Since January 2020 Elsevier has created a COVID-19 resource centre with free information in English and Mandarin on the novel coronavirus COVID-19. The COVID-19 resource centre is hosted on Elsevier Connect, the company's public news and information website.

Elsevier hereby grants permission to make all its COVID-19-related research that is available on the COVID-19 resource centre - including this research content - immediately available in PubMed Central and other publicly funded repositories, such as the WHO COVID database with rights for unrestricted research re-use and analyses in any form or by any means with acknowledgement of the original source. These permissions are granted for free by Elsevier for as long as the COVID-19 resource centre remains active.
ViDMASK dataset for face mask detection with social distance measurement

Najmath Ottakath a,∗, Omar Elharrouss b, Noor Almaadeed a, Somaya Al-Maadeed a, Amr Mohamed a, Tamer Khattab b, Khalid Abualsaoud a

a Qatar University, College of Engineering, Department of Computer Science and Engineering, Qatar
b Qatar University, College of Engineering, Department of Electrical Engineering, Qatar

ARTICLE INFO

Keywords:
Social distancing
Mask detection
Mask Video dataset
YOLOv4
YOLOv4-tiny
YOLOv5
Faster Mask RCNN with Resnet backbone and FPN
YOOLOR

ABSTRACT

The COVID-19 outbreak has extenuated the need for a monitoring system that can monitor face mask adherence and social distancing with the use of AI. With the existing video surveillance systems as base, a deep learning model is proposed for mask detection and social distance measurement. State-of-the-art object detection and recognition models such as Mask RCNN, YOLOv4, YOLOv5, and YOLOR were trained for mask detection and evaluated on the existing datasets and on a newly proposed video mask detection dataset the ViDMASK. The obtained results achieved a comparatively high mean average precision of 92.4% for YOLOR. After mask detection, the distance between people’s faces is measured for high risk and low risk distance. Furthermore, the new large-scale mask dataset from videos named ViDMASK diversifies the subjects in terms of pose, environment, quality of image, and versatile subject characteristics, producing a challenging dataset. The tested models succeed in detecting the face masks with high performance on the existing dataset, MOXA. However, with the ViDMASK dataset, the performance of most models are less accurate because of the complexity of the dataset and the number of people in each scene. The link to ViDMask dataset and the base codes are available at https://github.com/ViDMask/VidMask-code.git.

1. Introduction

Abundant surveillance systems capture terabytes of data on videos, but there is a lack of inference from them, which are real-time and fast. Several emerging technologies are charted for social distance monitoring, crowd control, and mask detection. Thermal scanning, ultrasound, and wireless tracking are among a few that need dedicated hardware to run [1]. Computer vision can be added to existing technology, allowing for easier implementation and better visualization. Deep learning and machine learning models are widely used in this context to achieve efficient and autonomous results [2,3]. Real-time data can be generated from computer vision systems [4–6]. This can support the human resources that are already present to make informed decisions, avoiding mis-interpretations and overcoming the lack of efficiency in detection.

Deep learning models are widely used for object detection in computer vision systems, which are pre-trained to detect several common objects using large datasets [7]. Transfer learning, one of the most efficient ways to use these models for custom object detection, is used here. Mask detection is one of the most efficient ways to prevent COVID infection, entitling its need as essential. Therefore, there is a requirement to monitor people with and without masks. Social distancing comes as an added aid for prevention wherein the detected people in the image can be measured using the detection parameters like the bounding box.

The dataset needed to train any model must be diverse and suitable for real-time monitoring scenarios, crowds with masks, and the like. A dataset for masks is used here, where it has a diverse set of images in terms of race, with and without crowds, blurred and clear, modelling multiple instances of real situations from a surveillance point of view [8]. This dataset, the prime focus of this paper, was used to train and test several models that are fast and accurate, well suited for real-time object detection and recognition problems. Social distancing and mask detection have been implemented individually in several state-of-the-art publications. However, a pipeline for mask detection and social distancing is rare [9].

This paper evaluates an existing dataset for advanced deep learning model for mask detection using state of art deep learning models, YOLOR, YOLOv5, YOLOv4 and a hybrid model for Fast Mask RCNN with Resnet 101 backbone using existing dataset as well as a new large dataset. The proposed framework used and dataset acquisition method is illustrated in the following figure (see Fig. 1). Further a new
Main contributions of this paper are:

- A deep learning pipeline that uses cutting-edge deep learning models to identify mask wearers and alert social distancing.
- Experimentation of mask detection methods using transfer learning on state-of-art object detection deep learning models, YOLOv4, Faster RCNN with resnet-101 backbone with FPN, YOLOv4-tiny, YOLOv5 and YOLOR.
- Comparison of the performance of deep learning models on both datasets.
- A very large and challenging dataset created from more than 60 videos, that has more than 30,000 images of people not wearing face masks and wearing face masks in diverse environments.

This paper provides a background study into mask detection using deep learning models and social distancing in Section 2. Section 3 details the proposed method with the dataset used and a brief description of the VIDMASK dataset. Section 4 is a description of the experimental setup with evaluation metrics. Section 5 presents the results and compares them with the state of art literature. Section 6, will conclude the paper.

2. Related works

One of the most effective ways to prevent the COVID-19 infection is to wear masks [10]. Mask detection can be considered as an object detection and recognition problem. Existing pre-trained models can be leveraged for object detection and training for mask detection [11]. Several techniques have been proposed in the literature where the authors employ hybrid machine learning and deep learning models for mask detection. They have proved that real-time and fast detection can be achieved with a deep learning model alone. The dataset, being central to the deep learning models, was used for transfer learning on several state-of-the-art pre-trained deep learning models like YOLOv3, SSD, and the Faster R-CNN model using ResNet-50 as a backbone with FPN (Feature Pyramid Network) detection. They were evaluated on the mean average precision (mAP) scale for accuracy and frames inferred per second (FPS) for speed.

