# A Deep Learning Based Light-Weight Face Mask Detector with Residual Context Attention and Gaussian Heatmap to Fight against COVID-19

FAN, Xinqi; JIANG, Mingjie; YAN, Hong

**Published in:**
IEEE Access

**Published: 01/01/2021**

**Document Version:**
Final Published version, also known as Publisher's PDF, Publisher's Final version or Version of Record



**Published version (DOI):**
10.1109/ACCESS.2021.3095191

**Publication details:**
FAN, X., JIANG, M., & YAN, H. (2021). A Deep Learning Based Light-Weight Face Mask Detector with Residual Context Attention and Gaussian Heatmap to Fight against COVID-19. *IEEE Access*, *9*, 96964-96974. [9475521]. https://doi.org/10.1109/ACCESS.2021.3095191

IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received May 31, 2021, accepted June 28, 2021, date of publication July 6, 2021, date of current version July 14, 2021.

*Digital Object Identifier 10.1109/ACCESS.2021.3095191*

# A Deep Learning Based Light-Weight Face Mask Detector With Residual Context Attention and Gaussian Heatmap to Fight Against COVID-19

**XINQI FAN, MINGJIE JIANG, AND HONG YAN, (Fellow, IEEE)**
Department of Electrical Engineering, City University of Hong Kong, Hong Kong, SAR, China

Corresponding author: Xinqi Fan (xinqi.fan@my.cityu.edu.hk)

This work was supported in part by the Innovation and Technology Commission of Hong Kong, and in part by the City University of Hong Kong under Project 7005230.

**ABSTRACT** Coronavirus disease 2019 has seriously affected the world. One major protective measure for individuals is to wear masks in public areas. Several regions applied a compulsory mask-wearing rule in public areas to prevent transmission of the virus. Few research studies have examined automatic face mask detection based on image analysis. In this paper, we propose a deep learning based single-shot light-weight face mask detector to meet the low computational requirements for embedded systems, as well as achieve high performance. To cope with the low feature extraction capability caused by the light-weight model, we propose two novel methods to enhance the model's feature extraction process. First, to extract rich context information and focus on crucial face mask related regions, we propose a novel residual context attention module. Second, to learn more discriminating features for faces with and without masks, we introduce a novel auxiliary task using synthesized Gaussian heat map regression. Ablation studies show that these methods can considerably boost the feature extraction ability and thus increase the final detection performance. Comparison with other models shows that the proposed model achieves state-of-the-art results on two public datasets, the AIZOO and Moxa3K face mask datasets. In particular, compared with another light-weight you only look once version 3 tiny model, the mean average precision of our model is 1.7% higher on the AIZOO dataset, and 10.47% higher on the Moxa3K dataset. Therefore, the proposed model has a high potential to contribute to public health care and fight against the coronavirus disease 2019 pandemic.

**INDEX TERMS** Face mask detection, residual context attention, synthesized Gaussian heat map regression, coronavirus disease 2019.

## I. INTRODUCTION

The World Health Organization (WHO) has stated that coronavirus disease 2019 (COVID-19) had infected over 160 million people and caused over 3.4 million deaths worldwide as of May 2021 [1]. Related large-scale respiratory diseases, severe acute respiratory syndrome (SARS) and Middle East respiratory syndrome (MERS), have occurred in the last two decades [2], [3]. SARS coronavirus 2 (SARS-CoV-2), the viral agent of COVID-19, has a higher reproductive number than SARS [4]. Increasing numbers of people are concerned about their health, and public health is a major priority of governments [5]. Various machine learning based methods have been applied in health care to assist the detection of COVID-19 cases from medical images [6]–[8]. One issue that limits machine learning methods for detecting COVID-19 cases is the lack of data. Fortunately, generative adversarial network based methods can be adopted to increase the size of datasets as in [9], [10].

For individuals, face masks could reduce the spread of coronaviruses by decreasing their emission in respiratory droplets [11]. N95 masks, medical masks, and homemade masks can block approximately 100%, 97%, and 95% of virus particles [12]. Currently, the WHO recommends that people should wear face masks if they have respiratory symptoms, or they are taking care of people with symptoms [13]. A recent study pointed out that most environments and contacts are under conditions of virus-limited where wearing face masks can effectively prevent virus spread [14]. Regions that had universal wearing of face masks have contributed more

The associate editor coordinating the review of this manuscript and approving it for publication was Huiyu Zhou.

to the control of COVID-19 than those without this requirement [15]. Many public service providers require customers to wear masks. However, some people still do not wear masks in public areas, which might lead to infection of themselves or others. Therefore, automatic detection of the wearing of face masks may help global society, but research related to this is limited.

The task of detecting face masks, or their being worn, refers to the localization of faces and judging whether masks are worn or not. Other recognition tasks relating to face masks include identifying their service stage [16] and efficiency [17], as these are useful to detect whether face masks can be re-used or their quality. These methods could play a complementary role with face mask detection algorithms to protect people from COVID-19. Face mask detection systems could be deployed in surveillance systems, internet of things systems, or smart cities to help public area managers ensure that all visitors are wearing masks, to reduce the risk of the spread of COVID-19. Face mask detection systems could take the place of workers who need to check the mask wearing status of visitors at supermarkets, universities, libraries, and similar locations.

Several studies have explored the detection of face masks. One approach is a two-step method which firstly detects faces using face detectors and then separately classifies whether a face mask is worn based on face mask classifiers [18], [19]. Although two-step methods may be sufficient in some scenarios, the operation of passing the results from the first step to the second step can degrade the speed significantly. End-to-end convolution neural network (CNN) based face mask detectors, which jointly detect faces and recognize face masks, may be more suitable for real-time face mask detection. A you only look once (YOLO) model with a residual network (ResNet) based face mask detector [20] can achieve high detection accuracy, but the network is heavy and not fast enough for edge devices. RetinaFaceMask proposed a light-weight version with MobileNet as its backbone, but it did not solve the problem of the light-weight model substantially decreasing the detection performance [21]. Other challenges in face mask detection come from the diversity of in-the-wild scenarios, which include, non-mask occlusion, various types of masks, different face orientations, and small or blurred faces (Fig. 1).

In this paper, we propose a novel single-shot light-weight face mask detector (SL-FMDet), which is able to detect face masks accurately and has a low hardware requirement. SL-FMDet uses a depthwise separable convolution based MobileNet as its backbone. It utilizes a feature pyramid network (FPN) to fuse high-level semantic information with low-level layers, and performs detection in multi-scale feature maps. However, FPN does not solve the problem that a light-weight model leads to worse feature extraction, so we propose two novel methods to achieve this. First, to extract rich context features and focus on crucial face mask related regions, we propose a novel residual context attention module (RCAM). Second, to learn more discriminating features for faces with and without masks, a novel auxiliary task is used to perform synthesized Gaussian heatmap regression (SGHR).


