I.J. Intelligent Systems and Applications, **2022, 6, 50-62**
Published Online on December 8, 2022 by MECS Press (http://www.mecs-press.org/)
DOI: 10.5815/ijisa.2022.06.05

Covid-19 Control: Face Mask Detection Using Deep Learning for Balanced and Unbalanced Dataset

Ademola A. Adesokan
American University of Beirut/Department of Computer Science, Beirut, 1107-2020, Lebanon
E-mail: aaa277@mail.aub.edu
ORCID iD: https://orcid.org/0000-0003-3803-5906

Received: 19 June 2022; Revised: 20 August 2022; Accepted: 19 September 2022; Published: 08 December 2022

Abstract: Facemask wearing is becoming a norm in our daily lives to curb the spread of Covid-19. Ensuring facemasks are worn correctly is a topic of concern worldwide. It could go beyond manual human control and enforcement, leading to the spread of this deadly virus and many cases globally. The main aim of wearing a facemask is to curtail the spread of the covid-19 virus, but the biggest concern of most deep learning research is about who is wearing the mask or not, and not who is incorrectly wearing the facemask while the main objective of mask wearing is to prevent the spread of the covid-19 virus. This paper compares three state-of-the- art object detection approaches: Haarcascade, Multi-task Cascaded Convolutional Networks (MTCNN), and You Only Look Once version 4 (YOLOv4) to classify who is wearing a mask, who is not wearing a mask, and most importantly, who is incorrectly wearing the mask in a real-time video stream using FPS as a benchmark to select the best model. Yolov4 got about 40 Frame Per Seconds (FPS), outperforming Haarcascade with 16 and MTCNN with 1.4. YOLOv4 was later used to compare the two datasets using Intersection over Union (IoU) and mean Average Precision (mAP) as a comparative measure; dataset2 (balanced dataset) performed better than dataset1 (unbalanced dataset). Yolov4 model on dataset2 mapped and detected images of masks worn incorrectly with one correct class label rather than giving them two label classes with uncertainty in dataset1, this work shows the advantage of having a balanced dataset for accuracy. This work would help decrease human interference in enforcing the COVID-19 face mask rules and create awareness for people who do not comply with the facemask policy of wearing it correctly. Hence, significantly reducing the spread of COVID-19.

Index Terms: Covid-19, Facemask, FPS, Haar-cascade, IoU, mAP, Mask-Wearing, MTCNN, YOLOv4.

1. Introduction

History has shown the vast negative impacts of recorded pandemics globally, Covid-19 is not an exemption, which most sectors felt its negative impact. The coming of the COVID- 19 pandemic has had an enormous impact on humanity and the healthcare sector. According to World Health Organization (WHO), as of 12th December 2021, there are 268,934,575 confirmed cases and 5,297,850 confirmed deaths [1], as depicted by regions in Table 1. COVID-19 primarily spreads by droplets discharged by an infected person via coughing or sneezing; thus, there is a high chance of spreading the virus to everyone who directly contacts an infected person with the coronavirus [2]. As a result, the virus spreads quickly among the public at large, which might not show immediate signs and symptoms.

The global controversy about taking the vaccine or not has been a contending issue by individuals and social media on how the vaccine might affect the fertility and pregnancy rate [4]. According to studies in [5], the most often reported adverse effects include fever, headache, weariness, and pains at the injection site, with the majority of side effects being mild to severe. However, like with the two-dose or single-dose main series, there is a small risk of side effects, which may be insignificant. The truth remains that the number of covid-19 vaccinations provided is still insufficient compared to the total world population, particularly in the least developed countries. According to data in [6], there are two doses offered to individuals; however, the number of people who received at least one of the two doses is roughly 56.1% of the worldpopulation, with 8.51 billion doses administered globally and 35.6 million administered daily.

One of the recommended among other methods to control the spread of the covid-19 virus that does not include vaccinations with little controversy and is primarily adopted in different facets of life is the use of facemasks to prevent the droplets from spreading. And the wearing of face masks in line with other measures has been part of our daily lives in respectiveof whether the vaccine has been taken or not.

Table 1. COVID-19 Confirmed Cases and Death [3].

WHO Region	New Cases in Last 7 Days (%)	Change in New Cases in Last 7 Days * (%)	Cumulative Cases (%)	Change in New Death in Last 7 Days * (%)	Cumulative Deaths (%)
Europe	2,593,221 (65%)	-7	91,631,852 (34%)	-3	1,598,688 (30%)
Americas	837,345 (21%)	-10	98,521,311 (37%)	-14	2,371,246 (45%)
West Pacific	213,915 (5%)	7	10,584,344 (4%)	4	147,539 (3%)
Africa	167,682 (4%)	111	6,522,517 (2%)	-1	153,766 (3%)
South-East Asia	98.021 (2%)	-10	44,737,006 (17%)	-50	714,303 (13%)
Eastern Mediterranean	90,633 (2%)	-4	16,936,781 (6%)	-3	312,295 (6%)
Global	**4,000,817 (≈100%)**	**-5**	**268,934,575 (100%)**	**-10%**	**5,297,850 (100%)**

The aim and objectives of this work is to correctly detect class label of mask worn incorrectly with two datasets (balanced and unbalanced) using one of the three proposed models with the best Frame Per Seconds (FPS) using intersection over Union (IoU) and mean Average Precision (mAP) as evaluation metrics.

1.1. Problem Statement and Approach

There have been numerous steps to reduce the transmission of COVID-19, several techniques are being developed, including wearing a face mask to help prevent the transfer of droplets from infected persons. The WHO has suggested different preventive measures for protection, and one such preventive action is to wear a facemask while visiting open places. Research has also shown that wearing face masks is 96% effective in decreasing the spread of the virus [2]. Even with the strict rules enforced by different governments globally and WHO on wearing a face mask to curtail the virus, many people still defy the measures [7].

Some of the researchers in [2,7-9] have worked towards using machine learning methods and applications to solve a similar problem. However, most researchers in the field have emphasized their work on applying machine learning techniques on wearing facemasks and not wearing facemasks. Nevertheless, there is still a gap in knowing who is wearing the face mask correctly or incorrectly, either by not covering their nose or mouth or dropping it to their chin.

This work proposed two datasets that focus on three classes with names, with face masks, without face masks, and improperly worn face masks. Considering the differences in these classes, most of the existing datasets perform poorly in real world usage on different models. The collected datasets consist of faces with masks, without masks, and improperly worn masks. The proposed methods are adapted for mask detection in real-time videos, benefiting from this dataset.