After the onset of COVID, mask detection has become a case study in several state-of-the-art [12–16]. Transfer learning on state-of-art models has been widely used for multiple stage detectors like SSD with mobile net as well as single-state detectors like YOLOV3 and inception net [12]. In [9], the authors used several object detection backbones to detect masks and perform social distance at different stages. However, a faster and more accurate model needs to be identified by leveraging the current deep learning models for real-time use. A dataset that has
the required diversity with a mixture of crowds and no crowds, with masks or no masks, is required here for training and testing. Several datasets are available to detect face masks [8,9] of which MOXA3k [8] was used in this paper for comparison. They can be enhanced with a larger number of images with people in different poses at different angles and expressions, which can be extracted from videos. This can, not only provide a better dataset but also improve model accuracy. Social distancing is achieved by maintaining a certain distance between two people [17–19]. Several techniques can be used to calculate the distance between two persons and count the number of people in a region of interest [1,9,20–23]. Deep learning has an essential impact on tools used for social distancing; several pre-trained models used for this purpose have shown its efficiency in social distance measurement as person detection can be achieved through pre-trained model [1,24–26]. Hardware based approaches for mobile crowd sensing using several schemes like smart phone applications and IoT (Internet of Things) devices were used to detect social distancing and crowd control [20,27–31]. P-dispersion problem was proposed in [32] that can solve the minimum number of persons in a place problem, however it covers only one entity of the social distancing that is number of persons in an area. A virtual queue was implemented to thwart crowd gathering at queues thereby maintaining social distance using machine learning model [33]. Mask RCNN was used to detect people in frame and then frame parameters were used to detect the social distance between them [34].

It can be noted that in state of art literature social distancing and mask detection are performed as individual tasks except that of in [9], a social distancing tool was used. However, it would be efficient if mask detection and social distancing were performed together in one pipeline which would accomplish both tasks at once. The proposed model performs a two steps process, one identifying the best fit open source model for the purpose of mask detection by performing transfer learning and then comparing with state of art literature, thereby exploring different deep learning models and their performance for mask detection.

The best trained weights were then used to measure social distancing to identity high risk and low risk individuals gaining a two-step safety through one model to prevent COVID19. This paper also introduces VIDMASK dataset, an image dataset created of videos with more than 60 videos which produces a more challenging set of data for mask detection. Produced here is analysis of the models with a part of the new dataset to evaluate its performance compared to MOXA3k dataset. Social distancing is further performed from the best of these by calculating the distance from the centroid of the bounding box coordinates of object detection.

3. Materials and methods

The proposed methodology uses deep learning methods for transfer learning along with state of art dataset. Introduced in this section is the new video dataset VIDMASK with its descriptions including the deep learning models used for transfer learning.

3.1. Proposed video dataset (VIDMASK)

VIDMASK contains videos consisting of images that are crowd and incidental images of people with and without masks. Five videos were used for experimentation from the 67 videos listed for training, testing, and validation for the initial experiment as stated. The images extracted from the videos after annotation were shuffled. The chosen dataset contains densely crowded scenes, blurred scenes, and less crowded scenes with a majority of mask-wearing people. 20,000 instances of masks were found in the images, along with 2500 non-mask-wearing people. The sources of the current setup were from YouTube and Pexel.com. The videos modelled a natural environment with people performing daily tasks or being interviewed, thereby containing natural poses and expressions.

The video dataset used has over 67 videos of varying lengths, from 1 min to 10 s. The videos consist of scenes from different locations and situations, with masked and non-masked people in varied poses, angles, and masks. The videos were clippings from YouTube channels, specifically TV channel clippings on the COVID-19 situation and interviews related to the COVID-19 pandemic and other situations that occurred during the COVID-19 pandemic, where masks had to be worn and people were in public view.

The video dataset contains street scenes, press conference scenes, marches, general crowds, and a typical busy day scene. The videos originate from various locations around the globe. The videos are of different resolutions, converted to frames; masked and non masked people’s faces are then labelled with a bounding box using cvat [35] and the Roboflow tool for reviewing and visualization. The bounding box coordinates are saved as .TXT as well as coco json files, as annotations for masked and non-masked people. Table 1, is the description of the dataset, which includes the number of masked and non-masked individuals with dataset profiles. Fig. 3 illustrates the annotated video mask dataset.

3.1.1. Dataset configuration

Each model has a dataset configured to its requirements in terms of annotation format and the image size required to train the model. The following are the dataset configurations for each model used.

(1) YOLOV4, YOLOV4-tiny, YOLOv5 and YOLOR are annotated in .TXT format, where mask and non mask bounding box coordinates are contained in a text file with their corresponding labels.

(2) Mask RCNN with FPN: Images are annotated in COCO JSON format, with bounding box information of two classes masks and no masks, mask wearing people with non-masked people. 

Fig. 2. Social distancing flowchart.
3.2. Deep learning models used for object detection and classification

Object detection models that are optimal for detection need to have a higher input network size for smaller object [36–38]. Multiple layers are needed to cover the increased size of input network for a higher receptive field. To detect multiple objects there is a need for different sizes in a single image [39]. The following are different models that accomplish some of these criteria and that are used in this setup.

3.2.1. YOLOV3

YOLOV3 is from the series of YOLO object detection models. YOLO detects multiple objects at a time. YOLOV3 consists of 53 convolutional layers, also called Darknet 53. For detection, it has 53 more layers, for a total of 106 layers [36]. Images are often resized in YOLO according to input network size to improve detection at various resolutions. The objectness score represents the probability of predicting the object inside the bounding box. Predicted probability to the intersection of the union, which is a measure of the predicted bounding box to the ground truth bounding box [8,39].