**FIGURE 1.** Challenges in face mask detection.

Evaluations of this study were performed on two publicly available face mask datasets, the AIZOO [22] and Moxa3K [23] face mask datasets. Experimental results showed that the proposed model achieved state-of-the-art results on both datasets. Compared with another light-weight model, YOLOv3-tiny, the mean average precision (mAP) of our model was 1.7% higher on the AIZOO dataset, and 10.47% higher on the Moxa3K dataset. The source code of our work is publicly available online.[1]

The rest of this paper is organized as follows. In Section II, we review related work on object detection, and face mask detection. The proposed methodology is presented in Section III. Section IV describes the datasets, implementation details, evaluation metrics, an ablation study, and quantitative and qualitative results. Section V concludes the paper and discusses future work.

## II. RELATED WORK

### A. OBJECT DETECTION

The Viola-Jones detector [24] achieves real-time detection of objects by an algorithm that extracts features using a Haar feature descriptor with an integral image method and a cascaded detector. It is still computationally expensive, even though it utilizes integral images to facilitate the algorithm. An effective feature extractor to detect humans, called histogram of oriented gradients (HOG), computes the directions and magnitudes of oriented gradients over image cells [25]. [26] detects object parts as a deformable part-based model and then connects them to judge classes that objects belong to.

Deep learning based detectors can perform well due to their robustness and high ability to extract features [27]. There are two popular categories, one- and two-stage object detectors. **One-stage detectors** directly regress the bounding boxes in a single step. The approach in YOLOv1 [28] divided the image into several cells and tried to find objects in each cell, but this was not good for small objects. YOLOv1 does not perform well by only using the last feature output, as the last feature map has a fixed receptive field and can only observe

[1] https://github.com/xinqi-fan/SL-FMDet

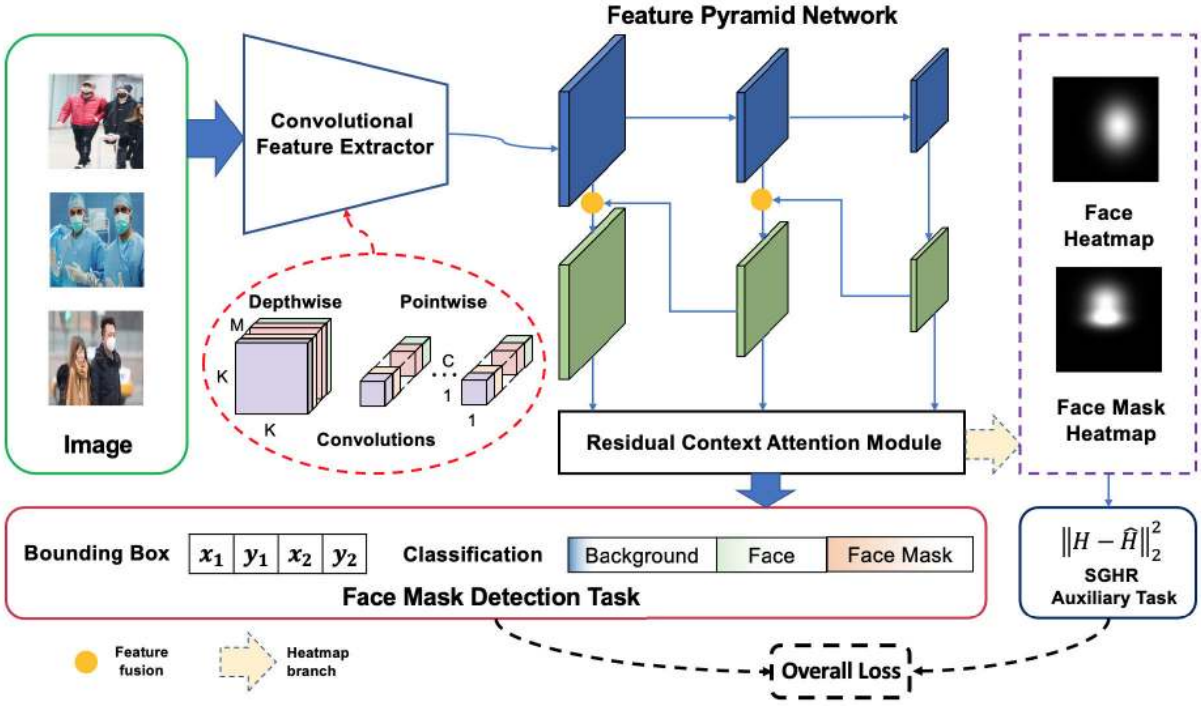



**FIGURE 2.** The pipeline of the proposed SL-FMDet. The backbone uses depthwise separable convolutions; FPN is used to fuse the high-level semantic information; RCAM can extract rich context information and focus on crucial face mask related regions; SGHR learns more discriminating features for faces with and without masks.

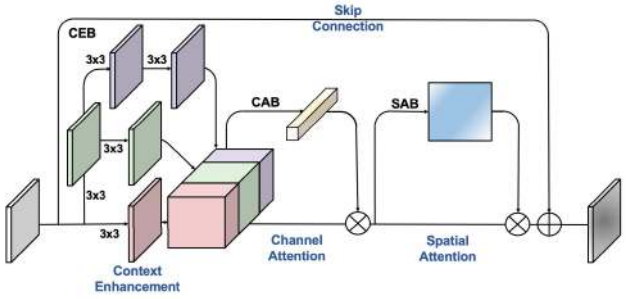
certain areas on the original images. Therefore, multi-scale detection was introduced into a single shot detector (SSD) to conduct detection on several feature maps and detect faces of different sizes [29]. To improve detection accuracy, Lin *et al.* [30] proposed RetinaNet by combining an SSD and an FPN architecture, which included a novel focal loss function to mitigate the class imbalance problem. In terms of the architecture, YOLOv2 has a similar improvement to SSDs using multi-scale features, and YOLOv3 is similar to RetinaNet by utilizing an FPN. **Two-stage detectors** generate region proposals in the first stage and then fine-tune these proposals in the second stage. The two-stage detector can provide high detection performance but at a low speed. Region-based CNN (R-CNN) [31] uses selective search to propose candidate regions that may contain objects. The proposals are fed into a CNN model to extract features, and a support vector machine (SVM) is used to recognize classes of objects. However, the second-stage of R-CNN is computationally expensive, since the network has to detect proposals in a one-by-one manner and uses a separate SVM for final classification. Fast R-CNN solved this problem by introducing a region of interest (ROI) pooling layer to input all proposed regions at once [32]. A region proposal network (RPN) introduced by faster R-CNN took the place of selective search, the speed limiting step of two-stage detectors [33]. Faster R-CNN integrated each detection component, region proposal, feature extractor, and detector into an end-to-end neural network architecture.

