# Explainable COVID-19 Infections Identification and Delineation Using Calibrated Pseudo Labels

Ming Li , Yingying Fang, Zeyu Tang, Chibudom Onuorah , Jun Xia, Javier Del Ser , *Senior Member, IEEE*, Simon Walsh, and Guang Yang , *Senior Member, IEEE*

*Abstract*—The upheaval brought by the arrival of the COVID-19 pandemic has continued to bring fresh challenges over the past two years. During this COVID-19 pandemic, there has been a need for rapid identification of infected patients and specific delineation of infection areas in computed tomography (CT) images. Although deep supervised learning methods have been established quickly, the scarcity of both image-level and pixel-level labels as well as the lack of explainable transparency still hinder the applicability of AI. Can we identify infected patients and delineate the infections with extreme minimal supervision? Semi-supervised learning has demonstrated promising performance under limited labelled data and sufficient unlabelled data. Inspired by semi-supervised learning, we propose a model-agnostic calibrated pseudo-labelling strategy and apply it under a consistency regularization framework to generate explainable identification and delineation results. We demonstrate the effectiveness of our model with the combination of limited labelled data and sufficient unlabelled data or weakly-labelled data. Extensive experiments have shown that our model can efficiently utilize limited labelled data and provide explainable classification and segmentation results for decision-making in clinical routine.

*Index Terms*—COVID-19, semi-supervised learning, pseudo-labelling, consistency regularization, explainability.

## I. INTRODUCTION

IT HAS been more than two years since we have lived under the shadow of coronavirus disease 2019 (COVID-19) pandemic, which is caused by the severe acute respiratory syndrome coronavirus 2. At the time of writing, over 386 million people have been diagnosed with COVID-19, and an estimated 5.7 million death cases were confirmed globally according to the World Health Organization. The virus binds to epithelial cells in the airway tract, from where it further invades the alveolar epithelial cells causing inflammation and scarring in the lung region [1], [2].

The reverse transcriptase-polymerase chain reaction (RT-PCR) test is the current mainstream diagnostic tool as it is rapid and easy to perform. Samples can be collected from respiratory sources of patients using multiple methods, including nasopharyngeal swabs and oropharyngeal swabs, etc. Nonetheless, challenges persist as the molecular detection methods can suffer from a high false-negative rate and low sensitivities in part due to low viral load, improper clinical sampling, and the quality of the test kit [3], [4]. While computed tomography (CT) can successfully detect COVID-19 in asymptomatic patients and patients with false-negative RT-PCR results [5]. Compared to RT-PCR, the CT-based diagnosis can be made directly after the image acquisition by the assessment of radiologists and also indicate the infected areas and severity of the illness [6]. Therefore, researchers are intent on exploring a burgeoning interest in CT to assist COVID-19 patient identification. Several deep learning methods has been explored to diagnose COVID-19 using CT scans [7]–[10]. Detailed reviews of the existing diagnostic models for COVID-19 can be found in [11], [12].

Albeit deep learning has demonstrated its success in diagnosing different diseases [13]–[16], particular challenges still exist in the diagnosis of COVID-19 [12]. The CT scans of COVID-19 patients share similar typical features with other pneumonia patients, such as the ground-glass opacity (GGO) in the lung periphery and lower lobes, rounded opacity, enlarged intra-infiltrate vessels, and consolidations in subsegmental areas [17], [18], making it difficult to distinguish the COVID-19 cases from other community-acquired pneumonia (CAP) cases and leading to the false-positive cases. Besides, the performance of most COVID-19 diagnostic models heavily relies on a large and unbiased labelled dataset, which is very expensive and
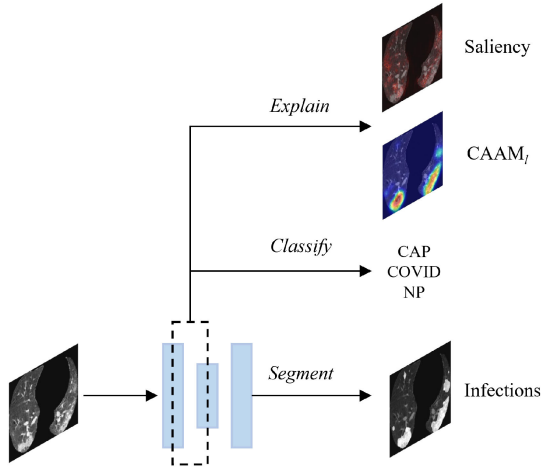
Fig. 1. Illustration of our multi-task method which produces classifications, segmentations as well as explainable visualizations at the same time. Here, Integrated Gradients is employed to obtain the saliency map.

time-consuming considering the cost of manual labelling by experts. Another challenge lies in the explainability of the predictions as the models are expected to assist doctors in clinical routine. Ideally, the infections in the CT scans are expected to be localized and support the predictions of those positive cases.

To alleviate the aforementioned three challenges: 1) false-positive predictions caused by similar features shared between COVID-19 and CAP cases, 2) the scarcity of both image-level and pixel-level labels, and 3) the lack of explainable transparency, in this paper, we propose a multi-task explainable identification and delineation method for COVID-19 infections. As shown in Fig. 1, our multi-task method produces classification results (CAP, non-pneumonia (NP), COVID-19) for identification, segmentation results (COVID-19 infections) as well as explainable visualizations (refined class activation map ($CAAM_l$), saliency map) as delineation. We use a multiscale loss function $\mathcal{CAM}_{loss}$ to constrain class-agnostic activation map (CAAM) closer to vanilla class activation map (CAM), enforcing intra-class compactness and inter-class separability, which helps to better distinguish COVID-19 features from CAP features. By fusing $CAAM_l$, saliency map and segmentation together, calibrated pseudo segmentation mask labels (pseudo label) are generated to address the scarcity of pixel-level infection labels. At the same time, $CAAM_l$ and saliency map act as the explainable role, providing auxiliary advice for decision making and verification for segmentation. The contributions of this paper are summarized as follows:

1) We propose a multi-task method for explainable identification and delineation of COVID-19 Infections.
2) The proposed multiscale $\mathcal{CAM}_{loss}$ brings an improvement in classification and segmentation performance.
3) We design a calibrated pseudo-labelling strategy and apply it under the consistency regularization framework, which only uses limited infection labels.
4) The proposed method achieves superior performance compared with existing methods, with an accuracy of 77.22%, and the area under the receiver operating characteristic

curve (AUC) of 0.8948 for the classification task as well as the dice score of 85.49%, and the mean intersection over union (mIoU) score of 76.97% for segmentation task.

