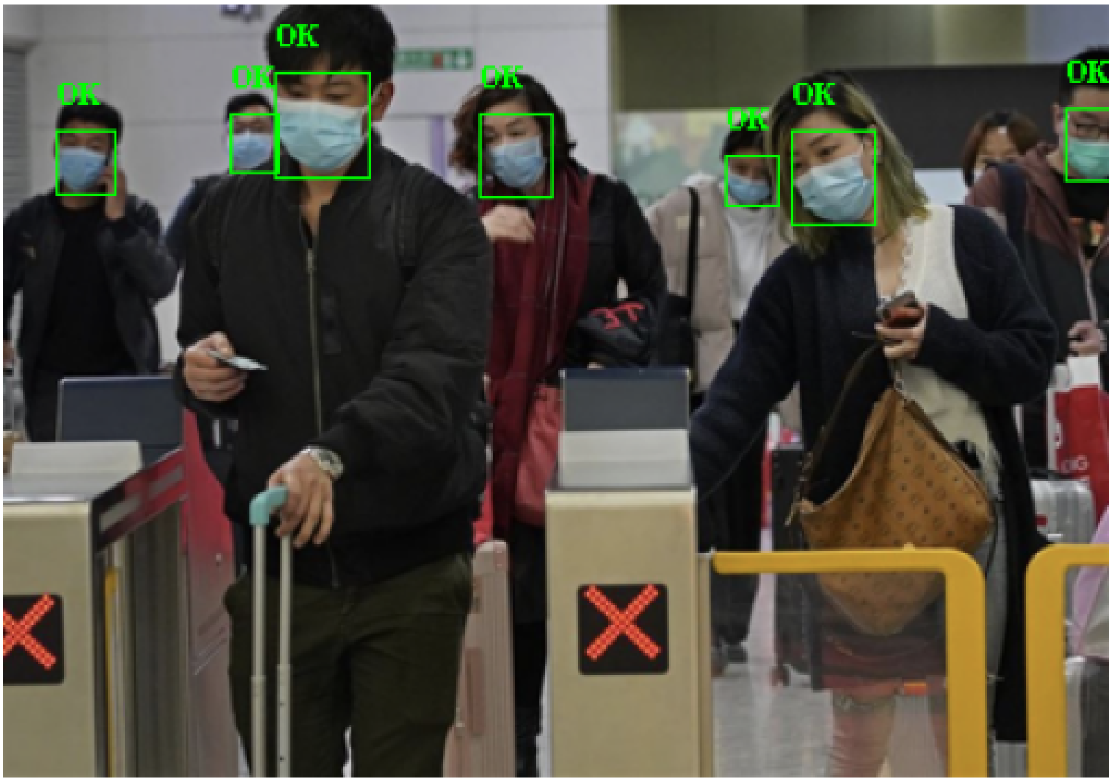
After training the network for about 5000 iterations we were able to reach an mAP 90% for 3 classes (1) with mask, (2) without mask, (3) incorrect.





3.6 Detection Examples

maksssksksss512.png



maksssksksss173.png

4 Conclusion

In this project, Real-time face mask detection, we used YOLOv3 algorithm with darknet53 architecture pre-trained and managed to achieve mAP 90% with the ability to detect in 3-6 frames per second using only CPU (average inference times of 0.2 seconds per image).

The other factors that contributed towards achievement of highly efficient model include transfer learning and a balanced dataset.

References

[1] academia.stackexchange.com (How to write related work)

[2] COVID-19: Face Mask Detector with OpenCV, Keras/TensorFlow, and Deep Learning by Adrian Rosebrock on May 4, 2020

[3] Shilpa Sethi, Mamta Kathuria, Trilok Kaushik, Face mask detection using deep learning: An approach to reduce risk of Coronavirus spread, Journal of Biomedical Informatics, Volume 120, 2021, 103848, ISSN 1532-0464, https://doi.org/10.1016/j.jbi.2021.103848.

[4] Arjya Das; Mohammad Wasif Ansari; Rohini Basak, Covid-19 Face Mask Detection Using TensorFlow, Keras and OpenCV, 2020 IEEE 17th India Council International Conference (INDICON), 05 February 2021, ISSN: 2325-9418, https://ieeexplore.ieee.org/document/9342585

[5] Susanto Susanto; Febri Alwan Putra; Riska Analia; Ika Karlina Laila Nur Suciningtyas, The Face Mask Detection For Preventing the Spread of COVID-19 at Politeknik Negeri Batam, 7-8 Oct. 2020, https://ieeexplore.ieee.org/document/9350556

[6] [Submitted on 5 May 2021] Real-time Face Mask Detection in Video Data, Yuchen Ding, Zichen Li, David Yastremsky, https://arxiv.org/abs/2105.01816

[7] This article was originally written by Michał Maj with further contributions from the Appsilon team. https://appsilon.com/object-detection-yolo-algorithm/

[8] J. Redmon, S. Divvala, R. Girshick and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 779-788, doi: 10.1109/CVPR.2016.91.

[9] Kaggle Dataset "Face Mask Detection" by Larxel with license under public domain, https://makeml.app/datasets/mask