### B. FACE MASK DETECTION

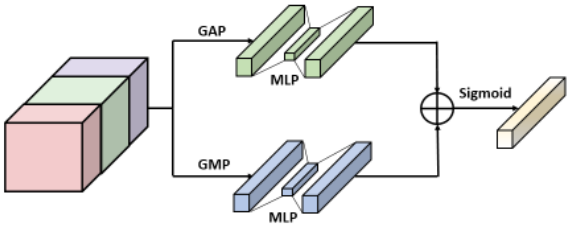
Face mask detection algorithms have become more topical recently, since masks can help control the spread of COVID-19 during the pandemic. The algorithmic task focuses only on detecting physical masks, as shown in [18], [20], [21], [23], [34], [35]. Among these, YOLO based models are the most popular detectors. ResNet based YOLOv2 was used by [20] to improve feature extraction for face mask detection. To enhance the robustness of detection by YOLOv3, an image mix-up and multi-scale method was utilized in [34]. A distance intersection over union non-maximum suppression (DIOU-NMS) algorithm was used to improve the post-processing stage of YOLOv3 [35]. YOLOv3 achieved the highest mAP in a comparison of YOLOv3, YOLOv3-tiny, SSD, and Faster R-CNN on the newly-established Moxa3K face mask detection dataset [23]. A person tracking system with a three-part face mask recognition system, a person detector, a tracker, and a face mask classifier, was developed to facilitate face mask detection applications in smart cities [36]. Face mask classification or recognition, assuming faces were detected, has also been studied [19], [37], [38].

## III. METHODOLOGY

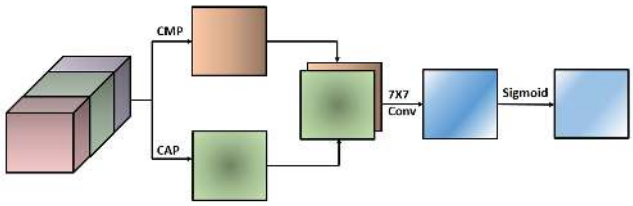
The overall pipeline of the proposed SL-FMDet is shown in Fig. 2. We first introduce the general architecture of the SL-FMDet, followed by two novel modules, RCAM and SGHR. Finally, we discuss the loss function, and the inference procedure.




(a) Overall architecture of RCAM, which consists of a CEB, a CAB and a SAB. The number denotes the kernel size of the convolution; $\oplus$: pixel-wise addition; $\otimes$: pixel-wise multiplication.




(b) The structure of the CAB. GAP: global average pooling; GMP: global maximum pooling.




(c) The structure of the SAB. CAP: channel average pooling; CMP: channel maximum pooling.

**FIGURE 3.** **Illustration of the RCAM. (a) Overall architecture of the RCAM. (b) The structure of the CAB. (c) The structure of the SAB.**

### A. NETWORK ARCHITECTURE

To reduce the size of the neural network, we propose to use a depthwise separable convolution network based backbone - MobileNet [39] that uses a depthwise convolution and a pointwise convolution in series to reduce the computational load. Assume the output shape of a standard convolution is $C \times H \times W$, and there are $C$ standard 2D convolution kernels of size $K \times K \times M$, the number of multiplications is $K \times K \times M \times C \times H \times W$. For a depthwise separable convolution, this is $(K \times K \times M \times 1 + 1 \times 1 \times M \times C) \times H \times W$, which is $\frac{1}{C} + \frac{1}{K^2}$ times smaller. Since the number of channels significantly influences the speed, we use the thinnest MobileNet, Mobilenet 0.25, with 0.25 times the number of channels of the ordinary MobileNet to make it smaller and have lower latency. Then, since each feature map corresponds to different receptive fields on the input images, we apply a multi-scale strategy to perform detection on three feature maps to find faces of different sizes. However, lower layers do not contain high-level semantic information, so we apply the FPN [40] to fuse high-level semantic information with lower layer feature maps. The size of the three feature maps used are $f_1 \in \mathbb{R}^{64\times80\times80}$, $f_2 \in \mathbb{R}^{64\times40\times40}$ and $f_3 \in \mathbb{R}^{64\times20\times20}$. We then generate two different size anchors on each feature map, and the details are given in section IV-B.

Although FPN can use high-level semantic information, it does not solve the problem caused by the separation of convolutions which reduces the capability of feature extraction. To cope with this problem, we propose two novel modules - RCAM, to focus on learning important information, in section III-B, and SGHR, to learn more discriminating features for faces with and without masks, in section III-C. RCAMs are directly applied to the fused feature maps from FPN. Then, we add a heatmap branch by performing a $1 \times 1$ convolution kernel on the output of RCAM to generate a one-channel map for SGHR. For the detection heads, we use a $1 \times 1$ convolutional kernel to form a $4 \times 2$ dimensioned bounding box of coordinates, and $n_c \times 2$ dimensioned classes, where the size 4 dimension is formed by the left corner $x_1$, $y_1$ and right corner $x_2$, $y_2$ coordinates, $n_c$ is the number of classes, and the size 2 dimension is formed by the two prior anchors of different sizes for each pixel.

### B. RESIDUAL CONTEXT ATTENTION MODULE

Compared with face detection, the task of face mask detection is more difficult, because it has to locate the face as well as distinguish faces with and without masks. To focus on face areas where masks may appear, we propose a novel RCAM (Fig. 3 (a)). RCAM contains three major blocks - a context enhancement block (CEB), a channel attention block (CAB), and a spatial attention block (SAB).

For the CEB, we form three parallel branches with $3 \times 3$, $5 \times 5$ and $7 \times 7$ receptive fields to enhance context information, similar to the context module in single-stage headless [41]. To reduce the number of parameters while maintaining the same receptive field size, all branches are implemented by $3 \times 3$ convolution kernels. The branch with a $5 \times 5$ receptive field is implemented by two consecutive $3 \times 3$ convolution kernels, and that with a $7 \times 7$ receptive field is realized by three

consecutive $3 \times 3$ convolution kernels. We then concatenate all feature maps from the branches to form an enhanced context feature map.

To focus on the important face mask related features, we cascade a convolutional block attention module (CBAM) [42] after the CEB, and add a skip connection. This attention module consists of a CAB (Fig. 3 (b)) and a SAB (Fig. 3 (c)). The CAB assigns the weights on each channel of the input features, while the SAB calculates a spatial attention map to focus on the specific part of the input feature. The computation of the CAB with input $f_c \in \mathbb{R}^{D\times H\times W}$ is

$$A_c = \sigma\Big(\mathrm{MLP}\big(\mathrm{GAP}(f_c)\big) + \mathrm{MLP}\big(\mathrm{GMP}(f_c)\big)\Big), \tag{1}$$

and that of SAB is

$$A_s = \sigma\Big(\mathrm{Conv2D}\Big(\mathrm{Concat}\big(\mathrm{CAP}(f_c), \mathrm{CMP}(f_c)\big)\Big)\Big), \tag{2}$$

where $A_c \in \mathbb{R}^D$ and $A_s \in \mathbb{R}^{H\times W}$ denote the channel and spatial attention; $\sigma$ is the sigmoid function to normalize the output to (0, 1); MLP refers to the multi-layer perceptron, which is a 3-layer fully connected network with the number of neurons of the intermediate layer ($D/8$); GAP and GMP stand for global average pooling and global maximum pooling; CAP and CMP stand for channel average pooling and channel maximum pooling; Conv2D represents 2 dimensional convolution; Concat is the channel concatenation operation. Finally, we add a skip connection to avoid information loss and gradient vanishing.

### C. SYNTHESIZED GAUSSIAN HEATMAP REGRESSION

Although the light-weight network is small and fast, it has a relatively weak feature extraction ability. To solve this problem, and enhance the feature learning of discriminating features for face areas with and without masks, we propose a novel auxiliary learning task as SGHR.