3.2.2. YOLOV4

YOLOV4 produces optimal speed and accuracy when compared to YOLOV3. It is a predictor with a backbone, a neck, dense predictions, and sparse prediction [39]. The neck enhances feature discriminability by methods like Feature Pyramid Network (FPN), Path Aggregation Network (PAN), and Receptive Field Block (RFB). The head handles the dense prediction using either Region Proposal Network (RPN), YOLO, or SSD. For the backbone, Darknet53 was tested to be the most superior model [39]. YOLOV3 is commonly used as the head of YOLOV4. To optimize training, data augmentation happens in the backbone. It is proven to be faster and more accurate than YOLOV3 with the MSCOCO dataset. The advantage of YOLOV4 is that it runs on a single GPU, enabling comparatively less computation [39].

3.2.3. YOLOV4-tiny

YOLOV4-tiny is a compressed version of YOLOV4 with its network size decreased, having a lower number of convolutional layers in the CSPDarknet backbone. The YOLO layers are reduced to three and the anchor boxes for prediction are also reduced, enabling faster detection [40].

Table 1

| Dataset  | Number of videos | Frames | Video format | Ann mask |
|----------|------------------|--------|--------------|----------|
| VIDMASK  | 60               | 10,000+| Mp4/avi      | 50,000+  |

3.2.4. MaskRCNN with FPN and Resnet-101 backbone

Mask R-CNN, FPN-R101 benchmarks have a modular architecture, the input goes through a CNN backbone to extract features, which are used to predict region proposals. Regional features and image features are used to predict bounding boxes [41]. The scalability of this model and region proposal feature enables accurate detection [42].

3.2.5. YOLOV5

YOLOV5 is a Python implementation of an improved version of YOLOV3, published in May 2020 by Glenn Jocher of Ultralytics LLC5 on GitHub6 [43]. It is an improved version of their well known YOLOV3 implementation for PyTorch7. It has a similar implementation to YOLOV4, where it incorporates several techniques like data augmentation and changes to activation functions with post processing to the YOLO architecture. It combines images for training and uses a self-adversarial training (SAT) claiming an accelerated inference [43, 44].

3.2.6. YOLOR

YOLOR [45] is an unified network that takes advantage of implicit and explicit knowledge together. The implicit knowledge identifies features of the deep layers. The explicit knowledge is attained from labelled data.

3.3. Social distancing

The distance between two objects can be calculated by finding the centroid of the bounding boxes of the objects recognized and then finding the euclidean distance between them. Bounding boxes are typically rectangular in shape resulting in four coordinates, Xmin, Ymin, Xmax and Ymax. Each representing the four corners of the bounding box through coordinate system as (Xmin,Ymax), (Xmax,Ymin), (Xmin,Ymin) and (Xmax,Ymax). The centroid (C_x, C_y) of the rectangle can be calculated by Eqs. (1) and (2).

\[
C_x = \frac{X_{\text{max}} - X_{\text{min}}}{2}
\]  

\[
C_y = \frac{Y_{\text{max}} + Y_{\text{min}}}{2}
\]
The dataset contains around 600 images from the Moxa3K dataset for training and testing different object detection models. The most efficient model is used for social distance measurement to identify high-risk individuals with a given threshold. The most efficient model is used for social distance measurement to identify high-risk individuals with a given threshold.

4.1. Datasets

The Moxa3K dataset is used for training and testing different object detection models. The dataset contains around 600 images from the Kaggle dataset of medical masks from China, Russia, and Italy taken at the time of the advent of the COVID-19 pandemic. The images are of large crowds containing a large number of people. Around 700 images are front-facing and side-facing profiles of people, taken in a close-up shot. The remaining images are of pictures taken from Google search images, which are mainly those of the Indian population in the current COVID-19 pandemic situation. A similar evaluation has also been performed on the proposed ViDMask dataset that includes different situations composed of people with and without masks.

4.2. Deep learning model setup

**YOLOV4:** The pre-trained model used has a backbone which is Darknet3 that has 29 convolutional layers $3 \times 3$, 725X725 receptive fields and 27.6 M parameters, the next stage has a Spacial Pyramid Pooling (SPP) and Path Aggregation Network (PAN), with that YOLOV3 is used for dense prediction. Both the Bag of Specials (BoS) and the Bag of Freebies (BoF) are included for tasks such as data augmentation, mish activation, and so on.

YOLOV4 is setup to run on a single GPU, with CUDA 10.1 version running on the Tesla K80. A pre-trained model weight is used to custom train the images on classes, masks, and no masks. For optimal results, the number of classes is set as 2, with the maximum number of batches for training as the number of class*2000. The YOLOV4 training was run on 64 batches with 16 subdivisions. The network size was setup to 416x416 and the number of filters were defined by the Eq. (4). Here $\text{num\_classes}$ is the number of classes in the dataset that is two. Training loss, precision, recall and mAP are calculated during training. The mAP was calculated at every 100 iterations and consequently every 1000. FPS was calculated during inference on test images.

$$\text{filters} = (\text{num\_classes} + 5) \times 3.$$  

**YOLOV4-tiny:** YOLOV4 tiny was setup to run training on a single GPU, with CUDA 10.1 version running on Tesla K80. The dataset was setup to custom train on a pre-trained model. The configuration file is similar to YOLOV4, with two classes, where number of classes is set as 2 according to masks and no masks. The number of filters are determined from Eq. (4). 64 batches with 16 subdivisions with the network size setup to $416 \times 416$ was used for training. The performance metrics, training loss, precision, recall and mAP are measured at every 1000 iterations of the training. FPS is calculated based on time taken to infer on tested image.