And the collected datasets used were obtained from two sources – dataset1 from Kaggle [10] and dataset2 from GitHub repo [11,12]. Dataset1 from Kaggle contains 853 images with their respective annotations, but there was an imbalance between the three classes, as shown in Fig. 1. In this paper, the dataset1 was run on three different widely used mask detection models: You Only Look Once version 4 (YOLOv4) object detection architecture, Haarcascade, and Multi-task Cascaded Convolutional Networks (MTCNN).

The method compares the three models to find and classify masked, unmasked, and incorrectly masked faces. In a further experiment, YOLOv4 was used to analyze the two datasets because it performed well with frame per seconds (FPS) greater than 48 on the datasets for facemask detection based on the three classes compared with the other two methods. The dataset was run with the three models to see their effectiveness, and YOLOv4 outperforms the Haarcascade and MTCNN in terms of FPS.

Therefore, YOLOv4 achieves high accuracy faster than the Haarcascade and MTCNN. This method would help implement the facemasks wearing policy while it will allow policymakers to know who is wearing the facemasks correctly or not and those not wearing at all. As a result, this approach will aid in the control of the spread of covid-19 in congestedand public settings by accurately detecting who is incorrectly wearing the facemask.

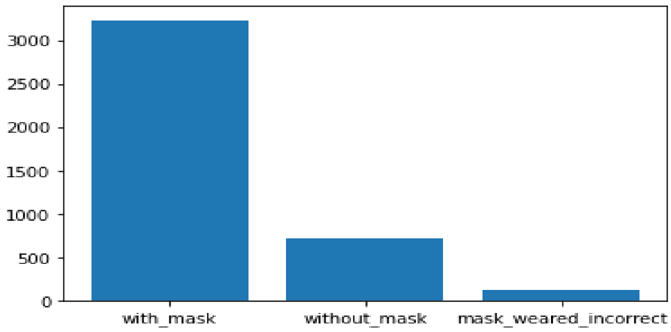


Fig.1. Dataset1 Class Imbalance.

1.2. *Paper's Organization*

After the brief introduction on the impact of Covid-19 and how to curtail the spread of the virus, followed by the aim and objectives and problem that arises from improper facemask wearing and how the process could be automated using deep learning approaches, the remainder of this paper has the following structure. Section 2 includes a survey of the related work by other researchers who have done similar work on facemask detection using different approaches. Section 3 entails the description of the datasets used in this work and the ratio of splitting the datasets. Section 4 outlines the paper's methodology used to achieve the objective of this research. Section 5 contains the paper's research experiment and results that include the model training, testing, and evaluation, including the results of the experiments and models used in this paper. Section 6 has concluding observations and future work for facemask detection.

2. Related Work

The use of facemasks cannot be overlooked when controllable measures are to be considered for the spread of Covid-19 droplets, with stringent policies and preventive measures on the wearing of facemasks by different governments worldwide. Hence, the emergence and implementation of various ways to automate these measures and ensure that the policies are strictly adhered to have been churned out using different machine learning approaches [13,14,15,16,17,18]. Studies have been conducted on facemask detection and classification, though most of the research was more concerned about wearing facemasks and not wearing facemasks.

Reference [19] have applied YOLO to detect facemask on a newly collected dataset. This is because face detection is a vital part of the facemask's detection process; it requires a significant amount of time and resources if done manually and increases the chance of making mistakes in detecting unmasked faces.

According to the state of arts and the results which have already been tested by some researchers, in this work [20], they developed a face mask detection for COVID-19 prevention by using the YOLOv4 algorithm at Politeknik Negeri Batam in real-time application to avoid the spread of COVID-19 in the campus area to detect who is/not wearing facemasks.

Reference [21] presents the masked face detection by comparing the performance of 3 famous algorithms of machine learning: K-Nearest Neighbor, Support Vector Machine, and MobileNet on different scenarios where it was discovered that MobileNet outperformed the two other algorithms considering the accuracy of the images and videos from a real-time camera.

Facemask detection uses four different steps: camera distance, eye line detection, and facial part detection [22]. Because video analytics is concerned with detecting people and events such as moving, dropping, and so on, the authors take advantage of the fact that a human and face detector is present in the system, while the algorithm analysis proposed improvements for the performance of facemask detection.

A real-time algorithm for face tracking was developed on two classes (wearing a mask and not wearing a mask) [23]. The face tracking algorithm performs better in detecting who is wearing a mask than those not wearing a face mask. But it has a problem of losing track of target in some cases for masked faces. The detector is only trained with faces without masks; they show that the proposed algorithm performs robustly in tracking faces without and with face masks.

Reference [24] was conducted to detect real-time facemask wearing using deep learning for large traffic datasets. In order to further verify the performance of the model, a speed measurement experiment was carried out on the FDDB dataset with the MY method and the classic face detection methods based on deep learning. The detection accuracy reached 0.919, and FPS got to 55. This work has been implemented in places like Beijing to curtail the spread of the epidemic by using an infrared thermal imaging temperature measurement warning system.

Reference [25] have introduced a novel approach for detecting and classifying a person wearing a mask or no mask on mobile devices using state-of-the-art MobileNetV2 architecture with transfer learning technique. This method uses a smartphone camera to test videos and images, which gained an accuracy of 99.2% in training and validation accuracy of 99.8%

Convolutional Neural Networks were used for facemask recognition, where Multi-Task Cascaded Convolutional Neural Network was used for face detection, while Google FaceNet embedding was used for facial features extraction, and finally, Support Vector Machine was used to perform the classification task [26]. FaceNet's pre-trained model was utilized to improve masked face recognition.

Facial parts were highlighted for face detection and their importance [27]. This paper measures the accuracy of three models (MTCNN, Retinaface, and DLIB). The model has the shortcoming of MTCNN not being able to detect a larger face when there is a face inside another look. Their work found an interesting behavior of the famous face detection algorithm MTCNN.

Reference [28] uses deep learning method to detect facemask in real- time by pre-processing, training a CNN, and finally performing real-time classification with 96% validation accuracy. The model used red colored rectangle to denote no mask and a green colored rectangle to portray mask-wearing. This method rapidly helps millions of people infected by the Corona Virus throughout the world

Taking into account the accuracy and reasoning speed of face mask detection tasks, paper [29] proposes a detection model using PP-YOLO-Mask based on PP-YOLO through transfer learning, data augmentation, and model compression methods with mAP of 86.69% and 11.842ms recognition speed for a single picture which perform better in terms of accuracy and speed than YOLOV3 and FasterRCNN. Region-based Convolutional Neural Network was utilized for facemask detection while comparing the mAP with SSD Inception V2 model and SSD MobileNet V2 model [30]. Their proposed method achieved 85.82%, mAP on the collected dataset. It also has a detection accuracy of 98.61% mAP for people not wearing facemask while masked faces have 68.72%. This was better than the total Mean Average Precision of the SSD Inception V2 model and for the SSD MobileNet V2 model.