We consider an image containing $n_1$ bounding boxes of face masks and $n_2$ bounding boxes of faces. For the $n_1$ face mask bounding boxes, we first generate the face Gaussian heatmaps $H^m_{j1}, j \in \{1, \ldots, n_1\}$ as

$$H^m_{j1}(x, y) = \exp\left(-\frac{1}{2}\left(\frac{(x-c_{jx})^2}{\sigma^2_{jx}} + \frac{(y-c_{jy})^2}{\sigma^2_{jy}}\right)\right), \tag{3}$$

where $(c_{jx}, c_{jy})$ is the central position, $h_j$ and $w_j$ are the height and width of the $j$th face bounding box; $\sigma_{jx}$ and $\sigma_{jy}$ control the radii of the corresponding heatmaps, and $\sigma_{jx} = h_j/6$, $\sigma_{jy} = w_j/6$. Then, we generate the Gaussian heatmaps for masks as,

$$H^m_{j2}(x, y) = \exp\left(-\frac{1}{2}\left(\frac{(x-\widehat{c}_{jx})^2}{\widehat{\sigma}^2_{jx}} + \frac{(y-\widehat{c}_{jy})^2}{\widehat{\sigma}^2_{jy}}\right)\right), \tag{4}$$

where $(\widehat{c}_{jx}, \widehat{c}_{jy})$ is the estimated central position of face mask $j$, which is calculated by $\widehat{c}_{jx} = c_{jx} + h_j/4, \widehat{c}_{jy} = c_{jy}. \widehat{\sigma}_{jx} = h_j/12$, $\widehat{\sigma}_{jy} = w_j/6$. Then we sum $H^m_{j1}$ and $H^m_{j2}$ to obtain the Gaussian heatmap for face masks,

$$H^m_j = H^m_{j1} + H^m_{j2}. \tag{5}$$

For the $n_2$ bounding boxes for faces without masks, their heatmaps only contain single face Gaussian heatmaps $H^f_i$, $i \in \{1, \ldots, n_2\}$, which is the same as the calculation in (3). Finally, we sum the face mask and face heatmaps and suppress the maximum value to obtain the final synthesized Gaussian heatmaps (SGHs) as

$$H = \sum_{i=1}^{n_1} H^f_i + \sum_{j=1}^{n_2} H^m_j \tag{6}$$

$$H \leftarrow \mathrm{clip}(H, 1), \tag{7}$$

where clip $(H, 1)$ is to avoid the maximum of $H$ exceeding 1. An example for computing an SGH is shown in Fig. 4.

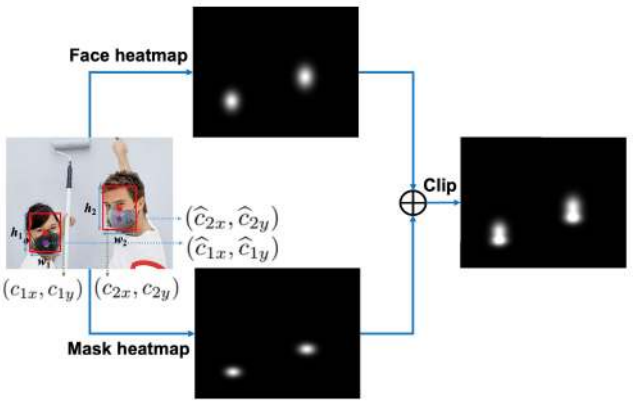



**FIGURE 4.** An example of computation of SGH. ⊕: pixel-wise addition.

The objective of SGHR is to predict heatmaps as close as possible to ground truth SGHs. Thus, an $l_2$ loss performs regression between the predictive heatmap $\widehat{H}$ and the ground truth heatmap $H$ as

$$\mathcal{L}_h(\widehat{H}, H) = \|\widehat{H} - H\|^2_2. \tag{8}$$

### D. LOSS FUNCTION

The model gives three outputs for each input image, a localization offset prediction $\widehat{Y}_l \in \mathbb{R}^{p\times 4}$, a classification confidence prediction $\widehat{Y}_c \in \mathbb{R}^{p\times n_c}$, and a predictive heatmap $\widehat{H}$, where $p$ and $n_c$ denote the number of generated anchors and the number of classes. We also have the prior anchors $P \in \mathbb{R}^{p\times 4}$, the ground truth boxes $Y_l \in \mathbb{R}^{o\times 4}$ and the classification label $Y_c \in \mathbb{R}^{o\times 1}$, where $o$ refers to the number of objects. Before calculating losses, we match and decode anchors $P$ with the ground truth boxes $Y_l$ and the classification label $Y_c$ to obtain $P_{ml} \in \mathbb{R}^{p\times 4}$ and $P_{mc} \in \mathbb{R}^{p\times 1}$, where each row in $P_{ml}$ or $P_{mc}$ denotes the offsets or top classification label for each anchor, respectively. The positive localization prediction and class are defined as $\widehat{Y}^+_l \in \mathbb{R}^{p_+\times 4}$ and $\widehat{Y}^+_c \in \mathbb{R}^{p_+\times 1}$. The positive matched anchors' localization offsets and class are defined as $P^+_{ml} \in \mathbb{R}^{p_+\times 4}$ and $P^+_{mc} \in \mathbb{R}^{p_+\times 1}$, where $p_+$ denotes the number of anchors whose top classification label is not zero.

To be robust to outliers, we use the smooth $L1$ loss [33] to regress the localization offsets as

$$\mathcal{L}_l(\widehat{Y}_l^+, P_{ml}^+) = \text{Smooth}_{L1}(\widehat{Y}_l^+ - P_{ml}^+). \tag{9}$$

Hard negative mining [43] is performed to obtain sampled negative matched anchors and the corresponding predictions, $P_{mc}^- \in \mathbb{R}^{p_- \times 1}$ and $\widehat{Y}_c^- \in \mathbb{R}^{p_- \times 1}$, where $p_-$ is the number of sampled negative anchors. The classification loss is computed by positive and negative samples using cross-entropy (CE) as

$$\mathcal{L}_c(\widehat{Y}_c^+, \widehat{Y}_c^-, P_{mc}^+, P_{mc}^-) = CE(\widehat{Y}_c^+, P_{mc}^+) + CE(\widehat{Y}_c^-, P_{mc}^-). \tag{10}$$

Together with the heatmap loss $\mathcal{L}_h$ in (8), we derive the total loss as

$$\mathcal{L} = \frac{1}{N}(\mathcal{L}_c + \alpha\mathcal{L}_l) + \beta\mathcal{L}_h, \tag{11}$$

where $N$ is the number of matched default anchors and $\alpha$ and $\beta$ are hyperparameters to weight the losses.