**Mask RCNN with Resnet 101 backbone and FPN:** For object detection and classification, a fast RCNN with a feature Pyramid Network (FPN) and an R101 backbone is used. Mask RCNN with FPN is GPU-accelerated using CUDA 10.1 and the Pytorch framework. It uses COCO validation parameters for inference and evaluation. Pre-trained weights

```
Algorithm 1 Face mask detection and Social Distance measurement

Require: Box coordinates
Predict masked and non-masked faces. Input the bounding box coordinates for face. Xmin, Ymin, Xmax, Ymax coordinates for each individual.
Compute centroid $(C_x, C_y)$
Compute distance $(D)$
Compute D to m.
while Predict do
  if $D < 3m$ then
    High risk
  else
    Low risk
  end if
end while

$$C_y = \frac{Y_{\text{max}} - Y_{\text{min}}}{2}$$ (2)
The distance then can be calculated by finding the Pythagorean distance between the centroids, assuming two images, image1 with centroid coordinates $(C_{x1}, C_{y1})$ and image2 with centroid coordinates $(C_{x2}, C_{y2})$, the distance between two people $(D)$ can be calculated as in Eq. (3).

$$D = \sqrt{(C_{x2} - C_{x1})^2 + (C_{y2} - C_{y1})^2}$$ (3)

Fig. 4 Illustrates how social distancing is calculated for monitoring. The algorithm 1, below is a computational methodology for face-mask detection and social distance measurement.

4. Experimental setup

Mask detection and social distancing can be achieved through the use of transfer learning, enabling better and faster detection. A comparison of the latest and current deep learning models for mask detection is done through the evaluation metrics mAP and FPS, described in the following section. The most efficient model is used for social distance measurement to identify high-risk individuals with a given threshold of distance. Figs. 2 and 3 illustrate the flow chart of the experimental setup of the proposed methodology for mask detection and social distancing.

4.1. Datasets

The Moxa3K dataset is used for training and testing different object detection models. The dataset contains around 600 images from the