On the Simulated Masked Face Dataset, [31] identify the face mask by presenting CNN and VGG16-based deep learning models to include and implement AI-based preventive methods to control covid-19 spread. The CNN model has a testing accuracy of 97.42%, validation of 96.35%, and 96.35% for training. And the VGG16 model achieved 99.47% training, 98.59% validation, and 98.97% test accuracy.

Real-time detection was done from the YOLO family to measure the performance of neural networks [32]. This method was compared with the Mask-R-CNN using computational and efficiency while increasing processing speed without compromising accuracy for detection.

Most of the related works have concentrated their research more on facemask detection using two label classes for mask wearing and not wearing mask. Therefore, this call for more research to consider people that are incorrectly wearing the facemasks.

3. Datasets

Dataset1 was gotten from Kaggle, and it has 853 labeled images that entail different categories of images, such as individual and group pictures. Dataset1 has three classes (with- mask, without-mask, and mask-wear-incorrect), and using data analysis; it depicts an imbalance between the classes as earlier shown in Fig. 1 with more data for the with-mask class and less data for mask-wear-incorrect.

The performance of the models for training, validation, and testing is dependent on equal class representation. In order to remediate the class imbalance bias and not cause the models to perform poorly in class representation. In this paper, dataset2 has 3000 images that were taken from [11,12], with each class having a thousand each to create balance.

Dataset1 has 853 data with 545 training data (80% of 682), 137 validation data (20% of 682), and 171 test data (20% of total dataset1), while dataset2 comprises 3000 data with 1920 training data (80% of 2400), 480.

validation data (20% of 2400), and 600 test data (20% of total dataset2).

The rationale behind using the two datasets is because dataset1 is the most well-known on Kaggle for facemask detection. However, because of its imbalance nature between the three class labels, it has produced biased results, and this calls for dataset2 to clear all doubt attached to the imbalance class labels in dataset1.

4. Methodology

In real-time video detection, and most importantly, congested situations like public places, the main features to be considered are fastness in terms of frame capture per seconds and accuracy in detection, which is this paper's main goal and objective in its methodology. Based on studies and research on modern models that have been used for facemask detection with good accuracy and fastness, the paper uses these three models: Haarcascade, MTCNN, and YOLOv4. The rationale behind this paper is to find the model with the best FPS on the controversial dataset1 as it has been the most common dataset been considered for facemask detection in real-time videos in most facemask detection studies.

Then, this work uses the model with the best performance to compare the IoU and mAP for dataset1 and dataset2. The three models and their brief introduction on how they work are explained below.

4.1. Haarcascade

Haarcascade is a machine learning-based technique that allows images of positive (images with face) and negative (without faces) images to be trained using the cascade function. It is also an effective classifier used for objection detection proposed by [33,34] and primarily used to detect an object in images and video streams.

In addition, features of the Haar classifier are synonymous with convolutional networks, where every component is calculated by deducting a single value of the total of the pixels in the white rectangle out of the sum of the pixels in the blackrectangle as in Fig. 2.

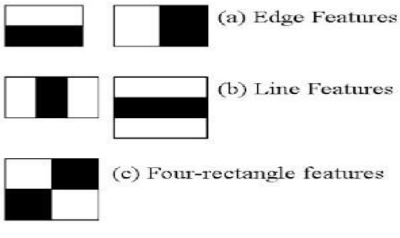
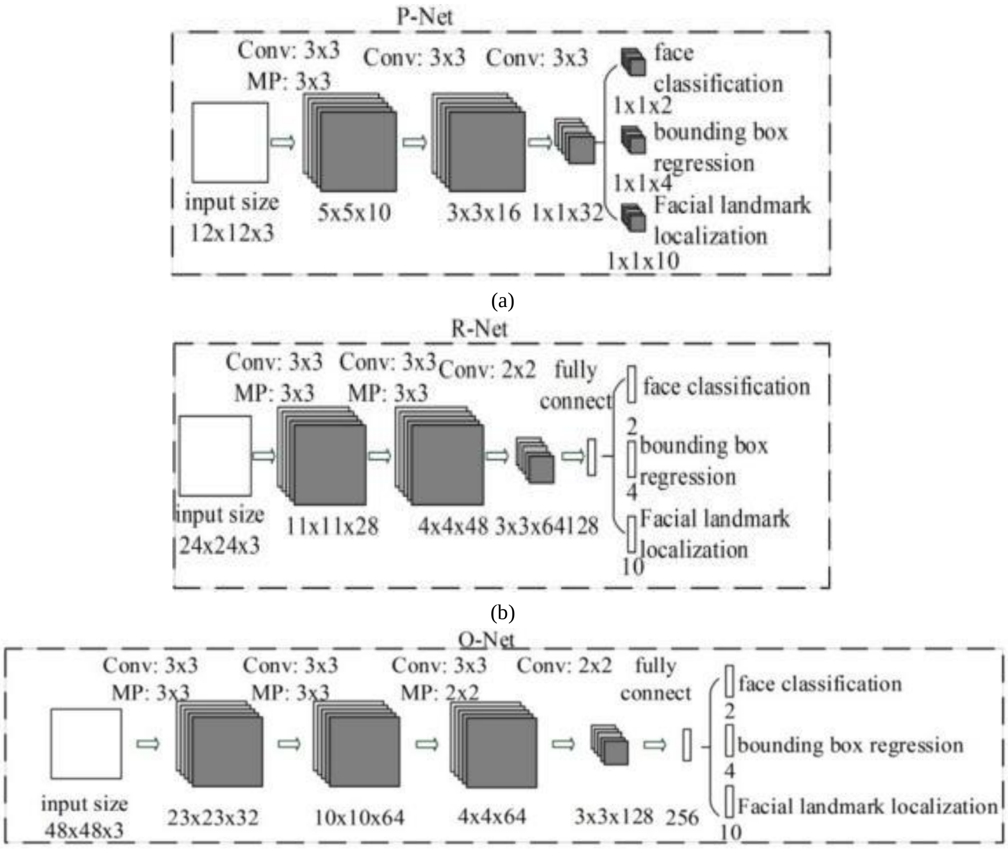


Fig.2. Haarcascade Features [34].

4.2. MTCNN

Multi-task Cascaded Convolutional Networks (MTCNN) is a state-of-the-art architecture used for human face detection that uses multitask learning. It can detect facial features such as mouth and eyes on a different range of datasets with the aim of achieving three major tasks face classification, bounding box regression, and localization of facial landmark.

