Journal of Imaging Science and Technology® 66(1): 010403-1–010403-10, 2022.

Real-time Face Mask-Wearing Detection and Temperature Measurement based on a Deep Learning Model

Chun-Liang Tung
Department of Information Management, National Chin-Yi University of Technology, Taichung, Taiwan

Ching-Hsin Wang
Department of Leisure Industry Management, National Chin-Yi University of Technology, Taichung, Taiwan
Institute of Innovation and Circular Economy, Asia University, Taichung, Taichung 41354, Taiwan
E-mail: chwang@ncut.edu.tw

Yong-Lin Su
Department of Information Management, National Chin-Yi University of Technology, Taichung, Taiwan

Abstract. *In the context of the COVID-19 outbreak in a global scenario, mandatory mask-wearing and temperature control can effectively prevent its spread and realize self-protection. Therefore, real-time face-mask wearing and temperature measurement technology is of greater importance against the background of infectious disease prevention and control. The present study adopted MobileNet as the backbone of the single-stage RetinaFace framework for real-time face detection and mask-wearing detection. Moreover, the focal loss function of α dynamic value was adopted to avoid the class imbalance problem and improve the classification accuracy in the training stage. Regarding face temperature measurement technology, non-contact and uncooled temperature-sensitive elements were used for temperature measurement, but it was easily affected by environmental variables. Therefore, an SVR model was employed for temperature calibration with the constant temperature blackbody as reference. The alignment errors for the accuracy of face detection, mask wearing detection and temperature correction were 89.58%, 97.84% and 4.85%, respectively. The parameter quantity of the face mask wearing detection model reached 0.42 M, while the computation quantity arrived at 2.039 GFLOPs. The detection model proposed in this study combines real-time mask-wearing detection with face temperature measurement, which can help to quickly measure the body temperature and detect whether one wears face masks properly in the context of COVID-19, so as to reduce the risk of epidemic spread.* © *2022 Society for Imaging Science and Technology.*
[DOI: 10.2352/J.ImagingSci.Technol.2022.66.1.010403]

1. INTRODUCTION

Since the outbreak of COVID-19 in 2020, countries around the world have issued their epidemic prevention policies, of which wearing face masks and measuring body temperature are the most fundamental requirements. An epidemic prevention measure that is often employed at borders and places of mass gathering is non-contact body temperature measurement and detection of no masks on faces with real-time image analysis technology [1–3]. At present, technologies related to Machine Learning (ML) have been widely applied to issues, including autonomous driving and National Language Processing (NLP) [4, 5]. Additionally, ML is aimed at predictions in relation to natural language processing and time-series data, and it has also gradually developed the cooperation of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) to make predictions, such as the analysis and prediction of COVID-19 positive cases [6].

In the field of image recognition and object detection, artificial neural networks (ANNs) are a widely used computational model for object detection and parameter learning. Through data training and detection procedures of the model, the trained ANNs model can be effectively applied to data clustering [7], data classification [8], and object recognition and detection [9], etc. LeCun et al. [10] proposed ConvNet based on the ANNs, which is a multi-layer convolutional neural network (CNN) architecture, hoping to solve the problem of handwritten character recognition. Its main concepts were originated from Neocognitron, a multi-layer neural network architecture. Thereafter, Lecun et al. [11] continued their research within the architecture, and published LeNet-5, which is the foundation of the CNN. Currently, the CNN has been widely applied to the identification of check numbers. Hinton and Alex adopted deep learning and GPU-accelerated AlexNet [12] and participated in the ImageNet object recognition competition. They achieved good outcomes: the error rate seen at 15.3%, which was 10.8% lower than the second place, making deep learning attract much attention again. The use of CNN becomes the mainstream in the subsequent ImageNet object recognition contest. By 2020, the recognition error rate of CNN-based image identifiers has reduced to less than 5%.

Generally, a sliding window method is used to scan the images one by one and predict the possible region in which objects are located. However, this method requires a large amount of computation. Therefore, it has been replaced by cascading, weight sharing and region proposal to reduce the cost incurred by large amounts of computation. Deep

Received June 9, 2021; accepted for publication Sept. 14, 2021; published online Oct. 13, 2021. Associate Editor: Yung-Kuan Chan.
1062-3701/2022/66(1)/010403/10/$25.00

learning based generic object detection can be divided into two categories in terms of the framework: the two-stage framework, which generates region proposal first and makes a prediction later, and the single-stage framework, which only necessitates one CNN forward to obtain category results and positions. The computational basis of the two-stage framework lies in how to effectively carry out the selective search algorithm and make predictions based on search results. A classic method is R-CNN which mainly locates object candidate boxes with handcrafted features. The use of R-CNN needs to train three models. Moreover, it takes much time to execute the selective search algorithm, and CNN may repeatedly extract candidate boxes in the feature extraction stage, resulting in the waste of computing resources. Fast R-CNN [13] was proposed in 2015 to share feature maps via the region of interest pooling (RoI pooling for short) to run the feature extraction network only once. Faster R-CNN [14] was proposed for the first time to replace the selective search algorithm with Anchor, and the region proposal network was used to predict proposal objects, which makes a significant contribution to the subsequent single-stage framework. Mask R-CNN [15] was proposed to reduce the deviation of RoI pooling from the original image for Fast R-CNN and Faster R-CNN; instead, RoI align was used to directly output the floating-point arithmetic value and correspond the feature map to the original image, so as to avoid the deviation.

The single-stage framework first appeared in DetectorNet [16]. It regards object detection as a regression problem and changed the softmax layer of the CNN to a bounding box offset region for the first time. However, this method necessitates the object boxes of different sizes as samples for subsequent model training and more computation for inference. OverFeat [17] adopts the architecture of a fully connected network. Moreover, it has multiscale input, and integrates the bounding box regression and classifiers into the same network. However, the accuracy is lower than that of R-CNN in the same period. In 2017, the mean Average Precision (mAP) of YOLOv2 (You Only Look Once, Version 2) [18] in COCO dataset surpassed that of Faster R-CNN, and the speed was 10 times faster. SSD (Single Shot Detector) [19], another method based on the anchor in 2016, follows the idea of Faster R-CNN and improves YOLO; it proposed to replace Anchor with Prior Box. The image dataset provided by VOC2007 at that time showed that its operation speed and detection accuracy were superior to those of YOLO and Faster R-CNN. The detection methods based on the single-stage framework gradually surpassed those based on the two-stage framework in terms of speed and accuracy, and the demand for edge computation increases rapidly. Many object detection methods realize improvement based on YOLO and SSD.