```
Fig. 4. Illustration of social distance measurement.
```
were used from a model zoo of Mask RCNN with FPN. The number of iterations in the training parameter was set to 1000, and the number of classes was set to two. The number of groups in each layer is 32, with a width per group of 8 and a depth of 101 for Resnet. The batch size per image is set to 16, which was used for the GPU computing capability of the training device. mAP, precision, recall, and AP were calculated during the training process. FPS was calculated during inference.

**YOLOV5:** YOLOV5 was setup to run on Torch 1.7 with the CUDA 10.1 version running on the Tesla V100. YOLOV5 was setup to train on custom object detection using transfer with pre-trained YOLOV5 weights. It was set to train for 300 iterations, with iterations decreased to achieve a better mAP. The number of classes was set to 2, mask, and no mask in the configuration file. In addition to mAP, precision, recall, and AP were plotted.

**YOLOR:** YOLOR was setup to run on Torch 1.7 with CUDA 10.1 version running on Tesla V100. A pretrained model was used to perform transfer learning. The model was trained for 10 epochs on VIDMASK dataset. The dataset was added noise and rotated to improve accuracy of detection. mAP, precision and recall were measured for the model.

### 4.3. Evaluation metrics

In order to have a fast and accurate model for real-time object detection, it is essential to measure the performance of the model and compare the model with certain metrics. mAP and FPS are used here to choose the optimal model for mask detection. Thus accuracy is measured in terms of detection and speed of detection in terms of Frames inferred per second. In order to identify the mAP, precision and recall has to be computed. The precision and recall are computed by identifying the True positive (TP), True negative (TN), False Positive (FP) and False Negative (FN). The True positive in general terms are referred for images that labelled true and the prediction produced true. The True negative are for images that are labelled false but predicted as false. False positive on the other hand are results that are labelled as false but predicted as false and False negative are the images that are labelled false but predicted as true. In the context of mask detection, due to multiple objects and multiple classes in one image itself, TP, FN, FP and TN were all measured for each detection based on the objectness of the detection resultant from the intersection of Union measurement which is as stated below.

**IoU: Intersection of Union:** The Intersection of Union (IoU), a commonly used evaluation metric which estimates regression quality. It is the IoU between a predicted bounding box and its corresponding assignment ground truth box [39]. The overlap of two boundaries is measured. The real object boundary is the ground truth to the predicted object boundary. In object detection models, True Positive (TP), False Positives (FP), False Negative (FN) and True Negative (TN) are defined by setting an IoU threshold. This paper uses a standard threshold of 50 percent match to classify the true positive or false positives in detection [41]. IoU which equals to or over 50% is set to True positive (TP) that is a correct detection, and IoU below 50% is False positive, a wrong detection. A FN is determined by counting an object not detected.

**Precision and Recall:** Precision is the percentage of the time your prediction is correct. Recall measures how well all the positive predictions are found. Eq. (5), and Eq. (6) are used to calculate precision and recall [41].

\[
\text{Precision} = \frac{TP}{TP + FP} \tag{5}
\]

\[
\text{Recall} = \frac{TP}{TP + FN} \tag{6}
\]

**Average Precision (AP):** Precision is the percentage of the time your prediction is correct. Recall measures how well all the positive predictions are found. Eq. (5), and Eq. (6) are used to calculate precision and recall [41,46]. The average precision can be calculated as the area enclosed under the Precision–Recall curve plotted on the coordinate axis. It is denoted as

\[
AP = \int_{0}^{1} (PR) \, dR
\]

In terms of object detection models the mean of the AP is calculated at different recall values.

**Mean Average Precision:** mAP, or mean Average Precision, is the metric used to determine performance on various object detection models. The classification and localization of the image are specifically determined [41]. The ground truth of object detection models, the class of the objects, and the bounding box of each object serve as parameters for calculating the mAP. The correctness of each image is determined by a bounding box through intersection of union (IoU), which is the ratio between the intersection and union of the predicted boxes and ground truth. A confidence threshold is kept, determining positive and negative boxes. From this, the mAP is calculated with a fixed confidence threshold and IoU value, typically 50% for Pascal VOC datasets. Different confidence thresholds are chosen in such a way that recall varies from 0 to 1, which makes AP, the mean of precision values at different recall values. Therefore, mAP is the average precision of all the classes in the dataset [39,41].

### 5. Evaluation and discussion

Within the experimental setup for each deep learning model, True positive (TP), False positive (FP), recall, precision, average precision, mAP, and FPS were plotted. Each model achieved object detection at different levels of efficiency. The models had varying precision and recall based on the type of images. With the requirements of optimal speed and accuracy for object detection in perspective, Tables 2 and 3 summarize the experimental results of the training of the models. Mask RCNN with FPN, achieves high precision and recall, with a 74.717% mAP at 50% IoU. YOLOV4 has higher average precision for masks and non-masks compared to other models. Table 3 compares the MOXA3k dataset to the state-of-the-art literature on 7 models for mask and non-mask detection. Fig. 8, illustrates the training results of YOLO with predictions of where images contain, single person, different densities of crowds facing different angles and maintaining diverse poses. The images further reinforces the superiority of one model over the other in terms of mask and non-mask detection.

The initial experimental setup included the evaluation of MOXA3k and ViDMASK with new and improved models. They were trained on YOLOV4, YOLOV4 tiny, YOLOv5, Mask RCNN with FPN. Identification of the optimum model for mask detection using the metrics mAP and FPS. With poor performance noted comparatively as seen in Table 4, a later and more accurate model according to the COCO evaluation parameters [45], YOLOR was experimented on. A pre-processing step was performed by adding noise and rotation for more diversity and better accuracy.