MTCNN uses a three-stage cascading structure for easy detection of facial features [35]. Firstly, the proposal network (P-NET) of a fully CNN in Fig. 3a is used to obtain bounding box and candidate windows, and FCNN does not use a dense layer, which majorly differentiates them from CNN. Secondly, the refine network (R-Net) as shown in Fig. 3b, is a CNN that receives the output of P-NET as its input which helps in reducing the number of candidates, bounding box calibration, and merging overlapping candidates. The production of the result shows if there is the presence of a face or not. Finally, the output network (O-NET) in Fig. 3c gives more information description than the R-NET, andits output produces eyes, nose, and mouth facial landmarks.



(c)

Fig.3. (a) P-Net (b) R-Net (c) O-Net [35]

4.3. YOLOv4

You only look once (YOLO) is one of the most popular and powerful real-time objects detection approaches, and its name (YOLO) was coined because it is a modern algorithm built to look once through a network. The YOLO object detector is a quick and accurate one-stage object detector. YOLO is being used in many projects and research works, such as in [10,15,19,20,29,32]. The first three years of 2016, 2017, and 2018 experienced the version release of YOLOv1, YOLOv2, and YOLOv3, respectively, and YOLOv4 was released was in 2020 which included the several state-of-the-art bags of specials (BoS) and bags of freebies (BoF) [36]. The BoS increase a small amount of inference cost while improving the detection accuracy of the object. At the same time, the BoF tends to enhance the detector's accuracy without the inferencetime increase.

Regarding performance metrics, [37] asserts that YOLOv4 has better results than other models for detecting objects by a vast difference. YOLOv4 has achieved more than 12% FPS and 10% AP than YOLOv3. As a consequence, it received a 43.5% AP value based on the COCO dataset, as well as a real-time performance of 65 frames per second on the TESLA V100 [38]. As a result, it outperformed the most accurate and quickest detector in terms of accuracy and speed. The working principles of YOLOv4 in Fig. 4 have been explained in many research work, some of which includes the following:

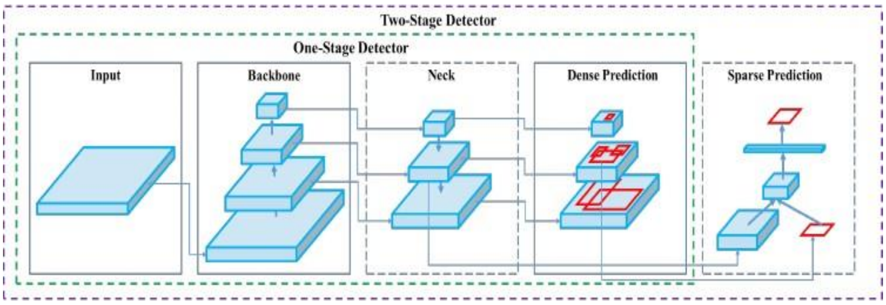


Fig.4. YOLOv4 [37]

- Input: Image, Patches, Image Pyramid
- Backbones: VGG16 [39], ResNet-50 [40], SpineNet [41], EfficientNet-B0/B7 [42], CSPResNeXt50 [43], CSPDarknet53 [43]
- Neck
  - Additional blocks: SPP [44], ASPP [45], RFB [46], SAM [47]
  - Path-aggregation blocks: FPN [48], PAN [49], NAS- FPN [50], Fully connected FPN, BiFPN [51], ASFF [52], SFAM [53]
- Heads
  - Dense Prediction (one-stage):
    - RPN [54], SSD [55], YOLO [56], RetinaNet [57] (anchor based)
    - CornerNet [58], CenterNet [59], MatrixNet [60],FCOS [61] (anchor free)
  - Sparse Prediction (two-stage):
    - Faster R-CNN [62], R-FCN [63], Mask RCNN [64] (anchor based)
    - RepPoints [65] (anchor free)

5. Discussions, Evaluation and Result

5.1. Data Preprocessing

Dataset1 has the directory structure with 853 images with their respective annotations as shown in Fig. 5. The 853 images in the annotation directory have bounding boxes in PASCAL VOC format. In this paper, the input uses the bounded region by the bounding box while the output uses the labels. The following steps were used in the data preprocessing of dataset1.

- Import libraries such as os, cv2, pickle, numpy, xml and Element Tree, matplotlib, to-categorical.
- Reading image path, label, bounding boxes from XML file – image and annotation path was read, and they were labeled as 'without-mask': 0, 'with-mask': 1, 'mask- weared-incorrect': 2
- The data were randomly checked to know what they look like and drawing bounding boxes while showing the images
- Data and target were gotten by taking region of interest as data(X) and target as categorical data as in Fig. 6 with the image size as (100, 100) for X and Y.
- Finally, the preprocessed data was saved.

```
Face Mask Detection
│
├───data
│   ├───annotations
│   │       .xml
│   └───images
│           .png
│
...
```

Fig.5. Images and Annotations

```
0 -> [1, 0, 0]
1 -> [0, 1, 0]
2 -> [0, 0, 1]
```

Fig.6. Categorical Data

5.2. Mask Detector

The mask detector model was done by training the inputs with transfer learning and InceptionV3 neural network architecture was used to build the model classifier for the three classes. The following steps were taken in training the mask detector.

- Import libraries but included seaborn, tensorflow, InceptionV3, train-test-split and confusion-matrix
- Transfer learning (InceptionV3) model from keras was used with weights = 'imagenet', input-shape= (100, 100, 3) to get the pre-trained model summary as shown in table 2. Table 3 also shows that there are 21,802,784 of total parameters with 21,768,352 trainable parameters and 34,432 non-trainable parameters.
- Keras was used to apply flatten, dense, activation, dropout to the final layer with 3 outputs for 3 categories to get a total parameter of 21,562,275, trainable parameters of 12,587,011, and non-trainable parameters of 8,975,264.
- The model was compiled and data were prepared for training with X.shape = (4072, 100, 100, 3) and Y.shape = (4072, 3).
- Data normalization was done to be X = X / 255 before splitting.
- The next step trained the dataset for 20 epochs and accuracy on the test set was 94.85%.