Li [20] adopted the cascade CNN architecture of the two-stage framework, and used the multi-scale method to solve the problem of multi-scale face detection. MTCNN [21], which appeared in 2016, followed the cascade CNN architecture, but adopted multi-task learning to solve the problem that the region proposal network of the two-stage framework in the cascade CNN architecture is time-consuming. MTCNN uses an image pyramid to deal with the problem of multi-scale face detection, and it is able to realize real-time detection in terms of the overall execution efficiency, so relevant applications emerge one after another. In 2017, the single-stage headless (SSH) face detection framework [22] referred to SSD and adopted the feature pyramid hierarchy to solve the problem of scale invariance. Moreover, it proposed the context module based on the self-attention mechanism to improve its accuracy, which is five times faster than the framework based on image pyramid. Face Attention Network (FAN) [23], published in the same year as SSH, proposes to solve the problem that objects cannot be detected due to occlusion when applied. Instead, it adopts the feature pyramid network [24], for it can retain more underlying features than the feature pyramid hierarchy. For each layer of output, the feature is extracted via the self-attention CNN, after which the exponential activation function and the original output are used to calculate the dot product. In addition, an attention-loss function is added for optimization. Retina Face [25], which appeared in 2020, combines the single-stage prior box of SSD, scale-invariance of FPN, context module of SSH, and multi-task learning to improve the accuracy and instantaneity of identification. In addition to the above new methods, few are produced by finely tuning the generic object detection method. For example, YOLOv3 [26] can generate an appropriate anchor box aspect ratio through K-mean clustering to improve the accuracy of the detection box. In 2017, Face R-CNN directly adopted the framework of faster R-CNN and optimized the loss function to solve the imbalance between positive and negative samples [27].

There are many different face detection methods, and researchers should select one according to their specific needs, i.e. choice between timeliness and accuracy. For example, this study mainly aims to integrate deep learning-based face detection technology with the thermal imaging camera as the core technology of real-time human body temperature detection. To achieve the best detection accuracy and real-time detection, the environment variables that affect face detection and thermal imaging cameras need to be considered simultaneously, which can avoid the situation that faces are precisely detected whereas face temperature cannot be accurately measured. Therefore, this study uses the Retina Face model as the infrastructure and Mobile Net model as the backbone to detect the masks on faces and measure face temperature in real time.

2. MASK WEARING DETECTION AND FACE TEMPERATURE MEASUREMENT TECHNOLOGY

Based on the RetinaFace model with a single-stage framework, the present study proposes an object detection method, which includes face mask wearing detection and real-time face temperature measurement. The detection model in this study mainly consists of five modules, namely the feature pyramid network (FPN) module, prior box module, context module, multi-task loss module and

real-time temperature measurement module. FPN uses Mobile Net as the CNN backbone architecture to realize real-time detection, for it requires less computation.

2.1 *Feature Pyramid Network*

In the traditional CNN feature extraction procedure, the feature map is not scale invariant. Therefore, when CNN is trained, the training set will increase the diversity of different size image samples through the preprocessing procedure of automatic scaling and reduction to improve the detection rate of different size objects [28–30]. The face detection model MTCNN uses the feature extraction method of the image pyramid network to reduce 1/2 of the original image sequentially, after which the reduced images are fed into different layers of the image pyramid. Its main purpose lies in giving different receptive fields to the CNN layers of the image pyramid by rescaling original images to make it scale invariant. However, the convolutional computation procedure of the CNN layers still requires a large amount of computation in the image pyramid. Therefore, the feature pyramid network (FPN) method is proposed to reduce the amount of convolution computation. FPN is a feature extraction method for object detection. The difference between FPN and image pyramid network (IPN) is that FPN only employs a single CNN architecture and outputs feature maps of different layers $P = \{p_1, p_2, \ldots, p_n\}$. FPN divides the backbone of the CNN architecture (e.g. ResNet [31], VGG [32], MobileNet [33], and AlexNet [12]) into multiple stages in a certain order, and the feature map of each stage is saved $C = \{c_1, c_2, \ldots, c_n\}$. Thereafter, the feature map of the next stage is magnified twice, and added to the feature map of the current stage, which enables the feature map to be used for prediction. The feature extraction of FPN is calculated by Eq. (1) as follows:

$$P = (f_\theta^1 \circ f_\theta^2 \cdots f_\theta^T)(C), \tag{1}$$

where P represents the feature map that each layer exports from the image pyramid after FPN operation, C represents the exported feature maps of each stage in the backbone $\{c_1, c_2, \ldots, c_n\}$, and f_θ^t represents the tth feature connection block. FPN is able to keep the underlying features that are close to those of original pictures without losing the features of each layer. In comparison with the IPN, FPN has better performance in terms of the recognition rate and efficiency. Therefore, a number of CNN-based detection models adopt FPN feature extraction methods, e.g. YOLOv3 [26] and RetinaNet [34].

To achieve the instantaneity of temperature measurement, MobileNet was chosen as the basis of the FPN backbone architecture. Each channel experienced depthwise convolution, after which the pointwise convolution with a kernel size of 1×1 to get the results close to the traditional convolution operation in a faster manner. Due to the difference in the size and number of kernels, there is a gap of $(1/N) + 1/(k \times k)$ in the speed between MobileNet Backbone and traditional convolution, wherein N indicates the number of filters when the kernel size is $k \times k$. In this study, FPN adopts MobileNet as the backbone, for its low computational complexity. Its architecture can be divided into three stages, so three feature maps that correspond to three stages $C = \{c_1, c_2, c_3\}$ can be obtained. In the convolution procedure of each stage, the strides of the receptive fields are {8,16,32}. The exported feature map of the FPN is $P = \{p_1, p_2, p_3\}$. MobileNet Backbone does not require a large amount of computation such that the feature map for each stage is reserved as the input of the context module.

2.2 *Context Module and Prior Box*

The context module in this study is a continuation of the three feature maps generated by FPN $\{p_1, p_2, p_3\}$, which corresponds to the computation of their respective context modules. The receptive fields of three different sizes with a kernel size of $\{(3 \times 3), (5 \times 5), (7 \times 7)\}$ are adopted. Lastly, they are concatenated to get three feature maps $\{m_1, m_2, m_3\}$. They are the feature maps of different scales with a shape size of $c_{m_i} \times h_{m_i} \times w_{m_i}, i = 1, 2, 3$, wherein c_{m_i} is the number of their channels, h_{m_i} is their height, and w_{m_i} is their width. Prior Box, which was first proposed in SSD, is an improved prior box generator based on the anchor of YOLO and Faster R-CNN. It can quickly generate a fixed number of prior boxes, which correspond to the feature maps of the bounding box head to realize rapid detection and classification. It is significantly faster in comparison with Faster Region-based Convolutional Network (Faster R-CNN) that employs the two-stage framework, i.e. prediction of the possible object location through the region proposals network and RoI pooling of the object location. The detection model based on the single-stage framework, such as YOLO [35] and SSD [19], can realize real-time detection because the position of the prior box for the object has been generated in advance, and the offset between the object box and the prior box as well as the category of the object can be predicted simultaneously within one CNN forward pass. In this study, the total number of prior boxes N_{pbox} is determined by the size of the three feature maps, which has been computed by the context module. The size of the originally input image affects the shape size of the feature map. The total number of prior boxes is computed via Eq. (2) below:

$$N_{\text{pbox}} = \sum_i num_{\text{pbox}} \times h_{m_i} \times w_{m_i}, i = 1, 2, 3, \tag{2}$$

where num_{pbox} is the number of prior boxes that can be generated by each value in the feature map (the present study set $num_{\text{pbox}} = 2$, hoping to avoid an excessive number of prior boxes and reduce the computational complexity). The coordinates of the center point of prior boxes in the original image are $(X_{\text{pbox}}, Y_{\text{pbox}})$, $X_{\text{pbox}} \leftarrow (X_{m_i} + 0.5) \times s_{m_i}$, $Y_{\text{pbox}} \leftarrow (Y_{m_i} + 0.5) \times s_{m_i}$, wherein (X_{m_i}, Y_{m_i}) represents the coordinates of all points in the two-dimensional feature map m_i (only with $h_{m_i} \times w_{m_i}$ included), 0.5 the offset of the coordinates, and s_{m_i} the stride in the layer. In the process of corresponding the feature map m_i to the prior box, m_1

corresponds to a prior box with a matrix order of 16 × 16, ${}_{m_1^1}B_{\text{pbox}} = [b_{ij}] \in \mathrm{R}^{16\times16}$, and a matrix order of 32 × 32, ${}_{m_1^2}B_{\text{pbox}} = [b_{ij}] \in \mathrm{R}^{32\times32}$. Moreover, m_2 corresponds to a prior box with matrix orders of 64 × 64, ${}_{m_2^2}B_{\text{pbox}} = [b_{ij}] \in \mathrm{R}^{64\times64}$, and 128 × 128, ${}_{m_2^2}B_{\text{pbox}} = [b_{ij}] \in \mathrm{R}^{128\times128}$. In addition, $m_3$corresponds to a prior box with matrix orders of 256 × 256, ${}_{m_3^1}B_{\text{pbox}} = [b_{ij}] \in \mathrm{R}^{256\times256}$ and 512 × 512, ${}_{m_3^2}B_{\text{pbox}} = [b_{ij}] \in \mathrm{R}^{512\times512}$. In the final stage, the feature maps m_1, m_2, and m_3 that come from the Context Module were preprocessed via the head, and the final results were $\{A_{\text{face}}, A_{\text{fmask}}, A_{\text{bbox\_ofs}}\}$, wherein $A_{\text{face}} = [a_{ij}] \in \mathrm{R}^{N_{\text{pbox}}\times2}$ represents the confidence score of the ith prior box for the jth face category, $A_{\text{fmask}} = [a_{ij}] \in \mathrm{R}^{N_{\text{pbox}}\times2}$ represents the confidence score of the ith prior box for the jth category of wearing face masks, and $A_{\text{bbox\_ofs}} = [a_{ij}] \in \mathrm{R}^{N_{\text{pbox}}\times4}$ represents the offset of the ith prior box for the values of the jth coordinate for the bounding box regression task. The prediction of the offset via the bounding box regression is able to stay scale-invariant.

2.3 *Multi-Task Learning*

Multi-task learning is a training method based on sharing weights. Single-task learning trains prediction targets using different networks (or hidden layers), whereas the multi-task learning uses common networks and parameters for training, which helps reduce computation and over-fitting when model training is underway even though it takes more time to find out the minimum losing value of each task. The present study aims to detect face temperature and people wearing masks at the same time. Therefore, the multi-task learning method needs to perform the training of three different tasks, namely face classification, face mask wearing classification and bounding box regression. To avoid the imbalance of positive and negative samples caused by the task category of face classification, the focal loss function was selected as the basis for adjusting the network weight. The focal loss function is shown in Eq. (3) below:

$$\begin{aligned}\text{loss}_{\text{face}}(g_i^{\text{face}}, o_i^{\text{face}}; \alpha, \gamma) = \sum_{i=1}^{N_{\text{pbox}}} -\alpha\,(g_i^{\text{face}} \times \log(o_i^{\text{face}}) \\ + (1 - g_i^{\text{face}})^{\gamma} \times (1 - \log(o_i^{\text{face}}))\,),\end{aligned} \tag{3}$$

where g_i^{face} represents whether the ith prior box is the label of a human face or not $g_i^{\text{face}} = \{g_i^{\text{face}(1)}, \ldots, g_i^{\text{face}(n)}\}$, $g_i^{\text{face}(n)} \in \{0, 1\}$, indicating whether the ith prior box is the confidence score of a human face $o_i^{\text{face}} = \{o_i^{\text{face}(1)}, \ldots, o_i^{\text{face}(n)}\}$, $0 \le o_i^{\text{face}(n)} \le 1$, in which α is the coefficient that adjusts the ratio of positive to negative samples, and γ is the coefficient that adjusts the weight of easy samples. The combination of α and γ is conducive to solving the imbalance between positive and negative samples, thus improving the speed and accuracy of convergence.

In the process of face-mask-wearing classification, this study detected faces and mask wearing at the same time. In addition, mask wearing is related to human faces. Therefore, whether the prior box is a human face is determined first, after which it necessitates the judgment of whether there is a mask on the human face. In the inference stage, the non-maximum suppression (NMS) is used to remove the face bounding box that does not meet the need of face detection. Therefore, when NMS is performed, face classification is the major outcome, which often results in the removal of the box for those who wear masks properly. To avoid this situation, the present study improves the focal loss function, making it suitable for the detection of additional attributes based on the prior box (such as the detection of masks and goggles on faces). The parameter α of the original focal loss function, a hyperparameter manually assigned, is adjusted mainly based on the ratio of positive to negative samples. The present study proposed an improved mask-wearing loss function $\text{loss}_{\text{fmask}}(v_i, g_i^{\text{fmask}}, o_i^{\text{fmask}}; \gamma)$, changing α into a dynamically adjusted parameter so that it can be regulated according to the confidence score of the face category. The larger the confidence score of a face, the greater the impact on the loss of mask wearing. This way, it is expected that the confidence score of mask wearing can follow the face detection box with the largest confidence score to improve the effect of inference, as shown in Eq. (4) below:

$$\begin{aligned}\text{loss}_{\text{fmask}}(v_i, g_i^{\text{fmask}}, o_i^{\text{fmask}}; \gamma) = \sum_{i=1}^{N_{\text{pbox}}} -v_i\,(g_i^{\text{fmask}} \times \log \\ (o_i^{\text{fmask}}) + (1 - g_i^{\text{fmask}})^{\gamma}(1 - \log(o_i^{\text{fmask}}))),\end{aligned} \tag{4}$$