## II. RELATED WORK

### A. Supervised COVID-19 Diagnosis

Currently, most CT-based COVID-19 diagnosis methods are based on supervised learning and obtain promising results. Wang et al. [19] proposed an attention residual learning framework that achieved 93.3% for accuracy, 87.6% for sensitivity, and 95.5% for specificity on a private CT dataset. Owais et al. [20] developed a multilevel deep aggregated boosted network that can diagnose COVID-19. The model was trained and evaluated on six public datasets, achieving 95.38% for accuracy, 92.53% for specificity, and 98.14% for sensitivity. Zhang et al. [9] introduced an explainable multi-input deep convolution attention network for COVID-19 classification, achieving a sensitivity of 98.1%, a specificity of 98.0%, and an accuracy of 98.0% on 86 cases. Wu et al. [21] developed a joint classification and segmentation framework (JCS) for real-time COVID-19 diagnosis using CT scans. They utilized attention mechanism and deep supervision in the classification and segmentation branch, achieving 95.0% sensitivity and 93.0% specificity for classification task and 95.9% dice score for segmentation task.

As mentioned in the introduction, the scarcity of both image-level and pixel-level labels has all these supervised methods over a barrel. Thus, we design a calibrated pseudo-labelling strategy and apply it under the consistency regularization framework to address the scarcity of labels, see more details in III-E.

### B. Semi-Supervised Classification and Segmentation

Existing studies are limited due to the shortage of labelled CT images since manually labelling can be costly and time-consuming. To combat this, various semi-supervised methods have been investigated to carry out infection detection without relying on large labelled datasets. Aviles et al. [22] proposed a graph-based semi-supervised COVID-19 classification framework. An optimization model for graph diffusion was introduced, which reinforced the natural relation between the smaller labelled dataset and the unlabelled dataset. The framework was evaluated on a chest X-ray dataset, achieving 95% accuracy. Fan et al. [23] developed a COVID-19 lung infection segmentation model for automatic identification of infected regions in CT scans. A semi-supervised framework based on a randomly selected propagation strategy was used to reduce the reliance on a large labelled dataset. Besides, Haque et al. [24] introduced a multi-task semi-supervised learning method (MultiMix) for the classification of pneumonia and segmentation of the lungs in chest X-ray images. They also preserved explainability in the lung area through a saliency bridge, but the saliency map they generated mostly focused on the lung periphery and did not match the lung segmentation results very well.

Though semi-supervised methods show promising potential, most of them only use vanilla CAM or inferior methods as

explanations, which are not enough for explainable transparency in the prediction of their models. Hence, we employ multiple explainable methods, i.e., CAAM$_l$ and saliency map to provide better explanations for our model, see more details in III-F.

### C. Explainable AI With COVID-19

The mainstream explanation methods used by existing COVID-19 diagnostic models are CAM-based methods [21], [22], [24]–[27]. CAM was first introduced in 2016 by Zhou *et al.* [28], a global average pooling layer was added after the last convolutional layer, which helped localize discriminative regions for a particular class. A more generalized gradient-weighted CAM was proposed in [29], which can produce a visual explanation for arbitrary model architecture. However, the CAM-based methods only focused on the most discriminative areas while neglecting subtle regions and returned coarse-grained explanation maps as the features it used had gone through a sequence of downsamplings. The saliency map [30] can produce fine-grained explanation maps with high resolution, but it was very sensitive to the input data and may become unreliable, for instance, the preprocessing procedure (e.g., normalization) can make undesirable changes in the derived saliency map [31], [32]. To alleviate these drawbacks, Hu *et al.* [25] proposed a weakly supervised multiscale joint saliency method for COVID-19 infection detection and classification. Another novel approach introduced by Baek *et al.* [33] was to sum up feature maps directly, which generated a class-agnostic activation map (CAAM), indicating the spatial distribution of the embedded features. Compared with the vanilla CAM, CAAM showed larger activated areas and richer features but was more error-prone in classification due to the redundant feature representations. Thus, Wang *et al.* [34] constructed a loss function CAM-loss that effectively reinforced intra-class compactness and inter-class separability by constraining CAAM closer to CAM.

## III. METHOD

To assist clinicians in identifying and delineating COVID-19 infections with explainability under extreme minimal supervision, we propose a semi-supervised multi-task method using limited labelled data and sufficient unlabelled data. The overview of the framework is depicted in Fig. 2. Following, we describe more details of the proposed method.

### A. Prerequisite: Lung Area Extraction

By suppressing the disturbance of the non-lung area [25], [35], we can facilitate the COVID-19 infection classification and segmentation with higher accuracy as well as better explainability. We follow the procedure described in [25], using the TCIA open dataset [36] to train a lung area segmentation model. The trained lung area segmentation model is then utilized to segment the entire lung structure of the COVID-19, CAP, and NP patients included in our private dataset.

### B. Backbone Architecture

As shown in Fig. 3, we use VGG architecture [37] as our backbone. Five convolutional blocks (i.e., Conv1, Conv2, Conv3,
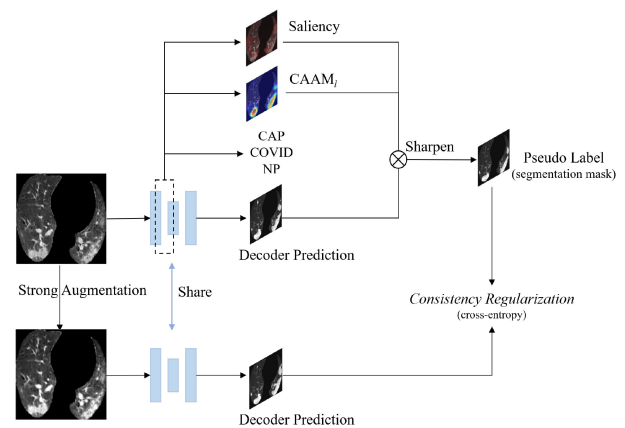


Fig. 2. Overview of the framework. Given one unlabelled CT slice, it is fed into the network to obtain a saliency map, CAAM$_l$, classification result as well as the decoder prediction. Then a calibrated pseudo label is generated via a sharpen combination module. Meanwhile, a strongly augmented version of the given unlabelled CT slice goes through the network and gets its decoder prediction. Next, we enforce consistency regularization on this decoder prediction to match the calibrated pseudo label via a pixel-wise cross-entropy loss.

Conv4, and Conv5) extract multiscale features, from bottom to top, each block consists of 2, 2, 3, 3, 3 convolutional layers, respectively. Every convolutional layer is a combination of convolution layers with $3 \times 3$ kernels, instance normalization, and Leaky Rectified Linear Units. There are 32, 64, 128, 256, and 256 filters in the corresponding convolutional blocks and 3 filters in the $1 \times 1$ convolution layer. Each convolutional block is followed by a max-pooling layer with a kernel size of $2 \times 2$.