The following describes the training process and setup of each model trained.

#### 5.1. YOLOV4

The YOLOV4 was trained for a custom object using pre-trained weights; the mask dataset with two labelled classes, masks and no masks, with a resultant mAP of around 68% at 4000 iterations for MOXA3k as shown in Fig. 6. Fig. 6 is a chart depicting training loss, which is shown to decrease at multiple iterations. The x-axis shows the number of iterations the training process took and the y-axis is the loss which is shown as steadily decreasing by number of iterations. In addition, the mAP shows a steady rise with a few dips in between due to the different sizes used in training. The maximum mAP was found to be 68% which is slightly more than the state-of-the-art literature, which is 66.3% for YOLOV3. With Vidmask dataset, the mAP starts at 46.3%
Table 2
Precision, recall and mAP evaluation with MOXA3K.

| Method                  | Precision | Recall | mAP @50% IOU | Mask AP | Non-mask AP |
|-------------------------|-----------|--------|--------------|---------|-------------|
| YOLOV4 416 × 416        | 0.91      | 0.85   | 68.2%        | 76.31%  | 60.13%      |
| YOLOV4-tiny 416 × 416   | 0.73      | 0.61   | 59.06%       | 66.59%  | 50.50%      |
| YOLOV5 416 × 416        | 0.45      | 0.65   | 65.5%        | 73.3%   | 57.0%       |
| Mask RCNN with FPN 800 × 800 | 0.95  | 0.95   | 74.717%      | 33.729% | 33.445%     |

Table 3
Precision, recall and mAP evaluation with part of ViDMASK dataset.

| Method                  | Precision | Recall | mAP @50% IOU | Mask AP | Non-mask AP |
|-------------------------|-----------|--------|--------------|---------|-------------|
| YOLOV4 416 × 416        | 0.78      | 0.85   | 49.03%       | 84.77%  | 4.64%       |
| YOLOV4-tiny 416 × 416   | 0.83      | 0.84   | 51.42%       | 86%     | 5.83%       |
| YOLOV5 416 × 416        | 0.443     | 0.577  | 44.1%        | 84.5%   | 3.74%       |
| Mask RCNN and FPN 800 × 800 | 0.577 | 0.386   | 47.712%      | 53.153% | 2.255%      |
| YOLOR 1280 × 1280       | 0.84      | 0.917  | 92.4%        | 97.1%   | 86.6%       |

Fig. 5. mAP, precision and recall plot for YOLOV5 (ViDMask).

and keeps its steady at this accuracy rate and reaches 49.03% which is comparatively lower than other models. The training loss drops rapidly as shown in Fig. 7. Fig. 8(a) and Fig. 9(d) illustrate the inference of YOLOV4 on several types of images from the MOXA3k dataset. Table 2 shows the precision and recall at the end of 4000 iterations. Table 3 represents the results of the ViDmask dataset for 300 iterations.

5.1.1. YOLOV4-tiny
YOLOV4-tiny produced faster predictions compared to other models, with an FPS rate of 139.314. With this FPS rate, the speed of detection can be improved compared to other models for real-time detection. With the light architecture of YOLOV4-tiny and computational complexity reduced, this is used on computationally deficient devices like mobile phones, achieving acceptable mAP and high speed. Fig. 8(b): YOLOV4-tiny prediction and inference on MOXA3k dataset images with bounding box results.

5.2. Faster R-CNN model using ResNet101 as a backbone with FPN
Mask RCNN with FPN architecture proved to be far superior in terms of accuracy, with 74.717% for AP at 50% IoU. This technique, comparatively, has more accurate predictions than state-of-art literature and YOLOV4. However, Mask RCNN with FPN has a low FPS rate compared to other models, which can be a reason for the slow processing of the frames and is not adequate enough for real-time fast object detection. Fig. 8(c) and Fig. 9(c) illustrate the inference and prediction on several types of images from the Moxa3k and ViDmask datasets.

5.3. YOLOV5
YOLOV5 achieves an mAP of 65.5% with an inference time of 83.3 FPS. Fig. 5 illustrates the results of training as well as testing for YOLOV5. The results are a detail description of the training and testing results of YOLOV5 on ViDMask. The process of training, precision and recall are plotted on the graph for each iteration. The predicted bounding box and the loss are also plotted on the left hand side. The validation losses are also represented below. Further classification loss for training and validation is also displayed in the plot. Each showing a downward flow per increase in the number of iterations, inferring that the model improves with more iterations on the ViDmask dataset.

Fig. 8(d) and Fig. 9(a) illustrate the inference and prediction on several types of images in the MOXA3k and ViDmask datasets. Precision and recall plots increase with training time. YOLOV5 has a notably high FPS compared to similar architectures, achieving similar accuracy but higher FPS.
Fig. 6. YOLOV4, training results (MoXA3k).

Table 4
Comparison with state of art literature.

| Model                  | mAP@50-MOXA | mAP@50-VIDMASK | FPS  |
|------------------------|-------------|----------------|------|
| YOLOV3 414 × 414 [8]   | 63.99%      | –              | 21.2 |
| YOLOV3 608 × 608 [8]   | 66.84%      | –              | 10.9 |
| YOLOV3 832 × 832 [8]   | 61.73%      | –              | 6.9  |
| YOLOV3Tiny 414 × 414   | 56.27%      | –              | 138  |
| YOLOV3Tiny 608 × 608   | 55.08%      | –              | 72   |
| YOLOV3Tiny 832 × 832   | 56.52%      | –              | 46.5 |
| SSD 300 MobileNetv2    | 46.52%      | –              | 67.1 |
| F-RCNN 300 Inceptionv2| 60.5%       | –              | 14.8 |
| YOLOV4 416 × 416       | 68.22%      | 49.03%         | 46.6 |
| YOLOV4 Tiny 416 × 416  | 59.06%      | 51.42%         | 139.3|
| YOLOV5 416 × 416       | 65.9%       | 44.1%          | 83.3 |
| Mask RCNN with FPN 800 × 800 | 74.71% | 47.71%      | 14.7 |
| YOLOR 1280 × 1280      | –          | 92.4%          | 16.9 |

5.4. YOLOR

YOLOR achieves a higher mAP of 92.1% with an inference time of 16.8 ms FPS. Fig. 9(e) is the result of testing on the VIDMASK dataset. Fig. 9 illustrates the mAP, precision, and recall graphs. With a few epochs, a higher mAP was achieved compared to other models.

Comparing Precision, Recall and AP for mask and non-maskAP for the models on VidMask, the highest precision and recall is evident in YOLOR as seen in Table 3, it produced a balanced mask AP and non-maskAP. It should be noted that data augmentation in terms of rotation and noise was added on to the VidMask for YOLOR, where as resizing was only performed on the other models. In all the other models, the imbalance in dataset is clear in the non-mask AP results as they are very negligible compared to the mask AP. The impact of data augmentation is an added advantage to the high accuracy of YOLOR apart from its complex architecture.