where v_i represents whether the ith prior box is the confidence score of the human face $v_i = \{v_i^{(1)}, \ldots, v_i^{(n)}\}$, $0 \le v_i^{(n)} \le 1$, g_i^{fmask} represents whether the ith prior box is the label of wearing masks $g_i^{\text{fmask}} = \{g_i^{\text{fmask}(1)}, \ldots, g_i^{\text{fmask}(n)}\}$, $g_i^{\text{fmask}(n)} \in \{0, 1\}$, o_i^{fmask} represents whether the ith prior box is the confidence score of wearing masks $o_i^{\text{fmask}} = \{o_i^{\text{fmask}(1)}, \ldots, o_i^{\text{fmask}(n)}\}$, $0 \le o_i^{\text{fmask}(n)} \le 1$, and γ is the coefficient that adjusts the weight of easy samples. The bounding box regression task can predict the offset between ground-truth and prior boxes, and the L2 Loss Function is used as the basis for adjusting the weight of the network, as shown in Eq. (5) below:

$$\text{loss}_{\text{bbox\_ofs}}(g_i^{\text{bbx\_ofs}}, o_i^{\text{bbx\_ofs}}) = \sum_{i=1}^{N_{\text{pbox}}} \left\| g_i^{\text{bbx\_ofs}} - o_i^{\text{bbx\_ofs}} \right\|_2^2, \tag{5}$$

where $g_i^{\text{bbx\_ofs}}$ represents the offset between the ith prior box and the ground-truth box $g_i^{\text{bbx\_ofs}} = \{g_i^{\text{bbx\_}ofs(1)}, \ldots, g_i^{\text{bbx\_}ofs(n)}\}$, $g_i^{\text{bbx\_}ofs(n)} \in \mathrm{R}$, and $o_i^{\text{bbx\_ofs}}$ represents the offset between the ith prior box and the predicted box $o_i^{\text{bbx\_ofs}} = \{o_i^{\text{bbx\_}ofs(1)}, \ldots, o_i^{\text{bbx\_}ofs(n)}\}$, $o_i^{\text{bbx\_}ofs(n)} \in \mathrm{R}$. In the training phase, the prior box is mapped to the feature map for each task, and the intersection over union (IoU) between the ground-truth bounding box B_{gt} and the prior

box B_{pbox} is calculated by Eq. (6) as follows:

$$J(B_{gt}, B_{\text{pbox}}) = \frac{\left|B_{gt} \cap B_{\text{pbox}}\right|}{\left|B_{gt} \cup B_{\text{pbox}}\right|}, \tag{6}$$

where B_{gt} represents the position of the ground-truth box in the training sample, B_{pbox} the prior box and the IoU between B_{gt} and B_{pbox}, which can be calculated via J function. With the prior box and the ground-truth object bounding box matched and IoU calculated, the IoU with the highest score can be regarded as pairing successfully. In other words, the optimal ground-truth box B^*_{gt} corresponding to each prior box is successfully identified. The ground-truth box B^*_{gt} with $IoU > 0.5$ can be regarded as a positive sample. Otherwise, it is a negative one. This way, the purpose of training can be achieved.

2.4 *Thermal Imaging Temperature Correction*

The present study uses the non-contact temperature sensor FLIR Lepton 3.5 to measure temperature. As a non-contact thermal camera, it is susceptible to the interference of environment variables. FLIR Lepton 3.5 has an error within 5% when measuring a target with a constant temperature of 35°. To reduce the error and achieve more accurate measurement of face temperature, it is necessary to correct temperature with the help of a blackbody that has constant temperature. The temperature of an object to be measured by a thermal imaging camera is affected by its emissivity. Only by setting objects to be measured with the same emissivity can correct temperature be obtained. The emissivity of a human body is 0.98 and that of a black body is 1.0; their emissivity is close to each other. Moreover, a black body can operate at constant temperature so that the black body can be used as the calibration basis. Support Vector Regression (SVR) algorithm [36, 37] can accurately predict numerical data. The Support Vector Machine (SVM) algorithm not only has excellent classification accuracy for object classification [38, 39], but its extended SVR algorithm can also accurately predict power generation [40–42]. Therefore, the present research employs the SVR algorithm for temperature correction of a thermal imaging camera.

SVR is an extension to the SVM classification algorithm. SVM is a supervised learning model, which consists of linear and non-linear ones. It is often applied to data classification and regression analysis. Linear SVM aims to find out the optimal hyperplane and classify the data optimally. It is assumed that the input data is $\{(x_i, y_i)\}$, $1 \le i \le n$, $y_i \in \{-1, 1\}$, $x_i \in R^d$, wherein x_i is the original input value while y_i is its label. Moreover, the optimal hyperplane can satisfy $w^T x + b = 0$, and its largest data classification distance (margin) is $2/\|w\|$.

The SVR algorithm provides a more flexible way for the model to set an acceptable allowable error. On the other hand, not all data follow a linear distribution, so SVR will map raw data into a high dimensional space through the kernel function, which is conducive to finding out the optimal hyperplane. SVR, like SVM, finds out the best hyperplane and minimizes the allowable error before data classification. The objective function of SVR is shown in Eq. (7) as follows:

$$\begin{aligned} \min & \left(\frac{1}{2}\|w\|^2 + C\sum_{i=1}^{\mathrm{N}}(\xi_i + \xi_i^*)\right) \\ \text{s.t.} \quad & y_i - wx_i - b \le \varepsilon + \xi_i \\ & -y_i + wx_i + b \le \varepsilon + \xi_i^* \\ & \xi_i, \xi_i^* \ge 0, \end{aligned} \tag{7}$$

where $\|w\|$ represents the distance, while ξ_i and ξ_i^* are the deviations of two different data sets. If the constant C is positive, overfitting can be avoided. The present study employs the SVR algorithm for temperature correction; The face bounding box B_{bbox} is obtained in the face detection stage, after which the temperature of the human face region T_{bbox} is obtained from the corresponding position of the thermal imaging camera temperature matrix T_{thermal}. The input variables of an SVR model include the maximum temperature of the black body region $\max(T_{\text{blackbody}})$, the actual temperature of the black body $t_{\text{blackbody}}$ and the maximum temperature of the face $\max(T_{\text{bbox}})$. The final predicted result of the model is the actual maximum temperature of the face region t^*_{bbox}, which can realize real-time temperature correction and reduce the error.

3. EXPERIMENT

In the present study, different experiments were conducted to verify the accuracy of the proposed method used to detect face mask wearing and non-contact face temperature.