### E. INFERENCE

In the inference stage, the model produces the object localization $L \in \mathbb{R}^{p\times 4}$ and object confidence $\widehat{Y}_c \in \mathbb{R}^{p\times 3}$. The second column of $\widehat{Y}_c$ is the confidence of faces, $\widehat{Y}_{cf} \in \mathbb{R}^{p\times 1}$, and the third column of $\widehat{Y}_c$ is the confidence of face masks, $\widehat{Y}_{cm} \in \mathbb{R}^{p\times 1}$. Then, we remove objects with confidence lower than $t_c$ and perform non maximum suppression (NMS) with a threshold $t_{nms}$ to produce the final localization and confidence of faces $L_f' \in \mathbb{R}^{n_f\times 4}$, $\widehat{Y}_{cf}' \in \mathbb{R}^{n_f\times 1}$, and those of face masks $L_m' \in \mathbb{R}^{n_m\times 4}$, $\widehat{Y}_{cm}' \in \mathbb{R}^{n_m\times 1}$, where $n_f$ and $n_m$ denote the number of selected faces and masks.

## IV. EXPERIMENT AND RESULT

### A. DATASET

#### 1) AIZOO FACE MASK DETECTION DATASET

The AIZOO face mask detection dataset is a public open-source dataset created by AIZOOTech [22] that is integrated with approximately 8,000 images selected from the WIDER FACE [44] and MAsked FAces (MAFA) [45] datasets, and re-annotated to fit the face mask detection context. To cover more real-world conditions, most normal faces came from WIDER FACE (50%), while faces wearing masks were from MAFA (50%), giving the dataset a good balance among different scenarios. A subset of 1,839 images was pre-defined for testing.

#### 2) Moxa3K FACE MASK DETECTION DATASET

The Moxa3K face mask detection dataset is a public dataset to facilitate face mask research [23]. It contains 3,000 images with 2,800 for training and 200 for testing. The dataset was constructed by combining images from a Kaggle dataset and Internet images. The disadvantage of the dataset is that it contains only a few faces without masks.

### B. IMPLEMENTATION DETAIL

In the experiments, we employed an adaptive moment (Adam) optimizer with an initial learning rate of $\alpha_{LR} = 10^{-3}$. A reducing on plateau LearningRateScheduler was used to dynamically reduce the learning rate by a power of 10, if there was no change in the validation loss over 20 epochs. The hyperparameters of loss were: $\alpha = 2$ and $\beta = 10^{-3}$. The network was initialized by weights pre-trained on ImageNet. The models were trained on an NVIDIA GeForce RTX 2080 Ti and an Intel Xeon Silver 4108. The algorithm was developed with the PyTorch [46] deep learning framework. Each experiment operated for $n_{ep} = 250$ epochs with batch size $m = 32$. The threshold of NMS was $t_{nms} = 0.3$. The number of anchors, coordinates of the anchors' centers and anchor sizes are given in Table 1. The details of the training of our models are shown in Algorithm 1, where MiniBatchSampler refers to the operation of randomly selecting $m$ pairs of samples from dataset $\mathcal{D}$, denoted as $\mathcal{B}$; DataAugmentation is the data augmentation operation including random image cropping, distorting and flipping; Preprocess resizes the image into 640 × 640 pixels and normalizes the pixel values by subtracting the mean red green blue (RGB) values.

**Algorithm 1** Details of the Training Procedure

**Require:** Training set $\mathcal{D}_{\text{train}} = \{(x_i, y_i)\}_{i=1}^{n}$; Validation set $\mathcal{D}_{\text{val}} = \{(x_i, y_i)\}_{i=1}^{n'}$; A parameterized model $f_\theta$; ImageNet pretrained weights $\theta'$; Number of epoch $n_{ep}$; Batch size $m$; Learning rate $\alpha_{LR}$; Loss hyperparamters $\alpha$, $\beta$; Minimal valitation loss $\mathcal{L}_{\min} = +\infty$

**Ensure:** A parameterized model after training $f_{\theta''}$

```
Initialize the model parameters θ by θ = θ'.
for i = 1 to n_ep do
    for j = 1 to ⌊n/m⌋ do
        B ← MiniBatchSampler(D_train, m)
        B ← DataAugmentation(B)
        B ← Preprocess(B)
        L(f_θ) ← (1/m) Σ_{(x,y)∈B} L_{α,β}((x, y), f_θ)
        f_θ ← Adam(f_θ, L(f_θ), α_LR)
    L_val = (1/n') Σ_{(x,y)∈D_val} L_{α,β}((x, y), f_θ)
    α_LR ← LearningRateScheduler(α_LR, L_val)
    if L_val < L_min then
        f_θ'' = f_θ
```

**TABLE 1.** Settings for the generation of prior anchors.

| # of anchors | Coordinate of center | Anchor size |
|---|---|---|
| $80 \times 80 \times 2$ | $(4 + 8i, 4 + 8j) \quad i, j \in [\lvert 0, 79 \rvert]$ | 16, 32 |
| $40 \times 40 \times 2$ | $(8 + 16i, 8 + 16j) \quad i, j \in [\lvert 0, 39 \rvert]$ | 64, 128 |
| $20 \times 20 \times 2$ | $(16 + 32i, 16 + 32j) \quad i, j \in [\lvert 0, 19 \rvert]$ | 256, 512 |

### C. EVALUATION METRICS

For each class, average precision (AP) serves as a comprehensive indicator of the area under the precision and recall curve,


**FIGURE 5.** Visualization of spatial attention yielded by RCAM without (upper) and with (lower) SGH.

where the precision (P) and recall (R) are defined as [47],

$$\begin{aligned} \mathrm{P} &= \frac{\mathrm{TP}}{\mathrm{TP+FP}} = \frac{\mathrm{TP}}{\text{All Detection}}, \\ \mathrm{R} &= \frac{\mathrm{TP}}{\mathrm{TP+FN}} = \frac{\mathrm{TP}}{\text{All Ground Truth}}, \end{aligned} \tag{12}$$

where TP, FP and FN denote the true positive, false positive and false negative counts, respectively. The calculation of precision and recall is based on predictions ranked in descending order by their predicted confidence scores, which start from 0.02. As in the PASCAL VOC [48] new evaluation metrics, all point interpolation is used to smooth the zigzag precision and recall curve to obtain AP as,

$$\mathrm{AP} = \sum_{k=i}^{n} (\mathrm{R}_{k+1} - \mathrm{R}_k) \max_{\tilde{\mathrm{R}}:\tilde{\mathrm{R}} \geqslant \mathrm{R}_{k+1}} \mathrm{P}(\tilde{\mathrm{R}}). \tag{13}$$

We use $\mathrm{AP_F}$ and $\mathrm{AP_M}$ to denote APs for faces and face masks. mAP was used to evaluate the performance of the models [47] and can be calculated by taking the mean of AP against each class as,

$$\mathrm{mAP} = \frac{1}{n_c} \sum_{j=1}^{n_c} \mathrm{AP}_j, \tag{14}$$

where $n_c$ is the number of classes, and $\mathrm{AP}_j$ is the AP for $j$th class. We use the intersection over union (IOU) as 0.5 to judge the prediction, which is denoted as mAP@0.5 in the literature.

### D. ABLATION STUDY

To demonstrate the effectiveness of the proposed components, we performed ablation studies on RCAM, SGHR, and the position of the SGHR branch. The experiments based on the AIZOO dataset are summarized in Table 2 with details below.

#### 1) RCAM

We compared the detector without and with RCAM attached to the outputs of the FPN feature maps. By using RCAM, there was a 0.7% increase in the AP for faces, a 1.8% increase in the AP for face masks, and a 1.2% increase in mAP.