### C. Multiscale $\mathcal{CAM}_{loss}$ for Classification

Considering the COVID-19 infections appear to be small in mild cases, while in severe cases they scatter over a large area [38], [39]. We follow [25] to implement a multiscale learning scheme to cope with the variation of infection location and size by learning different features in various scales. The mid-level layer can learn small lesions (e.g., GGO), but is not able to capture larger scattered lesions due to limited receptive field. While the higher-level layers with larger receptive fields are capable to detect larger scattered and diffuse patchy-like lesions. As depicted in Fig. 3, multiscale feature maps from Conv3, Conv4, and Conv5 are mapped down to the class score maps by $1 \times 1$ convolution. Followed with a Global Max Pooling operation, the class score maps are transformed into categorical score vectors. By summing up three categorical score vectors, we can make a final prediction with a Softmax activation function. Then a cross-entropy loss

$$\mathcal{L}_{\text{ce}} = -\sum_i^C y_i \log(x_i) \tag{1}$$

is utilized for the classification task, where $C$ is the number of classes, $y_i$ and $x_i$ are the ground truth (class) and the predicted class score of class $i$, respectively.

However, as aforementioned, the COVID-19 infections share very similar common features with CAP, for example, the GGO as well as the airspace consolidation, both of them distribute
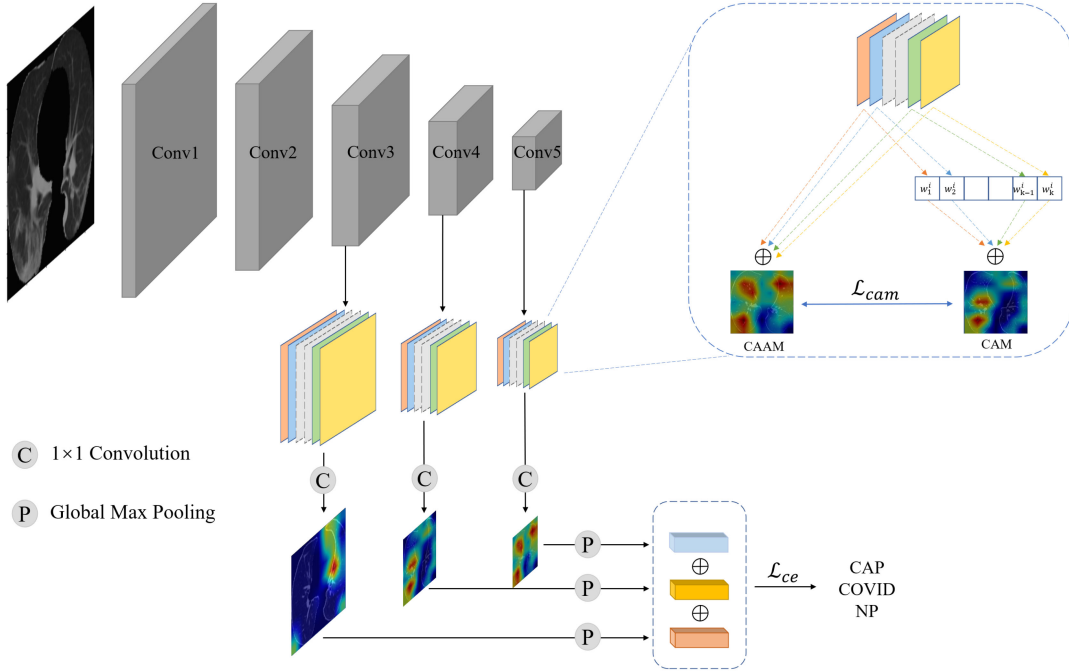
Fig. 3. Backbone architecture and $\mathcal{CAM}_{loss}$. The dashed box area (encoder and classifier) in Fig. 1 and Fig. 2.

bilaterally in the lung periphery. Thus [25] only achieved suboptimal performance on telling COVID apart from CAP. To better distinguish COVID-19 from CAP, we introduce a multiscale $\mathcal{CAM}_{loss}$ inspired by [34], [40].

For a given image, we take class score maps from Conv5 as an example. The vanilla CAM for class $i$ is a weighted sum of the feature maps, which can be formulated as

$$\text{CAM}_i = \sum_k w_k^i f_k, \tag{2}$$

where $f_k$ is the $k^{th}$ feature map, and $w_k^i$ represents the weight corresponding to class i for $f_k$. While CAAM is a direct summation of all feature maps, it is given by

$$\text{CAAM} = \sum_k f_k. \tag{3}$$

For a target category, CAAM generally shows richer features and larger areas than $\text{CAM}_i$ which only localizes partial regions of interest. However the redundant features in CAAM may result in higher confidence scores of non-target categories and lead to misclassification. To enforce intra-class compactness and inter-class separability, we constrain CAAM closer to $\text{CAM}_i$ by a loss $\mathcal{L}_{\text{cam}}$. The $l_1$ distance is employed in $\mathcal{L}_{\text{cam}}$ to measure the distance between CAAM and $\text{CAM}_i$, driving the backbone to capture target category features and suppress features of the non-target categories. This loss is defined as

$$\mathcal{L}_{\text{cam}} = \|\text{CAAM}' - \text{CAM}_i'\|_{l_1}, \tag{4}$$

where $\text{CAAM}'$ and $\text{CAM}_i'$ are min-max normalized CAAM and $\text{CAM}_i$. Besides Conv5 level, we also apply $\mathcal{L}_{\text{cam}}$ on Conv3 and Conv4 level to formulate a multiscale $\mathcal{CAM}_{loss}$. For the classification task, the final formal multiscale $\mathcal{CAM}_{loss}$ can be formulated as

$$\mathcal{CAM}_{loss} = \sum_s \alpha_s \mathcal{L}_{\text{cam}}^s + \mathcal{L}_{\text{ce}}, \tag{5}$$

where $\mathcal{L}_{\text{cam}}^s$ indicates $\mathcal{L}_{\text{cam}}$ at $s^{th}$ scales ($s = 1, \ldots, 3$, i.e., the Conv3, Conv4, Conv5 levels), and $\alpha_s$ represents the corresponding combination ratio.

### D. Saliency Map

How to generate a satisfactory pseudo label for segmentation? The most straightforward way is simply employing the confidence thresholded decoder prediction from a trained segmentation model [41]. However, this kind of pseudo label is barely desirable when training with limited labelled data and thus yield inferior performance [42]. Intuitively, we can solve this issue by combining multiple sources of prediction maps with the decoder prediction as compensation and further generating a well-calibrated pseudo label. Here, we employ the refined class activation map $\text{CAAM}_l$ and saliency map as the compensation.

The $\text{CAAM}_l$ is able to produce a more discriminative and locally consistent mask for segmentation but is still barely satisfactory in delineating crisp boundaries of the infection areas since it is obtained from low-resolution feature maps. While it is possible to generate high-resolution maps with subtle boundaries by saliency maps [37]. Here we employ Integrated Gradients [43] to obtain the saliency map. Since it is applicable to a variety of deep networks and can be implemented using a few calls to the gradients operator and has a strong theoretical justification.