3.1 *Face Recognition and Mask Wearing Detection Experiments*

In this study, AIZOOTech was used to collect the MAFA [43] and Wider Face [44] datasets. A total of 7,959 photos were collected, which were divided into training sets and testing sets. There were 6,120 photos in the training set, of which 3,006 were from MAFA dataset while 3,114 came from Wider Face dataset. There were 1,839 photos in the testing set, of which 1,059 came from MAFA and 780 were from Wider Face. The majority of the photos were covered faces, so the diversity of the samples was insufficient. To improve the detection accuracy, pre-training was conducted via Wider Face to obtain a pre-trained model. Moreover, a new model was trained based on the model parameters that experienced inheritance training. This study aims to use edge devices for real-time mask wearing detection and face temperature measurement. The experiment consisted of two parts: parameter quantity and detection speed, as well as the evaluation of detection accuracy. The parameter quantity is the sum of the CNN parameters for each layer, while the detection speed was measured based on floating point operations (FLOPs). Moreover, the model detection accuracy was evaluated using the receiver operating characteristic (ROC) curve, and the area under the curve (AUC) and mAP

were calculated as the benchmark. When the CNN model is loaded, the trained parameters are loaded into the memory to facilitate subsequent recall and computation. In addition, the parameter quantity affects the utilization of the memory. The computation of the parameter quantity needs to consider the convolution kernel size $k \times k$, the size of input channels C_{in}^i and output channels C_{out}^i for each convolution layer, which are calculated by Eq. (8) below. The parameter quantity of a model affects memory utilization. FLOPs serve as a speed indicator that reveals the computation speed, which involves the input size for each layer $H^i \times W^i$. Regarding edge devices, the initial input image size greatly affects the amount of subsequent computation. The FLOPs are computed by Eq. (9). The experimental results show the number of parameters in our proposed model is 0.42 M, and the amount of computation is 2.039 GFLOPs.

$$\text{parameters} = \sum_i (h^i \times w^i \times C_{\text{in}}^i + 1) \times C_{\text{out}}^i \tag{8}$$

$$\text{FLOPs} = \sum_i (h^i \times w^i \times C_{\text{in}}^i + 1) \times C_{\text{out}}^i \times H^i \times W^i. \tag{9}$$

Regarding the evaluation of the object detection model, both the object detection position and the confidence score of classification will affect its performance. If the offset of the position is too much and cannot be filtered through the threshold of confidence scores, an excessive number of false reports and detection will occur. In the present study, the intersection over union (IoU) was used as the benchmark to determine the similarity between the bounding box and the ground-truth box when the accuracy of the object detection box was evaluated. If the two boxes overlap, then $IoU = 1$, indicating that the detection is accurate.

When the object classification of the detection box was evaluated, whether there was a better confidence score threshold was evaluated to screen out wrong detection boxes. The confusion matrix under each threshold was computed to measure the difference between predicted and actual results. The matrix is composed of ground truth and predicted labels, which is conducive to the analysis of classification accuracy by True Positive (TP), False Positive (FP), False Negative (FN), and True Negative (TN). The number of classification prediction results for each object under each confidence score threshold can be obtained through the above confusion matrix. After computation, the ratio of true positive rates $\text{TPR} = \text{TP}/(\text{TP} + \text{FN})$ to false positive rates $\text{FPR} = \text{FP}/(\text{FP} + \text{TN})$ was calculated and used as the basis of ROC curve computation. TPR and FPR can measure the accuracy of the object detection model. In the present study, they were used to draw a ROC curve, as shown in Figure 1, with corresponding thresholds of confidence scores, serving as the basis for the accuracy of the detection model. It is used for detection through the area ratio of the AUC (i.e. probability); when the AUC is greater than 0.7, it means that the model is able to make predictions accurately under a given confidence score threshold. The benchmark can be

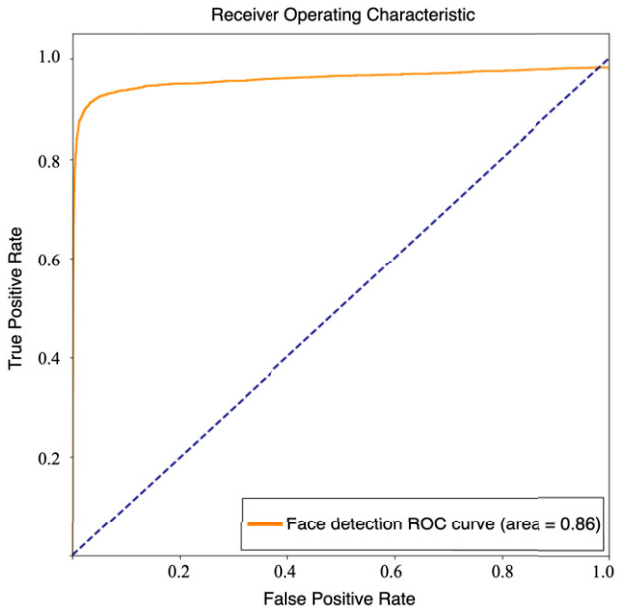

Figure 1. The ROC Curve of the Proposed Model with an AUC of 0.86.

used to predict the quality of the proposed model based on the AUC. The benchmark of $0.8 \leq \text{AUC} \leq 0.9$ means the excellent discrimination and that of $0.9 \leq \text{AUC} \leq 1.0$ means the outstanding discrimination. The AUC of the proposed model arrived at 0.86, indicating that the proposed model has the excellent discrimination.

There are a number of methods to measure the accuracy of object detection models, of which the most commonly used measuring metric is the average precision (AP). It is computed by using the confusion matrix to calculate the mean of the precision under different recall and IoU. The precision is calculated as TP/(TP+FP) and the recall is calculated as TP/(TP+FN). AP can be calculated with methods when it comes to different datasets. Therefore, this study used three different AP metrics to evaluate the accuracy of the model, namely AP for VOC 2007, AP for VOC 2010 and mAP for COCO. In VOC 2007 dataset, the AP is computed as follows: $IoU > 0.5$ is regarded as a positive sample and the confusion matrix under different confidence score thresholds are calculated. After that, the precision and recall (P_c, R_c) of the detection model are calculated. AP is computed via Eq. (10) below:

$$AP = \frac{1}{N_r} \sum_{i=1}^{N_r} g_i(P_c, R_c; r_i), \tag{10}$$

where N_r represents the number of recall thresholds. In the VOC 2007 dataset, $N_r = 11$, and r_i represents 11 recall thresholds, $r \in \{0, 0.1, 0.2, \ldots, 1\}$. The function $g_i(P_c, R_c; r_i)$ will return the maximum value that meets the criteria $R_c \geq r_i$. In the VOC 2010 dataset, the recall thresholds $r_i, r \in \{0, 0.14, 0.29, 0.43, 0.57, 0.71, 1\}\, N_r = 7$ was adjusted when AP was computed, while other computation methods were the same as those used to calculate the AP in VOC 2007 dataset. When the AP of COCO

Table I. Experimental results of face detection.

Dataset	Measuring metric	Results
MAFA + WIDERFACE Face Mask Dataset Val	VOC 2007 AP	0.8957508284908179
	VOC 2010 AP	0.8532333887049719
	COCO mAP	0.5544180875433907
	AUC	0.862302828740021

Table II. Average precision on the MAFA dataset.