**TABLE 2.** Ablation study of the proposed model (%).

| RCAM | SGH | Position | $\mathrm{AP_F}$ | $\mathrm{AP_M}$ | mAP |
|---|---|---|---|---|---|
| ✗ | ✗ | - | 89.6 | 89.9 | 89.8 |
| ✓ | ✗ | - | 90.3 | 91.7 | 91.0 |
| ✓ | ✓ | 3 | 92.8 | 93.1 | 92.9 |
| ✓ | ✓ | 2 | 93.6 | 94.0 | 93.8 |
| ✓ | ✓ | 1 | 93.3 | 92.9 | 93.1 |

This demonstrated that the proposed RCAM may be able to enlarge and focus on useful context information for face mask detection.

#### 2) SGHR AND ITS POSITION

We added SGHR to the model to show the effectiveness of the SGHR auxiliary task and ran three experiments to find the best position for the SGHR branch. An auxiliary branch was placed on the output of RCAM at input feature $f_1$ from FPN or on the output of RCAM at input feature $f_2$ or on the output of RCAM at input feature $f_3$. These positions were denoted as 1, 2 and 3 for brevity. The highest AP and mAP were achieved by placing the SGH auxiliary task branch at feature $f_2$. This may be due to the $f_2$ feature maps having appropriate anchor scales for the majority of objects. Compared with the model without the SGHR branch, a maximum increase of 2.8% in mAP was observed, and the APs for each class also have an observable improvement.

### E. VISUALIZATION OF ATTENTION MAP

In the above ablation studies, SGHR enhanced the final face mask detection performance. In this section, we visualized the spatial attention of RCAM to qualitatively demonstrate how SGHR helps learn more discriminating features to distinguish between the object and the background. In Fig. 5, the first row is generated from the model without SGHR, while the second row used SGHR. The spatial attention maps generated by the model with SGHR could differentiate between the object and the background. This shows that the proposed SGHR auxiliary task can boost the performance of RCAM, and thus the overall detection performance.

(a) Non-mask occlusion.

(b) Various mask types.

(c) Different face orientation.

(d) Small or blurred faces.

**FIGURE 6. Qualitative results on AIZOO and Moxa3K datasets demonstrating the capability of our model on face mask detection challenges.**

### F. COMPARISON WITH OTHER MODELS ON AIZOO

The performance of our model on the AIZOO face mask dataset was compared with existing models used in face mask detection. The baseline model is a modified SSD with a light-weight backbone [22]. Faster R-CNN is the best regarded two-stage detector using an RPN [33]. YOLOv3 [49] and YOLOv3-tiny [49] are the most popular fast detectors used in the face mask literature [23], [34], [35]. YOLOv3 uses Darknet-53 as its backbone and three detection heads to process three-scale features enhanced by FPN. YOLOv3-tiny is a lighter and faster version of YOLOv3 with a light backbone and only two detection heads. RetinaFace is a high performance face detector using FPN to fuse high-level semantic information [50]. RetinaFaceMask is a dedicated face mask detector, and its light-weight version powered by MobileNet is denoted as RetinaFaceMask-M [21].

The mAP and APs of faces and face masks are given in Table 3. The proposed SL-FMDet achieved the highest mAP and APs among all the models. Compared with the baseline SSD model, SL-FMDet increased mAP by 3.0% and the APs of faces and face masks were improved by 4.0% and 2.1%, respectively. YOLOv3 and RetinaFace had the closest performance to our model, but they used heavy backbones, Darknet-53 and ResNet-50, which are computationally expensive. YOLOv3-tiny is a lighter version of YOLOv3, but its mAP was less than the proposed model by 1.7%. RetinaFaceMask-M is also a light-weight model, but it performed poorly at finding face masks with a low $AP_M$ of 90.4%.

**TABLE 3.** Comparison with other models on the AIZOO dataset (%).

| Model | $AP_F$ | $AP_M$ | mAP |
|---|---|---|---|
| **SSD** [22] | 89.6 | 91.9 | 90.8 |
| **Faster R-CNN** [33] | 83.3 | 83.7 | 83.5 |
| **YOLOv3-tiny** [49] | 91.0 | 93.2 | 92.1 |
| **YOLOv3** [49] | 92.6 | 93.7 | 93.1 |
| **RetinaFace** [50] | 92.8 | 93.1 | 93.0 |
| **RetinaFaceMask-M** [21] | 93.6 | 90.4 | 92.0 |
| **SL-FMDet** | **93.6** | **94.0** | **93.8** |

We demonstrate some qualitative results in Fig. 6. The model can successfully distinguish some confusing occlusions, such as occlusion by hands, hair or other objects (Fig. 6(a) and all diverse mask types were detected (Fig. 6(b). Side views of faces with masks could be detected (Fig. 6(c) and results on small and blurred faces are shown in Fig. 6(d).

### G. COMPARISON WITH OTHER MODELS ON Moxa3K

Experiments were also conducted on the Moxa3K face mask dataset, and the mAP and APs are summarized in Table 4. We compared our model with the best results reported by [23]. SL-FMDet achieved the state-of-the-art performance on Moxa3K, outperforming the previous best, YOLOv3. The light-weight model with RCAM and SGHR achieved better performance than heavy models like YOLOv3. YOLOv3-tiny is a popular light-weight model, so it provides another insight into our model's performance on the Moxa3K dataset. SL-FMDet's performance exceeded YOLOv3-tiny by 10.47% in terms of mAP. However, as the Moxa3K dataset was created for closed circuit television applications, it contains more blurred or small faces, which are hard to detect and result in overall low performance. In addition to the results reported by [23], we conducted experiments on RetinaFace and RetinaFaceMask-M, and these models give 1-2% lower performance than SL-FMDet in terms of mAP. In Fig. 6(d), SF-FMDet can find most of these blurred or small faces in the wild. Although there are some failure cases, due to occlusions by people or objects, the result seems satisfactory.

**TABLE 4.** Comparison with other models on the Moxa3K dataset (%).

| Model | $AP_F$ | $AP_M$ | mAP |
|---|---|---|---|
| **SSD** [29] | - | - | 46.52 |
| **Faster R-CNN** [33] | - | - | 60.50 |
| **YOLOv3-tiny** [49] | - | - | 56.57 |
| **YOLOv3** [49] | - | - | 66.84 |
| **RetinaFace** [50] | 53.37 | 78.28 | 65.83 |
| **RetinaFaceMask-M** [21] | 54.21 | 75.64 | 64.93 |
| **SL-FMDet** | **55.56** | **78.52** | **67.04** |

**TABLE 5.** FLOPs and the number of parameters of different models.

| Model | FLOPs (G) | # of Params (M) |
|---|---|---|
| **SSD** [29] | 30.54 | 23.88 |
| **Faster R-CNN** [33] | 206.67 | 41.13 |
| **YOLOv3-tiny** [49] | 12.9 | 8.67 |
| **YOLOv3** [49] | 154.9 | 61.50 |
| **SL-FMDet** | **1.01** | **0.43** |

### H. COMPARISON WITH OTHER MODELS IN TERMS OF FLOPs AND THE NUMBER OF PARAMETERS

SL-FMDet requires the smallest number of floating point operations (FLOPs) and number of parameters (Params) of the methods examined (Table 5). SL-FMDet takes 1.01G FLOPs, and has 0.43M parameters, which is less than 10% of the requirement of YOLOv3-tiny.