Formally, let function $F$ represent the network and suppose we have a flattened input image denoted as $x = (x_1, \ldots, x_n) \in \mathbb{R}^n$

as well as a baseline input (black image filled with zeros in our case) $x' = (x'_1, \ldots, x'_n) \in \mathbb{R}^n$ where n is the number of pixels. Then $a_i$, the contribution of individual pixel $x_i$ to the prediction $F(x)$ relative to the baseline input $x'_i$, can be written as a vector $\mathbf{A}_F(x, x') = \frac{\partial F(x)}{\partial x} = (a_1, \ldots, a_n) \in \mathbb{R}^n$, which is also the gradient of $F(x)$ at $x$. Next, integrated gradients can be obtained by cumulating all gradients computed at each point along the straight-line path (in $\mathbb{R}^n$) from the baseline $x'$ to the input $x$. For the $i^{th}$ dimension of an input $x$ and baseline $x'$, the integrated gradient is given by

$$\text{InGrads}_i(x) = (x_i - x'_i) \times \int_{\alpha=0}^{1} \frac{\partial F(x' + \alpha(x - x'))}{\partial x_i} d\alpha, \tag{6}$$

where $\frac{\partial F(x)}{\partial x_i}$ is the gradient of $F(x)$ along the $i^{th}$ dimension. For the sake of computational efficiency, the integral of integrated gradients can be approximated via a summation as

$$\text{InGrads}_i(x) \approx (x_i - x'_i) \times \sum_{g=1}^{t} \frac{\partial F(x' + \frac{g}{t} \times (x - x'))}{\partial x_i} \times \frac{1}{t}, \tag{7}$$

where $t$ is the number of steps in Riemann's approximation of the integral and in practice, we compute the gradient in a for loop over the set of inputs (i.e., $g = 1, \ldots, t$). To obtain the saliency map, we directly use integrated gradients computed from the input CT scan and the corresponding multiscale classification output.

### E. Pseudo Label Calibration for Segmentation

Given the $\text{CAAM}_l$ $c$, the saliency map $s$, and the decoder prediction $p$, how to fuse them together for a better pseudo label $\tilde{y}$? Inspired by the fusion method proposed in PseudoSeg [42], we design a sharpen combination module $S$ to calibrate the pseudo label. The $S$, which consists of two key operations: Norm and Sharpen, can be formulated as

$$S(c, s, p) = \text{Sharpen} \left\{ \lambda \text{Softmax} \left( \frac{c}{\text{Norm}(c, s, p)} \right) \right.$$
$$+ \mu \text{Softmax} \left( \frac{s}{\text{Norm}(c, s, p)} \right)$$
$$\left. + \nu \text{Softmax} \left( \frac{p}{\text{Norm}(c, s, p)} \right), T \right\}, \tag{8}$$

where $\lambda + \mu + \nu = 1$. In view of $c$, $s$, and $p$ could have quite different degrees of overconfidence since they come from different decision mechanisms. We use a normalization operation

$$\text{Norm}(c, s, p) = \sqrt{\sum_{i}^{|c|} (c_i^2 + s_i^2 + p_i^2)}, \tag{9}$$

to ease the overconfident probability with Softmax and thus avoid the unbalanced term in $S$. Next, the distribution sharpening operation

$$\text{Sharpen}(a, T)_i = \frac{a_i^{1/T}}{\sum_j^C a_j^{1/T}}, \tag{10}$$

---

**Algorithm 1:** Training process.

**Initialization:** the parameters of model $\theta$;
**Optimization:**
1:   **for** number of training iterations **do**
2:      Calculate CAM, CAAM, and Saliency Map
3:      **if** performance $<$ baseline **then**
4:        Update $\theta$ with $\nabla_\theta (\mathcal{CAM}_{loss}{}^5, \mathcal{L}_s{}^{11})$
5:      **end if**
6:      Update $\theta$ with $\nabla_\theta (\mathcal{CAM}_{loss}{}^5, \mathcal{L}_s{}^{11}, \mathcal{L}_u{}^{12})$
7:   **end for**
8:   **return** optimal $\theta$

---

adjusts the categorical distribution [44] and produces the final calibrated pseudo label $\tilde{y}$. Here $T$ is a hyperparameter which is called the temperature scalar, it is used to control the categorical distribution.

For segmentation, our training procedure contains a supervised loss $\mathcal{L}_s$ for pixel-level labelled data $\mathcal{D}_s$ and a consistency regularization loss $\mathcal{L}_u$ for unlabelled data $\mathcal{D}_u$. The $\mathcal{L}_s$, a weighted binary cross-entropy loss function, is given by

$$\mathcal{L}_s = -y \times \log(F_\theta(x)) - w(1 - y) \times \log(1 - F_\theta(x)), \tag{11}$$

where $\theta$ represents the learnable parameters of the network function $F$, $x$ is the input, $y$ is the corresponding ground truth (infection mask), and $w$ is the weight coefficient set to cope with the imbalanced problem. With regards to the unsupervised counterpart, our method generates a pseudo label $\tilde{y}$ for every strongly augmented unlabelled data $x$ from $\mathcal{D}_u$. Following the consistency training technique [45], we design a consistency regularization loss based on weighted binary cross-entropy loss function, which can be written as

$$\mathcal{L}_u = -\tilde{y} \times \log(F_\theta(\Theta(x))) - w(1 - \tilde{y}) \times \log(1 - F_\theta(\Theta(x))), \tag{12}$$

where $\Theta$ indicates the strong data augmentation operation.

Finally, the training objective consists of the classification loss $\mathcal{CAM}_{loss}$, the segmentation loss $\mathcal{L}_s$, and the consistency regularization loss $\mathcal{L}_u$. All of the loss terms are optimized jointly with corresponding coefficients (i.e., $\beta \mathcal{CAM}_{loss} + \gamma \mathcal{L}_s + \eta \mathcal{L}_u$), while $\mathcal{L}_u$ is incorporated only after the $\mathcal{CAM}_{loss}$ and $\mathcal{L}_s$ are stable (i.e., until the classification and segmentation results become acceptable, which means the results are comparable with baseline models). The training process is summarized in Algorithm 1.