Methods	AP on MAFA
LEE-CNNs [43]	0.764
DEFace [45]	0.778
The method proposed by the present study	0.787

dataset was computed, a more rigorous evaluation method was adopted in comparison with that for the VOC dataset; VOC only adopts the threshold condition $IoU > 0.5$, but the AP of COCO adopted a group of threshold conditions $I_i, I \in \{0.5, 0.55, \ldots, 0.95\}$ and 101 recall thresholds $r_i, r \in \{0, 0.01, 0.02, \ldots, 1\}, N_r = 101$ for AP computations for ten times. Finally, the average of ten-time AP computation results is taken as the AP value of the COCO dataset. Table I shows the experimental results of the proposed detection model under different measuring metrics. Table II lists the experimental results of the proposed detection model in MAFA testing set. Its accuracy arrived at 0.787, the parameter quantity 0.42 M, and computation 2.039 GFLOPs.

During the training process of the face mask-wearing detection model, this study added dynamic α to the focal loss function, enabling an increase in the accuracy of mask wearing detection. After the detection model had been trained for 250 epochs, the loss value of the original focal loss function dropped from 9.6787 to 1.3312, and the loss value of the focal loss function with the added dynamic α decreased from 8.9384 to 1.3728. Besides, after the detection model employed the validation set as well as 250 epochs of tests, the loss value of the original focal loss function ranged between 0.0486 and 0.0487, and the loss value of the focal loss function with the added dynamic α declined from 0.0499 to 0.0479, which can show better robustness in object detection.

When mask-wearing detection is evaluated, the classification result is a subtask, for the detection of whether masks are worn is necessary only when the position of a face is successfully detected. Thus, IoU that is greater than a certain threshold is considered as the benchmark. In the present study, the samples of $IoU > 0.5$ were utilized to evaluate the ROC curve, AP for VOC 2007 and AP for VOC 2012. However, the number of samples to be evaluated decreased with the increase of IOU thresholds, thus resulting in an increase of AP value when mAP for COCO was measured. The ROC curve of mask-wearing detection is shown in Figure 2. AUC arrived at 0.98,

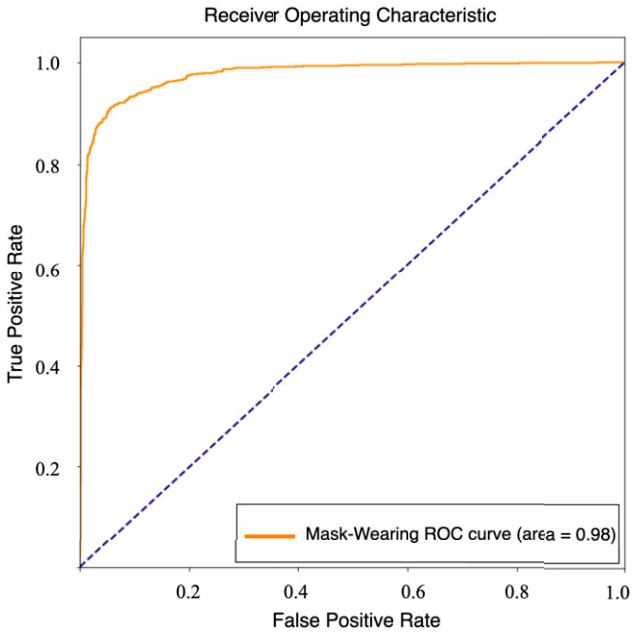

Figure 2. The ROC Curve of the Mask-Wearing Classification, AUC = 0.98.

Table III. Experimental Results of Mask Classification.

Dataset	Measuring metric	Results
MAFA+WIDERFACE Face Mask Dataset Val	VOC 2007 AP	0.8807275309514689
	VOC 2010 AP	0.8363384188626908
	COCO mAP	0.9518421957193862
	AUC	0.9783792991054246

indicating outstanding discrimination. The experimental results of mask-wearing detection under different metrics are shown in Table III, which shows that AP calculated based on the COCO dataset has the highest accuracy, and the mAP reaches 0.95. From these experimental results, the face mask-wearing can be successfully detected by the proposed method and the detected result of a reality scenario is shown in Figure 3.

3.2 *Thermal Imaging Camera Temperature Calibration Experiment*

The thermal imaging camera is susceptible to environmental variables, which reduces its measurement accuracy. Therefore, the present study used the blackbody with constant temperature and an emissivity of 1.0 for temperature correction. In this study, two blackbodies were utilized to train the thermal imaging camera temperature calibration model: one blackbody, which was set at the constant temperature of 40.00°C as reference, recorded its actual measurement temperature X^a, while the other, whose constant temperature was set between 36.00°C and 40.00°C, recorded the temperature it measured X^b at a sampling interval of every 0.05°C. In the data preprocessing stage, the blackbody was measured 8 times per second. Due to data

Figure 3. The detection results of the study in a simulated reality scenario. The proposed method is able to detect various face mask-wearing and face temperature with both the surveillance camera and the thermal imaging camera.

redundancy, duplicate data were deleted. There were 999,768 entries of raw data, whereas 23,445 entries of data were kept after repeated ones were removed.

When the SVR model was in the training phase, cross validation was employed to divide the dataset into training data, testing data and validation data based on a ratio of 0.8:0.1:0.1. In addition, the polynomial kernel function was adopted to map the data into a high dimensional space. The SVR model involves different hyperparameters, i.e. the degree of the polynomial d, the penalty function C, and adjustment coefficient g. During parameter optimization, the grid search with cross validation was utilized to find out the best hyperparameter combination and avoid overfitting. The data can be mapped into a high dimensional space via the polynomial kernel function $k(X_a, X_b)$ as shown in Eq. (11):

$$k(X_a, X_b) = (gX_a \cdot X_b + 1)^d, \tag{11}$$

where X_a and X_b represent the temperature records of two blackbodies. During the cross validation, three indicators were adopted for measurement, namely the coefficient of determination r^2, mean-square error (MSE), and mean absolute percentage error (MAPE). The coefficient of determination r^2 indicates the explanatory power of the regression model, and $r^2 > 0.5$ means that the model has a preliminary explanatory power for the input–output relationship. The MSE, which represents the error between predicted and actual values, is calculated via Eq. (12) below:

$$MSE = \frac{1}{N}\sum_{d=1}^{N}\left(y_d - \hat{y}_d\right)^2, \tag{12}$$

where N represents the sample number of validation data, y_d represents the actual temperature difference of the dth validation dataset, while $\hat{y}_d$ represents the predicted temperature difference of the dth validation dataset. MAPE indicates the percent error between the predicted and actual values, which is calculated via Eq. (13) below:

$$MAPE = \frac{1}{N}\sum_{d=1}^{N}\left|\frac{y_d - \hat{y}_d}{y_d}\right|. \tag{13}$$

Table IV. The best hyperparameters and accuracy of the SVR temperature calibration model.