## V. CONCLUSION

In this paper, we proposed a novel SL-FMDet, which is efficient and has low hardware requirements. To overcome the lower feature extraction capability caused by its light-weight backbone, we proposed RCAM and SGHR. RCAM can extract rich context information and focus on crucial face mask related areas. By using SGHR as an auxiliary task, the model is able to learn more discriminating features for faces with and without masks. The model with SGHR yielded a better attention map, which qualitatively supports the effectiveness of this auxiliary task. The proposed model achieved state-of-the-art results on two public face mask datasets, AIZOO and Moxa3K. Compared with another light-weight

model, YOLOv3-tiny, the mAP of our detector is 1.7% higher on AIZOO and 10.47% higher on Moxa3K. Experimentally, we have shown that light-weight models can achieve similar or even better performance than heavy models by using RCAM and SGHR. The qualitative results also show the model is capable of tackling the challenges present in face mask detection. Therefore, the proposed face mask detector has a high potential to contribute to public health care to control the spread of COVID-19. One drawback of the method is the extra computation required for generating heatmaps and, due to limitations of the datasets, the method cannot distinguish between correct and incorrect mask wearing.

In future work, we would like to build face mask detection datasets with no, correct and incorrect mask wearing states, or use a zero shot learning approach to make the model able to detect incorrect mask wearing states. New deep learning detectors may be used to further improve the performance. Recently, advanced work on anchor-free deep learning detectors, such as CenterNet [51] or CornerNet [52] has appeared. We believe anchor-free detectors operate more like how human beings detect objects than anchor-based methods such as our method. CenterNet first detects the center of the objects, and then regresses the coordinates of corners relative to the centers. DEtection TRansformer (DETR) a newly-proposed transformer-based deep learning detector [53] borrows advantages from language transformers to use patch-based sequential information, and shows the method does not require post processing. In addition, we will develop a real-world face mask detection system on high performance edge devices, and integrate it with the internet of things systems.

## REFERENCES

[1] *Coronavirus Disease 2019 Weekly Epidemiological Update—24 May 2021*, World Health Organization, Geneva, Switzerland, 2020.

[2] P. A. Rota, "Characterization of a novel coronavirus associated with severe acute respiratory syndrome," *Science*, vol. 300, no. 5624, pp. 1394–1399, May 2003.

[3] Z. A. Memish, A. I. Zumla, R. F. Al-Hakeem, A. A. Al-Rabeeah, and G. M. Stephens, "Family cluster of middle east respiratory syndrome coronavirus infections," *New England J. Med.*, vol. 368, no. 26, pp. 2487–2494, 2013.

[4] Y. Liu, A. A. Gayle, A. Wilder-Smith, and J. Rocklöv, "The reproductive number of COVID-19 is higher compared to SARS coronavirus," *J. Travel Med.*, vol. 27, no. 2, pp. 1–4, Mar. 2020.

[5] Y. Fang, Y. Nie, and M. Penny, "Transmission dynamics of the COVID-19 outbreak and effectiveness of government interventions: A data-driven analysis," *J. Med. Virol.*, vol. 92, no. 6, pp. 645–659, Jun. 2020.

[6] J. D. Arias-Londono, J. A. Gomez-Garcia, L. Moro-Velazquez, and J. I. Godino-Llorente, "Artificial intelligence applied to chest X-ray images for the automatic detection of COVID-19. A thoughtful evaluation approach," *IEEE Access*, vol. 8, pp. 226811–226827, 2020.

[7] S. Hu, Y. Gao, Z. Niu, Y. Jiang, L. Li, X. Xiao, M. Wang, E. F. Fang, W. Menpes-Smith, J. Xia, H. Ye, and G. Yang, "Weakly supervised deep learning for COVID-19 infection detection and classification from CT images," *IEEE Access*, vol. 8, pp. 118869–118883, 2020.

[8] Q. Liu, C. K. Leung, and P. Hu, "A two-dimensional sparse matrix profile DenseNet for COVID-19 diagnosis using chest CT images," *IEEE Access*, vol. 8, pp. 213718–213728, 2020.

[9] M. Loey, G. Manogaran, and N. E. M. Khalifa, "A deep transfer learning model with classical data augmentation and CGAN to detect COVID-19 from chest CT radiography digital images," *Neural Comput. Appl.*, pp. 1–13, Oct. 2020.

[10] M. Loey, F. Smarandache, and N. E. M. Khalifa, "Within the lack of chest COVID-19 X-ray dataset: A novel detection model based on GAN and deep transfer learning," *Symmetry*, vol. 12, no. 4, p. 651, Apr. 2020.

[11] N. H. L. Leung, D. K. W. Chu, E. Y. C. Shiu, K.-H. Chan, J. J. McDevitt, B. J. P. Hau, H.-L. Yen, Y. Li, D. K. M. Ip, J. S. M. Peiris, W.-H. Seto, G. M. Leung, D. K. Milton, and B. J. Cowling, "Respiratory virus shedding in exhaled breath and efficacy of face masks," *Nature Med.*, vol. 26, no. 5, pp. 676–680, May 2020.

[12] Q. Ma, H. Shan, H. Zhang, G. Li, R. Yang, and J. Chen, "Potential utilities of mask-wearing and instant hand hygiene for fighting SARS-CoV-2," *J. Med. Virol.*, vol. 92, no. 9, pp. 1567–1571, Sep. 2020.

[13] S. Feng, C. Shen, N. Xia, W. Song, M. Fan, and B. J. Cowling, "Rational use of face masks in the COVID-19 pandemic," *Lancet Respiratory Med.*, vol. 8, no. 5, pp. 434–436, May 2020.

[14] Y. Cheng, N. Ma, C. Witt, S. Rapp, P. S. Wild, M. O. Andreae, U. Pöschl, and H. Su, "Face masks effectively limit the probability of SARS-CoV-2 transmission," *Science*, vol. 372, no. 6549, pp. 1439–1443, Jun. 2021.

[15] V. C.-C. Cheng, S.-C. Wong, V. W.-M. Chuang, S. Y.-C. So, J. H.-K. Chen, S. Sridhar, K. K.-W. To, J. F.-W. Chan, I. F.-N. Hung, P.-L. Ho, and K.-Y. Yuen, "The role of community-wide wearing of face mask for control of coronavirus disease 2019 (COVID-19) epidemic due to SARS-CoV-2," *J. Infection*, vol. 81, no. 1, pp. 107–114, Jul. 2020.

[16] Y. Chen, M. Hu, C. Hua, G. Zhai, J. Zhang, Q. Li, and S. X. Yang, "Face mask assistant: Detection of face mask service stage based on mobile phone," *IEEE Sensors J.*, vol. 21, no. 9, pp. 11084–11093, May 2021.