### F. Explanation

Our method employs the $\text{CAAM}_l$ and the saliency map as explanations to increase the explainable transparency of the identification and delineation of the COVID-19 infections. As aforementioned, the $\text{CAAM}_l$, refined by $\mathcal{CAM}_{loss}$, is capable to perform better in suppressing non-target features and capturing target features than vanilla CAM. It learns more globally discriminative and locally consistent features of the lesion area. While the multiscale saliency map can provide more subtle boundaries in the lesion area. Both the $\text{CAAM}_l$ and the saliency
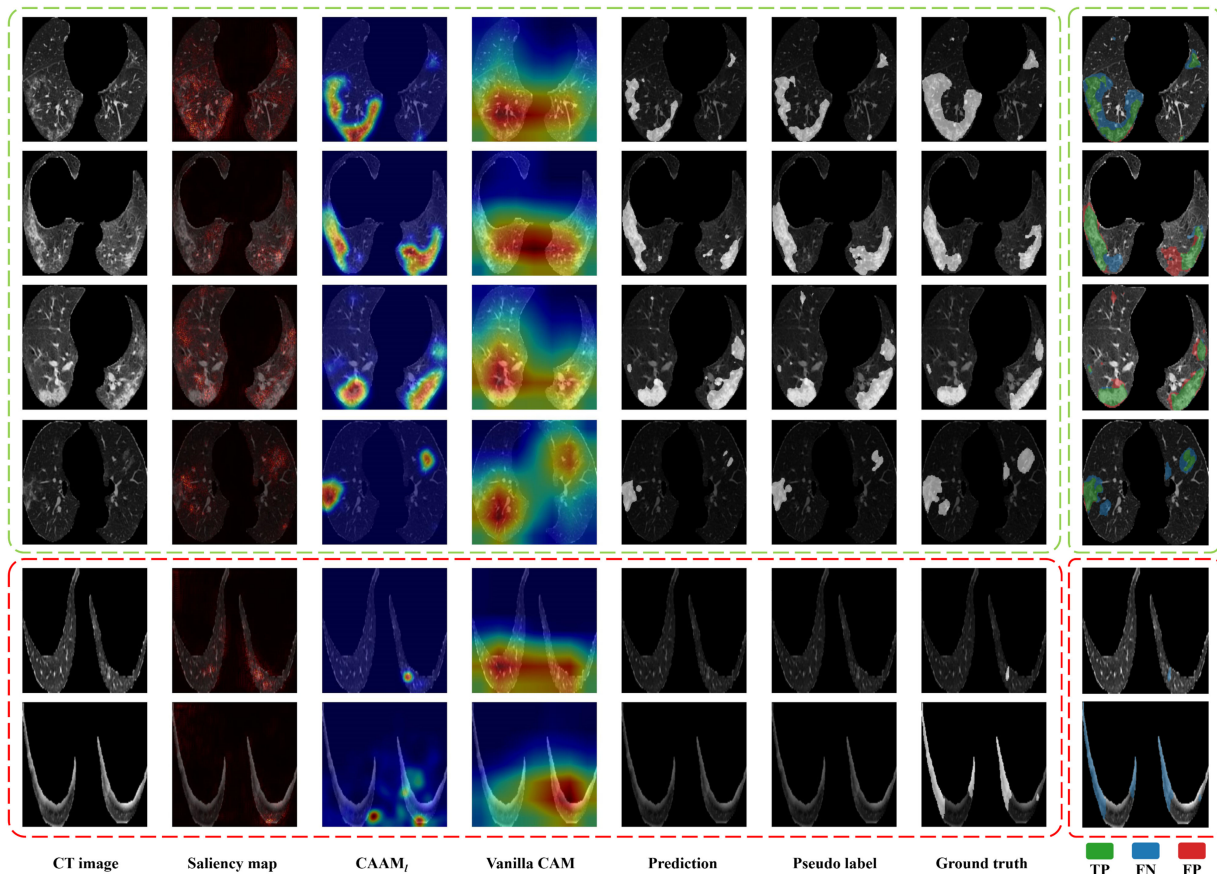
This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

6             IEEE TRANSACTIONS ON EMERGING TOPICS IN COMPUTATIONAL INTELLIGENCE

Fig. 4. Visualizations of the CT images and the corresponding saliency maps, $CAAM_l$, vanilla CAM, decoder prediction, pseudo label as well as the ground truth (infection mask). By combining saliency map, $CAAM_l$, and decoder prediction, we get the calibrated pseudo label. The last column shows the true positive (TP), false negative (FN) and false positive (FP) visualizations in different colors. (Top four rows in the green dashed box: good cases; Bottom two rows in the red dashed box: bad cases).

map were refined together with the decoder prediction by the sharpen combination module during the learning procedure. Finally, the trained model is able to provide convincing explanation results. Fig. 4 shows some visualization results (both good and bad cases).

## IV. EXPERIMENTS AND DISCUSSION

In this section, we detail the conducted experiments to validate our proposed method. We start by specifying the data, implementation, and evaluation details. Next, we conduct comparison studies in the supervised setting and the semi-supervised setting, respectively. Lastly, we evaluate our model design choices through various ablation studies.

### A. Data Description

We use an open dataset [46] with pixel-level annotations of lungs and infections made by experts to train our model. This open dataset contains 20 CT scans (4,151 slices) of patients diagnosed with COVID-19 as well as annotations made by experts. Besides, we collected 359 patient CT scans (60,693 slices) with patient-level labels from the Hospital of Wuhan Red Cross Society (WHRCH) and Shenzhen Second Hospital (SZSH) between September 2016 and March 2020. Our private dataset

contains 120, 124, 115 chest CT scans of COVID-19, CAP, and NP patients, respectively. All COVID-19 patients, scanned within 3 days of hospitalisation from December 2019 to March 2020, were tested positive by RT-PCR and confirmed at severe or critical stage according to the diagnosis and treatment guidelines [47] of COVID-19 issued by the Chinese National Health Commission. Both CAP and NP patients were from SZSH, for COVID-19 patients, 108 of them were from WHRCH, the other 12 patients were also from SZSH. Details are shown in Table I.

Specifically, 179 cases (29,964 slices, NP: 55 cases, CAP: 64 cases, COVID-19: 60 cases) from our private dataset and 13 cases (2,815 slices, COVID-19: 13 cases) from the open dataset are combined as the full training set (32,779 slices, NP: 55 cases, CAP: 64 cases, COVID-19: 73 cases). The remaining 180 cases (30,729 slices, NP: 60 cases, CAP: 60 cases, COVID-19: 60 cases) in our private dataset and 7 cases (1,336 slices, COVID-19: 7 cases) in the open dataset consists of the full testing set (32,065 slices, NP: 60 cases, CAP: 60 cases, COVID-19: 67 cases). See more details about data split in Table II.