Kernel function	Polynomial kernel function
The degree of the polynomial d	1
Penalty function C	1000.0
Adjustment coefficient g	0.0001
Coefficient of determination r^2	0.9950
MSE	0.0425
MAPE	4.8545

The best hyperparameter values obtained via the grid search with cross validation are listed in Table IV, wherein the MSE of y_d and $\hat{y}_d$ is 0.0425. In addition, the MAPE of the proposed model is 4.8545%, which is less than 10%. Therefore, it is a highly accurate prediction model. The benchmark of $MAPE < 10\%$ indicates highly accurate forecasting, and that of $10\% \leq MAPE < 20\%$ indicates good forecasting.

After the face temperature detection model is calibrated, the scatter plot for the predicted values versus standardized residuals between 25° C and 40° C is shown in Figure 4, wherein the residuals are all scatted around the 0 dashed line and fall within ±2 standard deviations of the 95% confidence

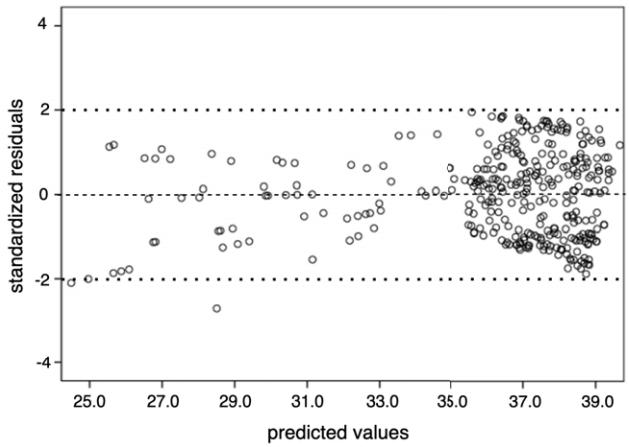

Figure 4. The scatter plot for the predicted values versus standardized residuals of the face temperature detection model.

interval, except for 2 predicted values containing larger residuals. The residuals of the predicted values between 35° C and 39° C all fall within ±2 standard deviations, indicating that the face temperature detection model in this study is applicable to the measurement of face temperature.

4. CONCLUSION

As of 2021, COVID-19 continues to spread all over the world. Wearing face masks and measuring body temperature remain to be the most important basic requirements for epidemic prevention. The present study used the multi-task RetinaFace framework for face detection and mask-wearing detection, and adopted MobileNet as the backbone because of its less parameter quantity and computation, which facilitates real-time feature extraction as well as computation and deployment of the detection system in the edge device. For the majority of object classification learning algorithms, the focal loss function was adopted to avoid the problem that the class distribution of data are highly unbalanced, which results in class imbalance. Consequently, the present study replaced the hyperparameter value α with dynamic value α, which was applied to the subtask training stage of the mask-wearing detection in order to improve detection accuracy and avoid class imbalance.

MAFA and Wider Face datasets were utilized in this research; a total of 7,959 samples (including the pictures of human faces, faces with masks and masquerades on them). The parameter quantity reached 0.42 M, and the computation quantity arrived at 2.039 GFLOPs. In terms of face detection accuracy, the accuracy of AP for VOC 2007 was 0.8958, and AUC 0.8623, which indicates that the detection accuracy of this model is high. In addition, the detection accuracy of MAFA was 0.787, and as this study focused on real-time face detection, MobileNet was set as the backbone for its low parameter quantity and computation. Regarding the accuracy of mask-wearing detection, the accuracy of mAP for COCO arrived at 0.9784, and AUC 0.9518, revealing that the detection accuracy of this model is very high. As for the validation of thermal imaging temperature correction model, the MSE arrived at 0.425 and the MAPE reached 4.8545, indicating that the model has highly accurate forecasting. In order to achieve safe and fast face temperature detection, this study adopted the Lepton 3.5 thermal camera module developed by FLIR and performed real-time temperature calibration. The experimental results for the SVR temperature detection model can only represent the higher prediction accuracy that can be achieved through a specific hardware combination, whereas those for other types of thermal cameras still need to be verified. This study used dynamic value α to modify the focal loss function of the face mask wearing classification task. Although it can improve the accuracy of multi-task learning, it is only applicable to additional sub-task classification. Moreover, the actual application will be limited by the accuracy of face detection, so that the accuracy of sub-tasks will reduce. Consequently, not only does future research need to focus on the accuracy of sub-task detection, but it also needs to enhance the accuracy of face classification at the same time, thereby lifting the overall object detection and recognition accuracy.

The detection model proposed in this study combines real-time mask-wearing detection with face temperature measurement, which can help to quickly measure the body temperature and detect whether one wears face masks properly in the context of COVID-19, so as to reduce the risk of epidemic spread.

REFERENCES

[1] D. K. Chu, E. A. Akl, S. Duda, K. Solo, S. Yaacoub, and H. J. Schuneman, "Physical distancing, face masks, and eye protection to prevent person-to-person transmission of SARS-CoV-2 and COVID-19: A systematic review and meta-analysis," Lancet **395**, 1973 (2020).

[2] L. Fu, B. Wang, T. Yuan, X. Chen, Y. Ao, T. Fitzpatrick, P. Li, Y. Zhou, Y.F. Lin, Q. Duan, and G. Luo, "Clinical characteristics of coronavirus disease 2019 (COVID-19) in China: A systematic review and meta-analysis," J. Infect. **80**, 656 (2020).

[3] J. Howard, A. Huang, Z. Li, Z. Tufekci, V. Zdimal, H. M. van der Westhuizen, A. von Delft, A. Price, L. Fridman, L. H. Tang, and V. Tang, "An evidence review of face masks against COVID-19," Proc. Natl. Acad. Sci. **118** (2021).

[4] S. Kuutti, R. Bowden, Y. Jin, P. Barber, and S. Fallah, "A survey of deep learning applications to autonomous vehicle control," IEEE Trans. Intell. Transport. Syst. **22**, 712 (2021).

[5] D. W. Otter, J. R. Medina, and J. K. Kalita, "A survey of the usages of deep learning for natural language processing," IEEE Trans. Neural Netw. Learning Syst. **32**, 604 (2021).

[6] P. Arora, H. Kumar, and B. K. Panigrahi, "Prediction and analysis of COVID-19 positive cases using deep learning models: A descriptive case study of India," Chaos, Solitons & Fractals **139**, 110017 (2020).

[7] S. H. Liao and C. H. Wen, "Artificial neural networks classification and clustering of methodologies and applications- literature analysis from 1995 to 2005," Expert Syst. Appl. **32**, 1 (2007).

[8] J. F. Mas and J. J. Flores, "The application of artificial neural networks to the analysis of remotely sensed data," Int. J. Remote Sens. **29**, 617 (2008).

[9] M. A. Mohammed, M. K. Abd Ghani, N. A. Arunkumar, R. I. Hamed, M. K. Abdullah, and M. A. Burhanuddin, "A real time computer aided object detection of nasopharyngeal carcinoma using genetic algorithm and artificial neural network based on haar features fear," Future Gener. Comput. Syst. **89**, 539 (2018).