[17] E. P. Fischer, M. C. Fischer, D. Grass, I. Henrion, W. S. Warren, and E. Westman, "Low-cost measurement of face mask efficacy for filtering expelled droplets during speech," *Sci. Adv.*, vol. 6, no. 36, Sep. 2020, Art. no. eabd3083.

[18] A. S. Joshi, S. S. Joshi, G. Kanahasabai, R. Kapil, and S. Gupta, "Deep learning framework to detect face masks from video footage," in *Proc. 12th Int. Conf. Comput. Intell. Commun. Netw. (CICN)*, Sep. 2020, pp. 435–440.

[19] M. Loey, G. Manogaran, M. H. N. Taha, and 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," *Measurement*, vol. 167, Jan. 2021, Art. no. 108288.

[20] M. Loey, G. Manogaran, M. H. N. Taha, and N. E. M. Khalifa, "Fighting against COVID-19: A novel deep learning model based on YOLO-v2 with ResNet-50 for medical face mask detection," *Sustain. Cities Soc.*, vol. 65, Feb. 2021, Art. no. 102600.

[21] M. Jiang, X. Fan, and H. Yan, "RetinaMask: A face mask detector," 2020, *arXiv:2005.03950*. [Online]. Available: http://arxiv.org/abs/2005.03950

[22] D. Chiang. (2020). *Detecting Faces and Determine Whether People are Wearing Mask*. [Online]. Available: https://github.com/AIZOOTech/FaceMaskDetection

[23] B. Roy, S. Nandy, D. Ghosh, D. Dutta, P. Biswas, and T. Das, "MOXA: A deep learning based unmanned approach for real-time monitoring of people wearing medical masks," *Trans. Indian Nat. Acad. Eng.*, vol. 5, no. 3, pp. 509–518, Sep. 2020.

[24] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Dec. 2001, pp. 1–9.

[25] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2005, pp. 886–893.

[26] P. Felzenszwalb, D. McAllester, and D. Ramanan, "A discriminatively trained, multiscale, deformable part model," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2008, pp. 1–8.

[27] Z. Zou, Z. Shi, Y. Guo, and J. Ye, "Object detection in 20 years: A survey," 2019, *arXiv:1905.05055*. [Online]. Available: http://arxiv.org/abs/1905.05055

[28] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 779–788.

[29] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "SSD: Single shot multibox detector," in *Proc. Eur. Conf. Comput. Vis.* Springer, 2016, pp. 21–37.

[30] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal loss for dense object detection," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2980–2988.

[31] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 580–587.

[32] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1440–1448.
[33] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 91–99.
[34] C. Li, J. Cao, and X. Zhang, "Robust deep learning method to detect face masks," in *Proc. 2nd Int. Conf. Artif. Intell. Adv. Manuf.*, Oct. 2020, pp. 74–77.
[35] X. Ren and X. Liu, "Mask wearing detection based on YOLOv3," *J. Phys., Conf. Ser.*, vol. 1678, no. 1, pp. 1–6, 2020.
[36] G. T. S. Draughon, P. Sun, and J. P. Lynch, "Implementation of a computer vision framework for tracking and visualizing face mask usage in urban environments," in *Proc. IEEE Int. Smart Cities Conf. (ISC2)*, Sep. 2020, pp. 1–8.
[37] P. Mohan, A. J. Paul, and A. Chirania, "A tiny CNN architecture for medical face mask detection for resource-constrained endpoints," 2020, *arXiv:2011.14858*. [Online]. Available: http://arxiv.org/abs/2011.14858
[38] M. M. Rahman, M. M. H. Manik, M. M. Islam, S. Mahmud, and J.-H. Kim, "An automated system to limit COVID-19 using facial mask detection in smart city network," in *Proc. IEEE Int. IoT, Electron. Mechatronics Conf. (IEMTRONICS)*, Sep. 2020, pp. 1–5.
[39] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "MobileNets: Efficient convolutional neural networks for mobile vision applications," 2017, *arXiv:1704.04861*. [Online]. Available: http://arxiv.org/abs/1704.04861
[40] T.-Y. Lin, P. Dollar, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2117–2125.
[41] M. Najibi, P. Samangouei, R. Chellappa, and L. S. Davis, "SSH: Single stage headless face detector," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 4875–4884.
[42] S. Woo, J. Park, J.-Y. Lee, and I. So Kweon, "CBAM: Convolutional block attention module," in *Proc. Eur. Conf. Comput. Vis.*, Sep. 2018, pp. 3–19.
[43] A. Shrivastava, A. Gupta, and R. Girshick, "Training region-based object detectors with online hard example mining," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 761–769.
[44] S. Yang, P. Luo, C. C. Loy, and X. Tang, "WIDER FACE: A face detection benchmark," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 5525–5533.
[45] S. Ge, J. Li, Q. Ye, and Z. Luo, "Detecting masked faces in the wild with LLE-CNNs," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 2682–2690.
[46] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, and A. Desmaison, "PyTorch: An imperative style, high-performance deep learning library," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 8024–8035.
[47] R. Padilla, S. L. Netto, and E. A. B. da Silva, "A survey on performance metrics for object-detection algorithms," in *Proc. Int. Conf. Syst., Signals Image Process. (IWSSIP)*, Jul. 2020, pp. 237–242.
[48] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The Pascal visual object classes (VOC) challenge," *Int. J. Comput. Vis.*, vol. 88, no. 2, pp. 303–338, Jun. 2010.
[49] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," 2018, *arXiv:1804.02767*. [Online]. Available: http://arxiv.org/abs/1804.02767
[50] J. Deng, J. Guo, E. Ververas, I. Kotsia, and S. Zafeiriou, "RetinaFace: Single-shot multi-level face localisation in the wild," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2020, pp. 5203–5212.
[51] K. Duan, S. Bai, L. Xie, H. Qi, Q. Huang, and Q. Tian, "CenterNet: Keypoint triplets for object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 6569–6578.
[52] H. Law and J. Deng, "CornerNet: Detecting objects as paired keypoints," in *Proc. Eur. Conf. Comput. Vis.*, Sep. 2018, pp. 734–750.
[53] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *Proc. Eur. Conf. Comput. Vis.* Springer, 2020, pp. 213–229.

**XINQI FAN** received the bachelor's degree in automation from Southwest University, and the master's degree in electrical and electronic engineering from The University of Western Australia. He is currently pursuing the Ph.D. degree with the City University of Hong Kong. His research interests include computer vision, machine learning, and deep learning.

**MINGJIE JIANG** received the B.Eng. degree from the Civil Aviation University of China, and the M.Sc. degree from McMaster University. He is currently pursuing the Ph.D. degree with the City University of Hong Kong. His research interests include medical image analysis, machine learning, and deep learning.

**HONG YAN** (Fellow, IEEE) received the Ph.D. degree from Yale University. He was a Professor of imaging science with The University of Sydney. He is currently a Chair Professor of computer engineering with the City University of Hong Kong. He has over 600 journal and conference publications in these areas. His research interests include image processing, pattern recognition, and bioinformatics. He is an IAPR Fellow. He received the 2016 Norbert Wiener Award from the IEEE SMC Society for contributions to image and biomolecular pattern recognition techniques.