5.5. Comparison with state of art literature

Table 3 illustrates the comparison of the state-of-the-art literature results with those of the models used in this paper. According to the most recent literature, and our experiments, all of the new and current models achieved nearly equal or better performance in terms of mAP score for detecting objects in the MOXA3K dataset and VIDMASK. YOLOV4, YOLOV5 and Mask RCNN with FPN achieved better performance in the accuracy of detection of masks and no masks compared to the state of the art literature for the MOXA3K dataset. However, the VIDMASK dataset was deficient in those models, but achieved high accuracy with YOLOR. In terms of FPS, YOLOV4-tiny is very close to YOLOV3-tiny, but better speed and accuracy are achieved for this model. With an mAP of 65.5% and an FPS rate of 83.3 for the MOXA3k dataset, YOLOV5 stands out in terms of optimal speed and accuracy for real-time object detection. Explicit and implicit learning enhanced
the detection in YOLOR by 92.4% for the ViDMask dataset. When compared to the two-stage detector Mask RCNN, which provided more accurate findings, one-shot detectors such as YOLOv5 and YOLOv4 produced faster inference. The two-stage detectors have more sophisticated models, which create inference delays, making them less preferable for real-time use. The tradeoff between speed and precision is indicated here. As can be observed, a larger dataset does not always yield better results; the issue lies in the quality of the data and the model utilized, which is appropriate for the model.

5.6. Social distancing

YOLOv5 was found to be the optimal model with its capacity for accuracy and speed of detection in terms of mAP and high FPS, which was then used for social distancing to classify high-risk and low-risk based on the pixel distance between them. The Figs. 10(a) and (b) illustrate the social distance labelled with yellow for high-risk individuals and with red for low-risk individuals. This achieves a two-step process using the same model, enhancing COVID-19 detection for social distancing with mask detection.

The line is an indication of distance between two individuals. Each individual can be part of both high risk and low risk based on the proximity. A first person can be at high risk towards a second person that is closer to 3 m, and a third person can be far away from a person who is consequently marked as low risk. Thus the indication is the risk distance between individuals. This method further infers that personal distance is required for social distancing and restricting the number of people in an area is required for proper social distancing. In addition, to the detrimental requirement of social distancing justified and identified. The social distance measurement performed as a pipeline reduces the overhead of the system as there is negligible change in FPS noted for social distancing.

6. Conclusion

An efficient system for social distancing and mask detection was chosen by experimental analysis and comparison to state-of-art literature with two datasets, the MOXA3K and the ViDMask dataset. Transfer learning was performed on five deep learning models. YOLOv5 was found to be efficient in terms of mAP and FPS for mask detection compared to state-of-art literature. The weights for this model are used for social distance measurement. The efficiency of social distance monitoring is thereby tagged to the efficiency of face mask detection, achieving intelligent mask detection and social distancing surveillance. A new, diverse and challenging video dataset, VIDMASK, containing more than 60 videos for face mask detection, was used for training and inference. This dataset produced high mean average precision for YOLOR with added noise and image rotation. This was a challenge for YOLOv4, YOLOv5 and Fast Mask RCNN with Resnet-101 backbone. VidMask can further be explored for mask detection and social distancing on the edge by compressing and quantizing the models or identifying lighter models with less computational complexity requiring only CPU usage. Besides the evident requirement for COVID situation, this can be further used for epidemics that require masks or environments that require proper social distancing and mask wearing such as laboratories that may have contaminants. Due to different camera angles used, it not only has a surveillance perspective, the model can be

![Fig. 7. YOLOv4, training results (ViDMASK).]
Fig. 8. Predicted results of the deep learning models with MOXA dataset.
Fig. 9. Obtained results on VIDMASK.
mounted on ground devices such as robots. This dataset can be further used for crowd counting applications for crowd that wear masks as some of the videos are of crowded scenes. Domain adaptation is a deep learning training process that can be used to adapt the model to specific domains thus can improve the model for diverse purposes requiring less computational complexity. In addition to the above, it can be used as a tool for social experiments where effectiveness of masks and social distancing can be confirmed.

CRediT authorship contribution statement

Najmath Ottakath: Data collection and annotation, Visualization, Curation, Software, Methodology, Writing – original draft. Omar Elharrouss: Data annotation, Visualization, Methodology, Software, Editing, Revision. Noor AlMaadeed: Revision, Editing and funding source. Somaya Al-Maadeed: Concept, Methodology, Visualization, Revision, Supervision. Amr Mohamed: Revision and funding source. Tamer
Khattab: Revision and funding source. Khalid Abualsaud: Revision, Editing and funding source.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement

This research work was made possible by research grant support (QUREX-CENG-SCDL-19/20-1) from Supreme Committee for Delivery and Legacy (SC) in Qatar.

References

[1] A.H. Ahamed, N. Zaini, M.F.A. Latip, Person detection for social distancing and safety violation alert based on segmented roi, in: 2020 10th IEEE International Conference on System Control, Computing and Engineering, ICSCE, IEEE, 2020, pp. 113–118.

[2] K. Gu, H. Liu, Z. Xia, J. Qiao, W. Lin, D. Thalmann, Pmn2.5 monitor: Use information abundance measurement and wide and deep learning, in: IEEE Transactions on Neural Networks and Learning Systems 32, 4278-4290wv Learn Syst, 2021, http://doi.org/10.1109/TNNLS.2021.3105394, 2021 Oct 32(10):4278-4290 Epub 2021 Oct 5. PMID: 34604093.