### B. Implementation Details and Evaluation Metrics

All CT slices are resampled to $1 \times 1 \times 1\,mm$ and resized to $224 \times 224$ for computational efficiency. The proposed model

TABLE I

DETAIL DESCRIPTIONS AND IMAGING PARAMETERS OF THE CT SYSTEMS OF BOTH OUR PRIVATE DATASET AND OPEN DATASET

| Centre | SZSH | | | WHRCH | Open Dataset [46] |
|---|---|---|---|---|---|
| Category | NP | CAP | COVID-19 | COVID-19 | COVID-19 |
| Infection label (pixel-level) | $\times$ | $\times$ | $\times$ | $\times$ | $\checkmark$ |
| Cases | 115 | 124 | 12 | 108 | 20 |
| Slice number (resampled) | 23,931 | 22,989 | 1,939 | 11,834 | 4,151 |
| Scanner | SIEMENS SOMATOM Emotion | | GE revolution 256 | SIEMENS SOMATOM go.Now16 | - |
| Tube voltage | 110 $KV$ | | 120 $KV$ | 130 $KV$ | - |
| Slice thickness | 1.2 $mm$ | | 0.625 $mm$ | 0.7 $mm$ | - |
| Reconstructed slice thickness | 1.5 $mm$ | | 2 $mm$ | 1 $mm$ | $(1, 1.5, 2, 4, 6)$ $mm$ |
| Pitch | 1.2 | | 1.375 | 1.5 | - |
| Matrix | $512 \times 512$ | | $512 \times 512$ | $512 \times 512$ | $512 \times 512$ |
| Field of view | 260 $mm \times$ 260 $mm$ | | 400 $mm \times$ 400 $mm$ | 350 $mm \times$ 350 $mm$ | - |
| Automatic tube current modulation | 70 $mAs$ | | 150 $mAs$ | 50 $mAs$ | - |

TABLE II

DETAILS OF THE DATA SPLIT (NUMBERS ARE SHOWN AS X CASES (Y SLICES)). IN THE LABELLED TRAINING SET, WE HAVE PATIENT-LEVEL CATEGORY LABELS AND PIXEL-LEVEL INFECTION LABELS (NP/CAP SLICES HAVE ALL ZERO SEGMENTATION MASKS AND ARE UNIFORMLY SAMPLED FROM ALL NP/CAP CASES IN THE FULL TRAINING SET), WHILE IN THE UNLABELLED TRAINING SET, THERE ARE NO LABELS AVAILABLE AT ALL DURING TRAINING

| Sets | Total | Private | | | Open |
|---|---|---|---|---|---|
| | | CAP | NP | COVID-19 | COVID-19 |
| Full dataset | 379 (64,844) | 124 (22,989) | 115 (23,931) | 120 (13,773) | 20 (4,151) |
| Full training set | 192 (32,779) | 64 (12,250) | 55 (11,025) | 60 (6,689) | 13 (2,815) |
| Full testing set | 187 (32,065) | 60 (10,739) | 60 (12,906) | 60 (7,084) | 7 (1,336) |
| Labelled training set | 132 (7,815) | 64 (2,500) | 55 (2,500) | - | 13 (2,815) |
| Unlabelled training set | 179 (24,964) | 64 (9,750) | 55 (8,525) | 60 (6,689) | - |
| Classification testing set | 187 (32,065) | 60 (10,739) | 60 (12,906) | 60 (7,084) | 7 (1,336) |
| Segmentation testing set | 7 (1,336) | - | - | - | 7 (1,336) |

is trained for 300 epochs on a Linux workstation with four Nvidia RTX 3090 GPUs using PyTorch 1.10. We adopt the Adam optimiser ($\beta_1 = 0.5$, $\beta_2 = 0.9$) with a initial learning rate $10^{-4}$. The learning rate is updated by multiplying by 0.1 in every 20 epochs. We employ contrast adjustment and sharpness adjustment as strong data augmentation methods. All $\alpha_s$ in $\mathcal{CAM}_{loss}$ are set to 5. The $\lambda$, $\mu$, and $\nu$ in sharpen combination module are 0.3, 0.4, 0.4, respectively. And $T$ in the sharpening operation is set to 0.5. As for the coefficients in the final loss function, $\beta$, $\gamma$, and $\eta$ are set to 1, 5, 5 separately.

With regard to evaluation metrics, for classification performance evaluation, we use accuracy, the area under the receiver operating characteristic curve (AUC), sensitivity, and specificity as suggested by [49]. For segmentation performance evaluation, we employ two standard metrics, dice score and mean intersection over union (mIoU).

### C. Comparison Study

We compare our method with other methods in two training settings (supervised, semi-supervised). In the supervised setting, we select a subset from the whole training set described in IV-A to make up the labelled training dataset (2,500 CAP slices and 2,500 NP slices from our private dataset, as well as 2,815 COVID-19 slices from the open dataset). In the labelled training dataset, both CAP and NP slices have patient-level category labels and all zero segmentation masks since they are not

COVID-19 cases, while all COVID-19 slices have patient-level category labels and pixel-level infection labels. As for the semi-supervised setting, in addition to the labelled training dataset, we use the rest of the data in the full training set described in IV-A without any labels to serve as the unlabelled training dataset for training. Both supervised and semi-supervised settings share the same test datasets, 7 COVID-19 cases (1,336 slices) with pixel-level infection labels for segmentation performance test, as well as 187 cases (32,065 slices, NP: 60 cases, CAP: 60 cases, COVID-19: 60 private cases and 7 open cases) with patient-level category labels for classification performance test. See more details about data split in Table II.

For classification performance comparison, we consider four supervised methods: VGG-16, ResNet-18, JCS [21], Hu *et al.* [25], and two semi-supervised methods: Pseudo-Label [48], MultiMix [24]. We use the classification performance of VGG-16 to serve as the supervised classification baseline. For segmentation performance comparison, we consider three supervised methods: SegNet, U-Net, JCS [21], and two semi-supervised methods: PseudoSeg [42], MultiMix [24]. We employ the segmentation performance of SegNet as the supervised segmentation baseline.

All detailed quantitative classification and segmentation comparison performances are reported in Table III and Table IV, respectively. We also provide some qualitative visual comparisons of all methods in Fig. 5. In the supervised setting, our method achieves comparable classification performance with