Table 2. InceptionV3 Model

Model "Inception_V3"			
Layer (type)	**Output Shape**	**Param #**	**Connected to**
input_2 (*InputLayer)*	[(None, 100, 100, 3)]	0	{}
conv2d_94 *(Conv2D)*	(None, 49, 49, 32)	864	['input_2[0][0]']
batch_normalization_94 (*BatchNormalization)*	(None, 49, 49, 32)	96	['conv2d_94[0][0]']
activation_94 *(Activation)*	(None, 49, 49, 32)	0	['batch_normalization_94[0][0]']
conv2d_95 *(Conv2D)*	(None, 47, 47, 32)	9216	['activation_94[0][0]']

Table 3. Model Parameters

Layer (type)	**Output Shape**	**Param #**	**Connected to**
Concatenate_3 *(Concatenate)*	[(None, 1, 1, 768)]	0	['activation_185[0][0]', 'activation_186[0][0]']
activation_187 (*Activation)*	(None, 1, 1, 192)	0	['batch_normalization_187[0][0]']
Mixed10 (*Concatenate)*	(None, 1, 1, 2048)	0	['activation_179[0][0]', 'mixed9_1[0][0]', 'concatenate_3[0][0]', 'activation_187[0][0]']
Total params: 21,802,784 ***Trainable params: 21,768,352*** ***Non-trainable params: 34,432***			

- The loss and accuracy were plotted as depicted in Fig. 7 and the model was saved

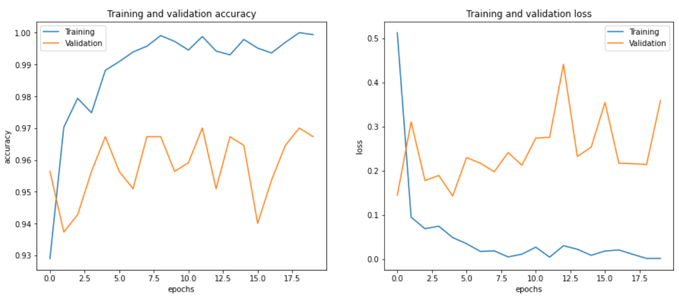


Fig.7. Model Accuracy and Loss

- Then finally, the true and predicted label was plotted as illustrated in Fig. 8.

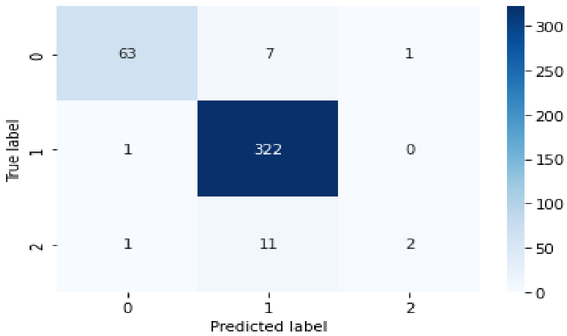


Fig.8. True and Predicted Label

5.3. Facemask Detection in Real-Time

In detecting the three classes in real-time, the model was loaded, and the detection from the webcam using Haarcascade from OpenCV gave an average of 16 FPS, but there was a bit of inaccuracy with the classifier in struggling to detect masked faces. When MTCNN was used to detect faces and classify the three classes, it performed well in detecting faces in the class categories, but it has an FPS of about 1.4, which is very low for crowded places that require fast quick checking of people.

Finally, for YOLOv4, the FPS was about 40 with the fact that it performed faster and detected quickly through the frames than Haarcascade and MTCNN. Because it detects faster and accurately with good FPS, YOLOv4 was then chosen to compare the iOU and mAP of dataset1 and dataset2.

5.4. Balance and Unbalanced Datasets

Because dataset1 has been labeled, in order to compare dataset1 and dataset2 to get better results, dataset2 was cleaned by labeling the data and the bounding boxes. The bounding boxes and the classes are labeled using the method in [66]. The COCO dataset's pre-trained weights were employed for transfer learning in data modeling. And the hyperparameter tuning was done in the configuration file with a few changessuch as.

- Batches subdivision to determine the parallel images processed
- Mosaic to curb over-reliance on vital features
- Blur was applied randomly at the average time
- Height and width to increase and improve resolution size
- Jitter for aspect ratio and to change images randomly
- Changes to hue and saturation

The model was trained to show the difference between balanced and unbalanced datasets. The mean average precision (mAP) and intersection over union (IOU) were utilized as a comparative metric. The result is shown in Fig. 9 and 10, with negligible difference between both datasets.

A. IoU and mAP

IoU is a measure of accuracy used for tightness of bounding box mainly used when gathering annotations for humans. In contrast, mAP is used for evaluating the detection of an object's accuracy from a model. IoU and mAP are related inversely. High IoU leads to a decrease in mAP, as depicted in Fig. 11. IoU iteration was taken as 0.25, 0.50, 0.75 and 0.90. Table 4 shows the result for IoU in relation to mAP for the two datasets, and it can be concluded that IoU = 0.95 gave mAP of 98.65 for dataset1 (Unbalance) while for dataset2, mAP was 98.19. Also, for IoU = 0.90, mAP for dataset1 is 8.88 while dataset2 has 13.27 mAP. It can be deduced that to set high IoU, there is a need to have a very high accurate bounding box, just like in Fig. 11 with excellent score for IoU = 0.9264.

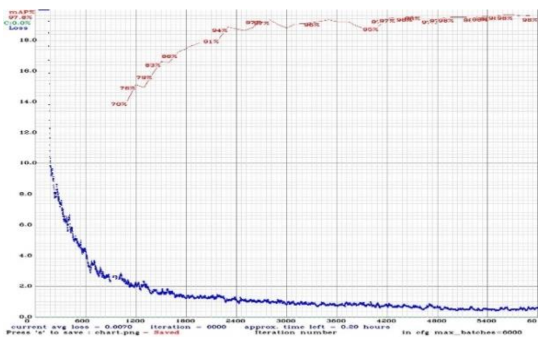


Fig.9. Dataset1 training

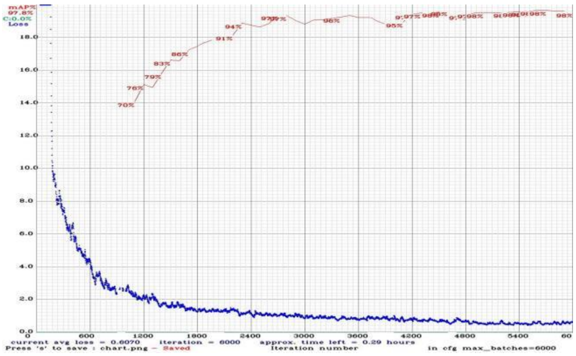


Fig.10. Dataset2 training

Fig.11. IoU Bounding [67]