[3] K. Gu, Z. Xia, J. Qiao, W. Lin, Deep dual-channel neural network for image-based smoke detection, IEEE Trans. Multimed. 22 (2019) 311–323.

[4] O. Elharrouss, N. Almaadeed, S. Al-Maadeed, F. Khelifi, Pose-invariant face recognition with multitask cascade networks, in: Neural Computing and Applications, 2022, pp. 1–14.

[5] A. Riahi, O. Elharrouss, S. Al-Maadeed, BEMD-3DCNN-based method for face mask detection, in: 2021 6th International Conference on Computing, Communication, and Inventive Technology, ICSSIT, IEEE, 2020, pp. 499–504.

[6] N. Subramanian, O. Elharrouss, S. Al-Maadeed, M. Chowdhury, A review of deep learning-based detection methods for COVID-19, Comput. Biol. Med. (2022) 105188.

[7] C.Y. Wang, I.H. Yeh, I.H.Y. Liao, You only learn one representation: Unified Network for multiple tasks, 2021a, arXiv preprint arXiv:2105.04206.

[8] B. Roy, S. Nandy, B. Ghosh, D. Dutta, P. Biswas, T. Das, Moxa: A deep learning based unmanned approach for real-time mon- itoring of people wearing medical masks, Trans. Indian Natl. Acad. Eng. 5 (2020) 509–518.

[9] S. Toubolbi, S. Messaoud, M.A. Hajaji, A. Mulba, Real-time implementation of ai-based face mask detection and social distancing measuring system for covid-19 prevention, in: Scientific Programming 2021:1-Based Face Mask Detection and Social Distancing Measuring System for COVID-19 Prevention. Scientific Programming, 2021.

[10] J.W. Tang, T.J. Liebner, B.A. Craven, G.S. Settles, A schlieren optical study of the human cough with and without wearing masks for aerosol infection control, J. R. Soc. Interface 6 (2009) 5727–5736.

[11] M. Loery, G. Manogaran, M.H.N. Taha, N.E.M. Khalifa, A hybrid deep transfer learning model with machine learning methods for face mask detection in the era of the covid-19 pandemic, Mea- surement 167 (2021) 108288.

[12] G. Jignesh Chowdary, N.S. Punn, S.K. Sonbhadra, S. Agarwal, Face mask detection using transfer learning of inceptionv3, in: International Conference on Big Data Analytics, Springer, 2020, pp. 81–90.

[13] P. Nagrath, R. Jain, A. Madan, R. Arora, P. Kataria, J. Hemanth, Sudmmv2: A real time dius-based face mask detection system using single shot multibox detector and mobilenetv2, Sustainable Cities Soc. 66 (2021) 102692.

[14] T. Meenpal, A. Balakrishnan, A. Verma, Facial mask detection using se- man tic segmentation, in: 2019 4th International Conference on Computing, Communications and Security, ICCCS, IEEE, 2019, pp. 1–5.

[15] A. Chavda, J. Dsouza, S. Badgujar, A. Damani, Multi-stage cnn architecture for deep learning-based detection methods for COVID-19, Comput. Biol. Med. (2022) 105188.

[16] K. Paarporn, C. Eksi, J.S. Weitz, J.S. Shamma, Epidemic spread over networks with agent awareness and social distancing, in: 2015 53rd Annual Allerton Conference on Communication, Control, and Computing, Allerton, IEEE, 2015, pp. 51–57.

[17] Y. Sahraoui, C.A. Kerrache, A. Koriči, B. Nour, A. Adrane, R. Hussain, Deepidp: a deep-learning-based iov framework for real-time objects and distance violation detection, IEEE Internet Things Mag. 3 (2020) 30–34.

[18] M. Cristani, A. Del Bue, V. Murino, F. Vitiello, L. Vurli, The social visual distancing problem, Ieee Access 8 (2020) 126876–126886.

[19] M. Fazio, A. Buzachiz, A. Galletta, A. Celesti, M. Villari, A proximity-based indoor navigation system tackling the covid-19 social distancing measures, in: 2020 IEEE Symposium on Computers and Communications, ISCC, IEEE, 2020, pp. 1–6.

[20] R.J. Milne, C. Delea, L.A. Cotfas, C. Ioan, Evaluation of boarding modes adapted for social distancing when using apron busses, IEEE Access 8 (2020) 151650–151667.

[21] Y.C. Hou, M.Z. Baharuddin, S. Yusuf, S. Drumliify, Social distancing detection with deep learning model, in: 2020 8th International Conference on Information Technology and Multimedia, ICIMU, IEEE, 2020, pp. 334–338.

[22] I. Ahmed, M. Ahmad, J.J. Rodrigues, G. Jean, S. Din, A deep learning-based social distance monitoring framework for covid-19, Sustainable Cities Soc. 65 (2021) 102571.

[23] A. Ketkuri, B. Brik, An edge-based social distancing detect- tion service to mitigate covid-19 propagation, IEEE Internet Things Mag. 3 (2020) 35–39.

[24] M.E. Rusti, S. Yusuf, M. Ali, A.A.A. Hassan, Mydu: a smart social distancing monitoring system, in: 2020 6th International Conference on Information Technology and Multimedia, ICUIM, IEEE, 2020, pp. 399–403.

[25] V. Selamneni, A. Dave, S. Mondal, P. Mihailovic, P. Sahayta, Large area pressure sensor for smart floor sensor applica- tions—an occupancy limiting technology to combat social distancing, IEEE Consum. Electr. Mag. 10 (2020) 98–103.

[26] H. Wei, X.B. Chen, Flocking for multiple subgroups of multi-agents with different social distancing, Ieee Access 8 (2020) 164705–164716.

[27] J. Kudela, Social distancing as p-distersion problem, Ieee Access 8 (2020) 149402–149411.

[28] A.I. Kyriidis, M. Deriaz, A queue management approach for social distancing and contact tracing, in: 2020 Third International Conference on Artificial Intelligence for Industries, AI4I, IEEE, 2020, pp. 66–68.

[29] S. Gupta, R. Kapil, G. Kanahasabai, S.S. Joshi, A.S. Joshi, Sdm: a social distancing detector, in: 2020 12th International Conference on Computational Intelligence and Communication Networking Int: Work, CICN, IEEE, 2020, pp. 306–311.

[30] E. Vespasian, V. Saxena, S.K. Singh, Video annotation toolkit, in: 2018 International Conference on Advances in Computing, Communication Control and Networking, ICACCON, IEEE, 2018, pp. 911–914.

[31] R. Pakin, S. Pang, Q.V. L. Quang, T. Tran: Scalable and efficient object detection, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 10781–10790.

[32] J. Liao, Z. Fong, Z. Zhang, A. Liu, A survey of deep learning based object detection, Ieee Access 7 (2019).

[33] A. Bochkovskiy, C.Y. Wang, I.H.Y. Liao, Yolov4: Op- timal speed and accuracy of object detection, 2020, arXiv preprint arXiv:2004.10934.

[34] Zichong Jiang, Liang Zhao, Shuaiyong Li, Yanfeng Ji, Real-time object detection method based on improved YOLOv4-tiny, CoRR (2020) arXiv:2011.04244.

[35] M. Fazio, A. Buzachiz, A. Galletta, A. Celesti, M. Villari, A proximity-based indoor navigation system tackling the covid-19 social distancing measures, in: 2020 IEEE Symposium on Computers and Communications, ISCC, IEEE, 2020, pp. 1–6.

[36] Y. Wu, A. Kirillov, F. Massa, W.Y. Lo, R. Girshick, Detec- tron2, 2019.

[37] M. Cristani, A. Del Bue, V. Murino, F. Vitiello, L. Vurli, The social visual distancing problem, Ieee Access 8 (2020) 126876–126886.

[38] C.Y. Wang, I.H. Yeh, I.H.Y. Liao, You only learn one representation: Unified Network for multiple tasks, 2021a, arXiv preprint arXiv:2105.04206.