TABLE III
CLASSIFICATION PERFORMANCE COMPARISON AMONG DIFFERENT METHODS

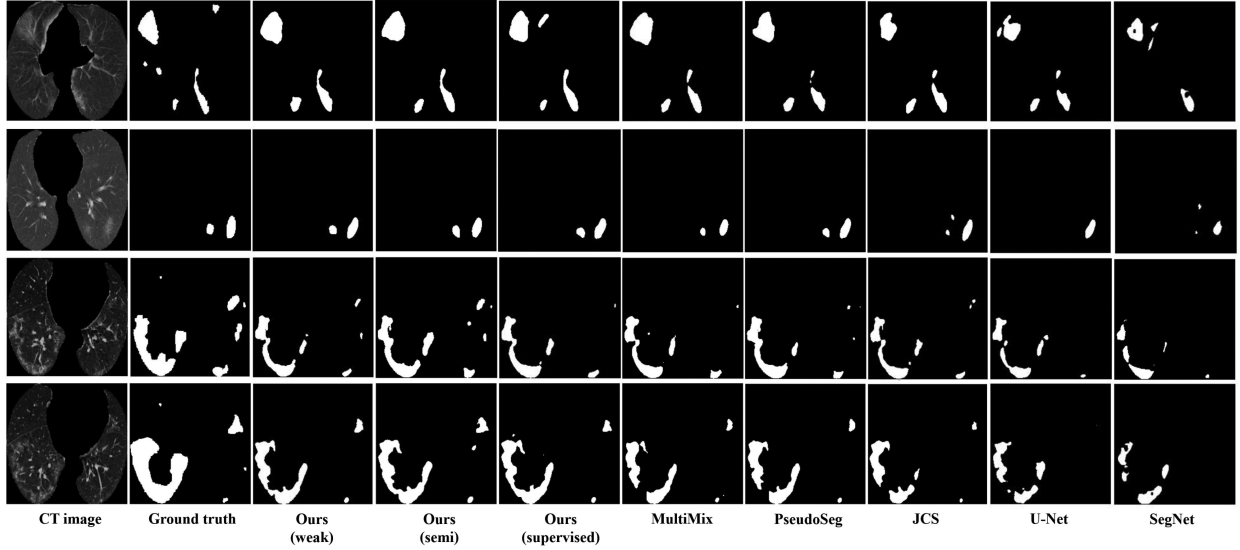| Method | Accuracy(%) | Sensitivity(%) | Specificity(%) | AUC |
|---|---|---|---|---|
| VGG-16 (baseline)† | 70.65 | 72.93 | 84.65 | 0.8600 |
| JCS [21]† | 71.90 | 74.65 | 85.58 | 0.8548 |
| ResNet-18† | 71.42 | 73.68 | 84.91 | 0.8689 |
| Hu et al. [25]† | 71.05 | 73.45 | 84.93 | 0.8703 |
| Pseudo-Label [48]⋆ | 72.31 | 74.76 | 86.02 | 0.8810 |
| MultiMix [24]⋆ | 72.91 | 74.37 | 85.69 | 0.8701 |
| Ours† | 71.40 | 73.82 | 85.54 | **0.8851** |
| Ours⋆ | **75.13** | **76.35** | **86.89** | 0.8748 |

† :supervised, ⋆ :semi-supervised.



Fig. 5. Visual comparisons of different methods for COVID-19 infection lesions segmentation.

TABLE IV
SEGMENTATION PERFORMANCE COMPARISON AMONG DIFFERENT METHODS

| Method | Dice(%) | mIoU(%) |
|---|---|---|
| SegNet (baseline)† | 75.64 | 67.31 |
| U-Net† | 78.90 | 70.52 |
| JCS [21]† | 79.71 | 71.36 |
| PseudoSeg [42]⋆ | 82.88 | 73.99 |
| MultiMix [24]⋆ | 84.80 | 76.17 |
| Ours† | 84.53 | 75.90 |
| Ours⋆ | **85.49** | **76.97** |

† :supervised, ⋆ :semi-supervised.

supervised counterparts (JCS, ResNet-18, and Hu *et al.*) and surpasses the supervised baseline (VGG-16) in all metrics. Our method also gain the best segmentation performance compared with all supervised methods and the supervised baseline in both dice and mIoU. In the semi-supervised setting, our method still achieves superior classification (75.13% accuracy, 76.35% sensitivity, 86.89% specificity, and 0.8748 AUC) and segmentation results (85.49% dice, 76.97% mIoU) compared with all the other methods and baselines. Additionally, from Fig. 5, we can get a more straight observation of the segmentation performance. In contrast to other methods, our method shows superior segmentation results in both supervised and semi-supervised settings.

These quantitative and qualitative results demonstrate that our method is able to make good use of the unlabelled dataset and obtain superior performance with limited labelled data.

### D. Ablation Study

To validate the effectiveness of every proposed component in our method, we perform an extensive ablation study. Concretely, we remove the $\mathcal{CAM}_{loss}$ (only keep $\mathcal{L}_{ce}$ for classification task), saliency map, sharpen combination module, and the multiscale learning strategy from our method, respectively (expressed as w/o in Table V). Then we test their performance in both semi-supervised and weakly-supervised settings. With regard to the weakly-supervised setting, we add patient-level category labels to the unlabelled training dataset for training and want to know whether patient-level category labels can provide performance gain or not.

The ablation study results are reported in Table V. As we can see, every component does contribute to the performance, especially in the semi-supervised setting. The full version of our method achieves the best classification and segmentation results in all metrics. The results under the weakly-supervised setting also illustrate a similar conclusion. Whereas there is a little difference in the segmentation performance. We can observe that,

TABLE V
ABLATION STUDY RESULTS (PATIENT-LEVEL LABELS: - INDICATES SEMI-SUPERVISED SETTING; ✓ REPRESENTS WEAKLY-SUPERVISED SETTING)

| Method | patient-level labels | Classification | | | | Segmentation | |
|---|---|---|---|---|---|---|---|
| | | Accuracy(%) | Sensitivity(%) | Specificity(%) | AUC | Dice(%) | mIoU(%) |
| w/o cam loss | - | 73.66 | 75.53 | 86.26 | 0.8633 | 80.53 | 71.62 |
| w/o saliency map | - | 73.80 | 74.21 | 85.81 | 0.8634 | 84.09 | 75.41 |
| w/o sharpen | - | 74.65 | 75.90 | 86.63 | 0.8597 | 82.14 | 73.28 |
| w/o multiscale | - | 74.78 | 76.23 | 86.72 | 0.8661 | 82.69 | 73.90 |
| full | - | **75.13** | **76.35** | **86.89** | **0.8748** | **85.49** | **76.97** |
| w/o cam loss | ✓ | 74.85 | 76.32 | 86.77 | 0.8681 | 82.82 | 74.05 |
| w/o saliency map | ✓ | 74.88 | 76.49 | 86.84 | 0.8690 | 82.57 | 73.82 |
| w/o sharpen | ✓ | 75.54 | 77.01 | 87.15 | 0.8817 | 83.35 | 74.58 |
| w/o multiscale | ✓ | 75.00 | 76.16 | 86.75 | 0.8745 | **84.64** | **76.05** |
| full | ✓ | **77.22** | **78.01** | **87.97** | **0.8948** | 83.67 | 74.92 |

when adding the patient-level category label in the unlabelled training dataset, the full version of our method still obtains the best classification performance but achieve slightly lower segmentation performance compared with the w/o multiscale version. We deem that this is caused by the coarse patient-level category labels. In COVID-19 cases, not all slices are infected, thus there are no infections in this kind of slices. These slices are actually as 'healthy' as NP and CAP cases but assigned with COVID-19 category labels. This brings noisy labels to our model and influences the performances. While the number of this kind of slice is not large for the classification task, we can see the classification performance is still the best. However they can be fatal for segmentation tasks since the number of labelled infections is limited and the quality of pseudo label relies on the saliency map and CAAM$_l$, which are controlled by the classification results. The visual comparisons in Fig. 5 also show that the segmentation results under weakly-supervised settings are inferior to those under supervised or semi-supervised settings, which confirms our deduction.