Table 4. IoU and mAP Evaluation

Dataset 1 (Unbalance Dataset)					*Dataset 2 (Balance Dataset)*			
IoU= 0.25, mAP = 98.65					*IoU= 0.25, mAP = 98.19*			
	TP	FP	AP			TP	FP	AP
with_mask	754	48	99.26		with_mask	752	44	99.34
No_mask	151	28	98.89		No_mask	146	18	97.97
Incorrect_mask wear	39	1	97.8		Incorrect_mask wear	44	3	97.25
IoU= 0.90, mAP = 8.88					*IoU = 0.90, mAP = 13.27*			
with_mask	195	607	12.99		with_mask	208	588	15.62
No_mask	23	156	6.27		No_mask	36	128	12.63
Incorrect_mask wear	7	33	7.39		Incorrect_mask_wear	13	34	11.55

B. Visualizations

Fig. 12 and 13 displays a visual from the model's output where dataset2 performed better than dataset1 because of its balanced dataset classification. For Fig. 12, dataset1 classify two labels to a face as not wearing a mask and incorrectly wearing a mask, while Fig. 13 clearly classified and labeled that the face was incorrectly wearing a mask.

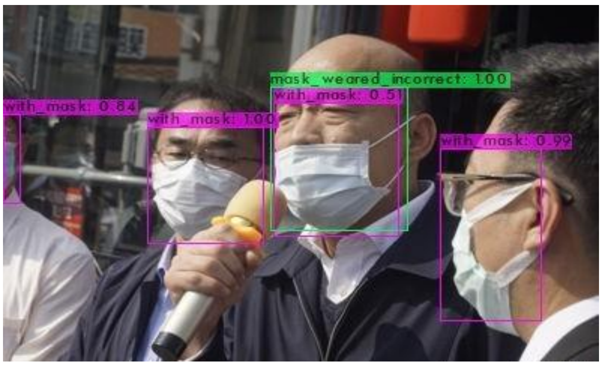


Fig.12. Dataset1 Visual Result

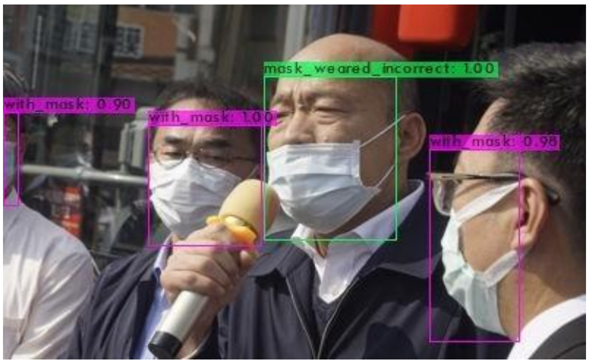


Fig.13. Dataset2 Visual Result

6. Conclusion and Future Work

The Covid-19 era has changed our lifestyle to adopt wearing the facemask as a preventive measure to being infected by the virus. Most facemask detection techniques have been implemented with two classes to know who is wearing the facemask or not. However, the aim of wearing a facemask might be defeated if droplets can still be transmitted while wearing the mask incorrectly. This paper compares three object detection approaches (Haar-cascade, MTCNN, and YOLOv4) to detect and classify faces by considering three classes and labels of who is wearing a facemask, not wearing a facemask, and incorrectly wearing a facemask. The three approaches were to achieve the best FPS as real-time facemask detection in crowded places has to do with how fast and quick plus the number of people the model can detect in every second. This paper has two datasets, with dataset1 being an unbalanced dataset and dataset2 being a balanced dataset with an equal number of classes. FPS was checked using the three approaches, and YOLOv4 outperforms Haar and MTCNN. In order to gain more insights from this paper, IoU and mAP were used as a yardstick to compare the two datasets, and Dataset2 performs better than dataset1 in terms of accuratefacemask classification.

One of the limitations in this work is the need to have more accurate bounding boxes so as to set high IoU to achieve a more robust result which is an avenue for future work in facemask detection. And also, future work plan to build a real-life system that would use real life datasets from social media (Instagram) or live video, and also after facemask detection, victims can get emails or text messages to inform them that they are not wearing the mask properly.

Acknowledgment

I wish to use this opportunity to thank Dr. Shady Elbassuoni and for his guidance and feedback towards the completion of this work. My utmost appreciation goes to Mastercard Foundation Program at American University of Beirut for the privilege to participate in this program.

References

[1] World Health Organization 2021. Coronavirus disease (COVID-19) – World Health Organization. [online] Available at: who.int [Accessed 14 December 2021].

[2] H. Adusumalli, D. Kalyani, R. K. Sri, M. Pratapteja and P. V. R. D. P. Rao, "Face Mask Detection Using OpenCV," 2021 Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV), 2021, pp. 1304-1309, doi: 10.1109/ICICV50876.2021.9388375.

[3] "Weekly epidemiological update on COVID-19 - 14 December 2021," Who.int. [Online]. Available: https://www.who.int/publications/m/item/ weekly-epidemiological-update-on-covid-19 14-december-2021. [Accessed: 14-Dec-2021].

[4] "COVID-19 vaccines: Myth versus fact," Hop- kinsmedicine.org.[Online]. Available: https://www.hopkinsmedicine.org/health/conditions-and-diseases/coronavirus/covid-19-vaccines-myth-versus-fact. [Accessed: 14-Dec-2021].

[5] CDC, "Possible side effects after getting a COVID-19 vaccine," Centers for Disease Control and Prevention, 14-Dec-2021. [Online]. Available:https://www.cdc.gov/coronavirus/2019-ncov/vaccines/expect/ after.html. [Accessed: 14-Dec-2021].

[6] H. Ritchie et al., "Coronavirus Pandemic (COVID-19)," Our World in Data, 2020.

[7] A. Negi, P. Chauhan, K. Kumar and R. S. Rajput, "Face Mask Detection Classifier and Model Pruning with Keras-Surgeon," 2020 5th IEEE International Conference on Recent Advances and Innovations in Engineering (ICRAIE), 2020, pp. 1-6, doi: 10.1109/ICRAIE51050.2020.9358337.

[8] M. S. Ejaz, M. R. Islam, M. Sifatullah and A. Sarker, "Implementation of Principal Component Analysis on Masked and Non-masked Face Recognition", 2019 1st International Conference on Advances in Science Engineering and Robotics Technology (ICASERT), pp. 1-5, 2019.

[9] BOSHENG QIN and DONGXIAO LI, "Identifying Facemask- wearing Condition Using Image Super-Resolution with Classification Network to Prevent COVID- 19", May 2020, [online] Available: https://doi.org/10.21203/rs.3.rs-28668/v1+].

[10] "Face Mask Detection," kaggle.com. https://www.kaggle.com/ andrewmvd/face-mask-detection.