[10] Y. Lecun, B. Boser, J.S. Denker, D. Henderson, R.E. Howard, W. Hubbard, and L.D. Jackel, "Backpropagation applied to handwritten zip code recognition," Neural Comput. **1**, 541 (1989).

[11] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," Proc. IEEE **86**, 2278 (1998).

[12] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," Adv. Neural Inf. Process. Syst. **1097** (2012).

[13] R. Girshick, "Fast R-CNN," *IEEE Int. Conf. Comput. Vis. (ICCV)* (IEEE, Piscataway, NJ, 2015), p. 1440.

[14] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: towards real-time object detection with region proposal networks," IEEE Trans. Pattern Anal. Mach. Intell. **39**, 1137 (2017).

[15] O. Cakiroglu, C. Ozer, and B. Gunsel, ""Design of a Deep Face Detector by Mask R-CNN," *27th Signal Process. Commun. Appl. (SIU)* (IEEE, Piscataway, NJ, 2019), p. 1,".

[16] C. Szegedy, A. Toshev, and D. Erhan, "Deep neural networks for object detection," Adv. Neural Info. Process. Syst. **26**, 2553 (2013).

[17] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun, Overfeat: Integrated recognition, localization and detection using convolutional networks, Computer Vision and Pattern Recognition, preprint arXiv:1312.6229 [cs.CV] [online] (2013).

[18] J. Redmon and A. Farhadi, "YOLO9000: Better, Faster, Stronger," *IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)* (IEEE, Piscataway, NJ, 2017), Vol. 6517.

[19] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C. Y. Fu, and A. C. Berg, "SSD: Single shot multibox detector," *European Conf. on Computer Vision ECCV 2016: Computer Vision–ECCV* (2016), Vol. 21, pp. 21–37.

[20] H. Li, Z. Lin, X. Shen, J. Brandt, and G. Hua, "A convolutional neural network cascade for face detection," *IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)* (IEEE, Piscataway, NJ, 2015), p. 5325.

[21] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, "Joint face detection and alignment using multitask cascaded convolutional networks," IEEE Signal Process. Lett. **23**, 1499 (2016).

[22] M. Najibi, P. Samangouei, R. Chellappa, and L. S. Davis, "SSH: Single stage headless face detector," *IEEE Int'l. Conf. Comput. Vis. (ICCV)* (IEEE, Piscataway, NJ, 2017), p. 4885.

[23] J. Wang, Y. Yuan, and G. Yu, Face attention network: An effective face detector for the occluded faces, Computer Vision and Pattern Recognition, arXiv e-prints arXiv:1711.07246 [cs.CV] (2017).

[24] T. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," *IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)* (IEEE, Piscataway, NJ, 2017), p. 936.

[25] J. Deng, J. Guo, E. Ververas, I. Kotsia, and S. Zafeiriou, "RetinaFace: single-shot multi-level face localisation in the wild," *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)* (IEEE, Piscataway, NJ, 2020), p. 5203.

[26] F. Gurkan, B. Sagman, and B. Gunsel, "YOLOv3 as a Deep Face Detector," *11th Int. Conf. Electr. Electron. Eng. (ELECO)* (IEEE, Piscataway, NJ, 2019), p. 605.

[27] H. Jiang and E. Learned-Miller, "Face Detection with the Faster R-CNN," *12th IEEE Int'l. Conf. Autom. Face Gesture Recognit. (FG)* (IEEE, Piscataway, NJ, 2017), p. 650.

[28] L. Zhao, Q. Li, C. H. Wang, and Y. C. Liao, "3D brain tumor image segmentation integrating cascaded anisotropic fully convolutional neural network and hybrid level set method," J. Imaging Sci. Technol. **64**, 040411 (2020).

[29] K. Bai, Q. Li, and C. H. Wang, "Integrating improved U-Net continuous maximum flow algorithm for 3D brain tumor image segmentation," J. Imaging Sci. Technol. **64**, 040412 (2020).

[30] C. H. Wang, "An intuitionistic fuzzy set-based hybrid approach to the innovative design evaluation mode for green products," Adv. Mech. Eng. **8**, 1–16 (2016).

[31] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)* (IEE, Piscataway, NJ, 2016), p. 770.

[32] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," Computer Vision and Pattern Recognition, preprint arXiv:1409.1556 [online], (2014).

[33] 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," Computer Vision and Pattern Recognition, arXiv e-prints arXiv:1704.04861 (2017).

[34] T. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," *IEEE Int'l. Conf. Comput. Vis. (ICCV)* (IEEE, Piscataway, NJ, 2017), p. 2999.

[35] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: unified, real-time object detection," *IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)* (IEEE, Piscataway, NJ, 2016), p. 779.

[36] A. J. Smola and B. Schölkopf, "A tutorial on support vector regression," Stat. Comput. **14**, 199 (2004).

[37] K. P. Lin, H. F. Chang, T. L. Chen, Y. M. Lu, and C. H. Wang, "Intuitionistic fuzzy C-regression by using least squares support vector regression," Expert Syst. Appl. **64**, 296–304 (2016).

[38] 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 **167**, 108288 (2021).

[39] Y. Zhou, X. Zhao, K. P. Lin, C. H. Wang, and L. Li, "A gaussian process mixture model-based hard-cut iterative learning algorithm for air quality prediction," Appl. Soft Comput. **85**, 105789 (2019).

[40] Z. F. Liu, S. F. Luo, M. L. Tseng, H. M. Liu, L. Li, and A. Hashan Md Mashud, "Short-term photovoltaic power prediction on modal reconstruction: A novel hybrid model approach," Sustain. Energy Technol. Assess. **45**, 101048 (2021).

[41] L. Li, X. D. Chen, M. L. Tseng, C. H. Wang, K. J. Wu, and M. K. Lim, "Effective power management modeling of aggregated heating, ventilation, and air conditioning loads with lazy state switching," J. Clean. Prod. **166**, 844–850 (2017).

[42] L. Li, J. Sun, C. H. Wang, Y. T. Zhou, and K. P. Lin, "Enhanced gaussian process mixture model for short-term electric load forecasting," Inf. Sci. **477**, 386–398 (2019).

[43] S. Ge, J. Li, Q. Ye, and Z. Luo, "Detecting masked faces in the wild with LLE-CNNs," *IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)* (IEEE, Piscataway, NJ, 2017), p. 426.

[44] S. Yang, P. Luo, C. C. Loy, and X. Tang, "WIDER FACE: A face detection benchmark," *IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)* (IEEE, Piscataway, NJ, 2016), p. 5525.

[45] T. M. Hoang, G. P. Nam, J. Cho, and I. J. Kim, "DEFace: Deep efficient face network for small scale variations," IEEE Access **8**, 142423 (2020).