[11] Adnane Cabani, Karim Hammoudi, Halim Benhabiles, and Mahmoud Melkemi, "MaskedFace-Net — A dataset of correctly/incorrectly masked face images in the context of COVID-19" Smart Health, ISSN 2352–6483, Elsevier, 2020, DOI:10.1016/j.smhl.2020.100144

[12] Karim Hammoudi, Adnane Cabani, Halim Benhabiles, and Mahmoud Melkemi,"Validating the correct wearing of protection mask by taking a selfie: design of a mobile application "CheckYourMask" to limit the spread of COVID-19", CMES-Computer Modeling in Engineering Sciences, Vol.124, №3, pp. 1049–1059, 2020, DOI:10.32604/cmes.2020.011663.

[13] O. Cakiroglu, C. Ozer, and B. Gunsel, "Design of a deep face detector by mask r-cnn," in 2019 27th Signal Processing and Communications Applications Conference (SIU). IEEE, 2019, pp. 1–4.

[14] T. Meenpal, A. Balakrishnan, and A. Verma, "Facial mask detection using semantic segmentation," in 2019 4th International Conference on Computing, Communications and Security (ICCCS). IEEE, 2019, pp. 1–5.

[15] M. R. Bhuiyan, S. A. Khushbu, and M. S. Islam, "A deep learning based assistive system to classify covid-19 face mask for human safety with yolov3," in 2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT). IEEE, 2020, pp. 1–5.

[16] A. S. Joshi, S. S. Joshi, G. Kanahasabai, R. Kapil, and S. Gupta, "Deep learning framework to detect face masks from video footage," in 2020 12th International Conference on Computational Intelligence and Communication Networks (CICN). IEEE, 2020, pp. 435–440.

[17] Y. Wang, B. Luo, J. Shen, and M. Pantic, "Face mask extraction in video sequence," International Journal of Computer Vision, vol. 127, no. 6-7, pp. 625–641, 2019.

[18] A. Chavda, J. Dsouza, S. Badgujar, and A. Damani, "Multi-stage cnn architecture for face mask detection," arXiv preprint arXiv:2009.07627.

[19] S. Abbasi, H. Abdi and A. Ahmadi, "A Face-Mask Detection Approach based on YOLO Applied for a New Collected Dataset," 2021 26th International Computer Conference, Computer Society of Iran (CSICC), 2021, pp. 1-6, doi: 10.1109/CSICC52343.2021.9420599.

[20] S. Susanto, F. A. Putra, R. Analia and I. K. L. N. Suciningtyas, "The Face Mask Detection for Preventing the Spread of COVID-19 at Politeknik Negeri Batam," 2020 3rd International Conference on Applied Engineering (ICAE), 2020, pp. 1-5, doi: 10.1109/ICAE50557.2020.9350556.

[21] W. Vijitkunsawat and P. Chantngarm, "Study of the Performance of Machine Learning Algorithms for Face Mask Detection," 2020 - 5th International Conference on Information Technology (InCIT), 2020, pp. 39-43, doi: 10.1109/InCIT50588.2020.9310963.

[22] G. Deore, R. Bodhula, V. Udpikar and V. More, "Study of masked face detection approach in video analytics," 2016 Conference on Advances in Signal Processing (CASP), 2016, pp. 196-200, doi: 10.1109/CASP.2016.7746164.

[23] X. Peng, H. Zhuang, G. -B. Huang, H. Li and Z. Lin, "Robust Real-time Face Tracking for People Wearing Face Masks," 2020 16th International Conference on Control, Automation, Robotics and Vision (ICARCV), 2020, pp. 779-783, doi: 10.1109/ICARCV50220.2020.9305356.

[24] Y. Meng, N. Liu, Z. Su, X. Wang and H. Wang, "RESEARCH ON REAL-TIME DETECTION METHOD OF FACE WEARING MASK WITH LARGE TRAFFIC BASED ON DEEP LEARNING," The 8th International Symposium on Test Automation Instrumentation (ISTAI 2020), 2020, pp. 121-126, doi: 10.1049/icp.2021.1338.

[25] S. Asif, Y. Wenhui, Y. Tao, S. Jinhai and K. Amjad, "Real Time Face Mask Detection System using Transfer Learning with Machine Learning Method in the Era of Covid-19 Pandemic," 2021 4th International Conference on Artificial Intelligence and Big Data (ICAIBD), 2021,

[26] M. S. Ejaz and M. R. Islam, "Masked Face Recognition Using Convolutional Neural Network," 2019 International Conference on Sustainable Technologies for Industry 4.0 (STI), 2019, pp. 1-6, doi: 10.1109/STI47673.2019.9068044.

[27] P. Hofer, M. Roland, P. Schwarz, M. Schwaighofer and R. Mayrhofer, "Importance of different facial parts for face detection networks," 2021 IEEE International Workshop on Biometrics and Forensics (IWBF), 2021, pp. 1-6, doi: 10.1109/IWBF50991.2021.9465087.

[28] R. K. Kodali and R. Dhanekula, "Face Mask Detection Using Deep Learning," 2021 International Conference on Computer Communication and Informatics (ICCCI), 2021, pp. 1-5, doi: 10.1109/ICCCI50826.2021.9402670.

[29] W. Jian and L. Lang, "Face mask detection based on Transfer learning and PP-YOLO," 2021 IEEE 2nd International Conference on Big Data, Artificial Intelligence and Internet of Things Engineering (ICBAIE), 2021, pp. 106-109, doi: 10.1109/ICBAIE52039.2021.9389953.

[30] J. Gathani and K. Shah, "Detecting Masked Faces using Region- based Convolutional Neural Network," 2020 IEEE 15th International Conference on Industrial and Information Systems (ICIIS), 2020, pp. 156-161, doi: 10.1109/ICIIS51140.2020.9342737.

[31] J. Negi, K. Kumar, P. Chauhan and R. S. Rajput, "Deep Neural Architecture for Face mask Detection on Simulated Masked Face Dataset against Covid-19 Pandemic," 2021 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS), 2021, pp. 595-600, doi: 10.1109/ICCCIS51004.2021.9397196.

[32] R. Liu and Z. Ren, "Application of Yolo on Mask Detection Task," 2021 IEEE 13th International Conference on Computer Research and Development (ICCRD), 2021, pp. 130-136, doi: 10.1109/ICCRD51685.2021.9386366.

[33] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001, 2001, pp. I-I, doi: 10.1109/CVPR.2001.990517.

[34] "OpenCV: Cascade Classifier," Opencv.org. [Online]. Available: https://docs.opencv.org/3.4/db/d28/tutorial cascade classifier.html. [Accessed: 14-Dec-2021].

[35] R. Gradilla, "Multi-task cascaded convolutional networks (MTCNN) for face detection and facial landmark alignment," Medium, 27-Jul-2020. [Online]. Available: MTCNN on medium. [Accessed: 14-Dec-2021].

[36] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, "Joint face detection and alignment using multi-task cascaded convolutional networks," arXiv [cs.CV], 2016.

[37] C. Supeshala, "YOLO v4 or YOLO v5 or PP-YOLO? Which should I use?" Towards Data Science, 23-Aug-2020. [Online].

Available: https:// towardsdatascience.com/yolo-v4-or-yolo-v5-or-pp-yolo-dad8e40f7109. [Accessed: 14-Dec-2021].
[38] J. Solawetz, “Breaking down YOLOv4,” Roboflow Blog, 04-Jun-2020. [Online]. Available: https://blog.roboflow.com/ a-thorough-breakdown-of-yolov4/. [Accessed: 14-Dec-2021].
[39] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556, 2014.
[40] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 770–778, 2016.
[41] Xianzhi Du, Tsung-Yi Lin, Pengchong Jin, Golnaz Ghiasi, Mingxing Tan, Yin Cui, Quoc V Le, and Xiaodan Song. SpineNet: Learning scale-permuted backbone for recognition and localization. arXiv preprint arXiv:1912.05027, 2019.
[42] Mingxing Tan and Quoc V Le. EfficientNet: Rethinking model scaling for convolutional neural networks. In Proceedings of International Conference on Machine Learning (ICML), 2019.
[43] Chien-Yao Wang, Hong-Yuan Mark Liao, Yueh-Hua Wu, Ping-Yang Chen, Jun-Wei Hsieh, and I-Hau Yeh. CSPNet: A new backbone that can enhance learning capability of cnn. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshop (CVPR Workshop), 2020.
[44] 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 (TPAMI), 37(9):1904–1916, 2015.
[45] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), 40(4):834–848, 2017.
[46] Songtao Liu, Di Huang, et al. Receptive field block net for accurate and fast object detection. In Proceedings of the European Conference on Computer Vision (ECCV), pages 385–400, 2018.
[47] Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon. CBAM: Convolutional block attention module. In Proceedings of the European Conference on Computer Vision (ECCV), pages 3–19, 2018.
[48] Tsung-Yi Lin, Piotr Doll´ar, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 2117–2125, 2017.
[49] Shu Liu, Lu Qi, Haifang Qin, Jianping Shi, and Jiaya Jia. Path aggregation network for instance segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 8759–8768, 2018.
[50] Golnaz Ghiasi, Tsung-Yi Lin, and Quoc V Le. NAS-FPN: Learning scalable feature pyramid architecture for object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 7036–7045, 2019.
[51] Mingxing Tan, Ruoming Pang, and Quoc V Le. Efficient-Det: Scalable and efficient object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2020.
[52] Songtao Liu, Di Huang, and YunhongWang. Learning spatial fusion for single-shot object detection. arXiv preprint arXiv:1911.09516, 2019.
[53] Qijie Zhao, Tao Sheng, Yongtao Wang, Zhi Tang, Ying Chen, Ling Cai, and Haibin Ling. M2det: A single-shot object detector based on multi-level feature pyramid network. In Proceedings of the AAAI Conference on Artificial Intelligence (AAAI), volume 33, pages 9259–9266, 2019.
[54] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In Advances in Neural Information Processing Systems (NIPS), pages 91–99, 2015.
[55] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. SSD: Single shot multibox detector. In Proceedings of the European Conference on Computer Vision (ECCV), pages 21–37, 2016.
[56] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 779–788, 2016.
[57] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Doll´ar. Focal loss for dense object detection. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), pages 2980–2988, 2017.
[58] Hei Law and Jia Deng. CornerNet: Detecting objects as paired keypoints. In Proceedings of the European Conference on Computer Vision (ECCV), pages 734–750, 2018.
[59] Kaiwen Duan, Song Bai, Lingxi Xie, Honggang Qi, Qingming Huang, and Qi Tian. CenterNet: Keypoint triplets for object detection. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), pages 6569–6578, 2019.
[60] Abdullah Rashwan, Agastya Kalra, and Pascal Poupart. Matrix Nets: A new deep architecture for object detection. In Proceedings of the IEEE International Conference on Computer Vision Workshop (ICCV Workshop), pages 0–0, 2019.
[61] Zhi Tian, Chunhua Shen, Hao Chen, and Tong He. FCOS: Fully convolutional one-stage object detection. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), pages 9627–9636, 2019.
[62] Krishna Kumar Singh, Hao Yu, Aron Sarmasi, Gautam Pradeep, and Yong Jae Lee. Hide-and-Seek: A data augmentation technique for weakly-supervised localization and beyond. arXiv preprint arXiv:1811.02545, 2018.
[63] Jifeng Dai, Yi Li, Kaiming He, and Jian Sun. R-FCN: Object detection via region-based fully convolutional networks. In Advances in Neural Information Processing Systems
[64] Kaiming He, Georgia Gkioxari, Piotr Doll´ar, and Ross Girshick. Mask R-CNN. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), pages 2961–2969, 2017.
[65] Ze Yang, Shaohui Liu, Han Hu, Liwei Wang, and Stephen Lin. RepPoints: Point set representation for object detection. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), pages 9657–9666, 2019
[66] Cartucho, “OpenLabeling/README.md at master Cartu- cho/openlabeling,” GitHub. [Online]. Available:https://github.com/Cartucho/OpenLabeling/blob/master/README.md. [Accessed: 14-Dec- 2021].
[67] T. Pariwat and P. Seresangtakul, “Multi-stroke Thai finger-spelling Sign language recognition system with deep learning,” MDPI, 04-Feb-2021. Available: https://www.mdpi.com/2073-8994/13/2/262/htm. [Accessed: 14-Dec-2021].

Authors' Profiles

Ademola Adesokan is a second-year master's student in Computer Science Department at the American University of Beirut. He received a bachelor's degree in computer science from Federal University Dutsin-ma.

His current research interest is in using machine learning models and information retrieval to solve health related issues, claim verification and crisis management problems. He is interested in travels, sports, and social development.

How to cite this paper: Ademola A. Adesokan, "Covid-19 Control: Face Mask Detection Using Deep Learning for Balanced and Unbalanced Dataset", International Journal of Intelligent Systems and Applications(IJISA), Vol.14, No.6, pp.50-62, 2022. DOI:10.5815/ijisa.2022.06.05