To further support the results in Table V, we also plot the visualizations of the CT images and the corresponding saliency maps, CAAM$_l$, vanilla CAM, decoder prediction, pseudo label as well as the ground truth in Fig. 4. As illustrated, by fusing saliency map, CAAM$_l$ and decoder prediction through the sharpen combination module, we can obtain reliable pseudo labels. Moreover, $\mathcal{CAM}_{loss}$ does bring benefits in explainability, we can see a clear improvement from vanilla CAM to CAAM$_l$ and the activation areas match the infection areas in ground truth well. Besides, we want to discuss more about the bad cases, as we can see in the bottom two rows, our model is not able to segment the infection areas. The main reasons here are twofold in these kinds of cases: firstly, the lung or the infection areas are too small for our model to produce accurate segmentation predictions and pseudo labels; secondly, the number of such kind of peripheral slices is quite limited during training procedure.

Lastly, we look more closely at the timing of using $\mathcal{CAM}_{loss}$, i.e., finding the proper time to add $\mathcal{L}_{cam}$ to $\mathcal{L}_{ce}$. As shown in Fig. 6, the most suitable timing is the $20^{th}$ epoch, where the model gains the best AUC. At that moment, the vanilla CAM has already learnt obvious target category features, and it is a good time to use $\mathcal{CAM}_{loss}$ to enforce intra-class compactness and inter-class separability and refine the activation areas.
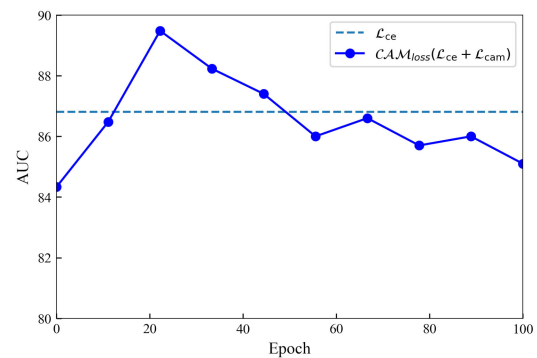


Fig. 6. Ablation study of timing by using $\mathcal{CAM}_{loss}$ in the first 100 epochs. The dashed line indicates the performance of only using $\mathcal{L}_{ce}$ for the classification task.

## V. CONCLUSION

In this paper, we propose a semi-supervised multitask explainable identification and delineation method for COVID-19 infections, producing classification, segmentation and explainable visualizations at the same time. We demonstrate the effectiveness of our method with the combination of limited labelled data and unlabelled data or weakly-labelled data. The key behind the performance gain is two-fold: Firstly, the $\mathcal{CAM}_{loss}$ helps to enforce intra-class compactness and inter-class separability, further bringing benefits in explainability. Secondly, we design a calibrated pseudo-labelling strategy and apply it under the consistency regularization framework, which only uses limited infection labels.

Further exploration of patient-level category label issue, strong data augmentation method, the overall modest classification results caused by domain gap among centres worth more study as future directions.

## REFERENCES

[1] B. Hu, H. Guo, P. Zhou, and Z.-L. Shi, "Characteristics of SARS-CoV-2 and COVID-19," *Nature Rev. Microbiol.*, vol. 19, no. 3, pp. 141–154, 2021.

[2] A. F. Rendeiro *et al.*, "The spatial landscape of lung pathology during COVID-19 progression," *Nature*, vol. 593, no. 7860, pp. 564–569, 2021.

[3] T. Ai *et al.*, "Correlation of chest CT and RT-PCR testing for coronavirus disease 2019 (COVID-19) in China: A report of 1014 cases," *Radiology*, vol. 296, no. 2, pp. E32–E40, 2020.

[4] F. Khatami *et al.*, "A meta-analysis of accuracy and sensitivity of chest CT and RT-PCR in COVID-19 diagnosis," *Sci. Rep.*, vol. 10, no. 1, pp. 1–12, 2020.

[5] X. Xie *et al.*, "Chest CT for typical coronavirus disease 2019 (COVID-19) pneumonia: Relationship to negative RT-PCR testing," *Radiology*, vol. 296, no. 2, pp. E41–E45, 2020.

[6] C. Long *et al.*, "Diagnosis of the coronavirus disease (COVID-19): RRT-PCR or CT?," *Eur. J. Radiol.*, vol. 126, 2020, Art. no. 108961.

[7] X. Wu, C. Chen, M. Zhong, J. Wang, and J. Shi, "COVID-AL: The diagnosis of COVID-19 with deep active learning," *Med. Image Anal.*, vol. 68, 2021, Art. no. 101913.

[8] J. Cheng *et al.*, "Automated diagnosis of COVID-19 using deep supervised autoencoder with multi-view features from CT images," *IEEE/ACM Trans. Comput. Biol. Bioinf.*, 2021, to be published, doi: 10.1109/TCBB.2021.3102584.

[9] Y.-D. Zhang, Z. Zhang, X. Zhang, and S.-H. Wang, "MIDCAN: A multiple input deep convolutional attention network for COVID-19 diagnosis based on chest CT and chest x-ray," *Pattern Recognit. Lett.*, vol. 150, pp. 8–16, 2021.

[10] F. Ucar and D. Korkmaz, "COVIDiagnosis-Net: Deep Bayes-SqueezeNet based diagnosis of the coronavirus disease 2019 (COVID-19) from x-ray images," *Med. Hypotheses*, vol. 140, 2020, Art. no. 109761.

[11] L. Wynants *et al.*, "Prediction models for diagnosis and prognosis of COVID-19: Systematic review and critical appraisal," *BMJ*, vol. 369, 2020, Art. no. m1328.

[12] M. Roberts *et al.*, "Common pitfalls and recommendations for using machine learning to detect and prognosticate for COVID-19 using chest radiographs and CT scans," *Nature Mach. Intell.*, vol. 3, no. 3, pp. 199–217, 2021.

[13] M. Li *et al.*, "Recurrent aggregation learning for multi-view echocardiographic sequences segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.- Assist. Intervention*. Springer, 2019, pp. 678–686.

[14] M. Li *et al.*, "Unified model for interpreting multi-view echocardiographic sequences without temporal information," *Appl. Soft Comput.*, vol. 88, 2020, Art. no. 106049.

[15] X. Liu *et al.*, "Deep learning-based automated left ventricular ejection fraction assessment using 2D echocardiography," *Amer. J. Physiol.-Heart Circulatory Physiol.*, vol. 321, no. 2, pp. H390–H399, 2021.

[16] M. Li, C. Wang, H. Zhang, and G. Yang, "MV-RAN: Multiview recurrent aggregation network for echocardiographic sequences segmentation and full cardiac cycle analysis," *Comput. Biol. Med.*, vol. 120, 2020, Art. no. 103728.

[17] S. P. Adhikari *et al.*, "Epidemiology, causes, clinical manifestation and diagnosis, prevention and control of coronavirus disease (COVID-19) during the early outbreak period: A scoping review," *Infect. Dis. Poverty*, vol. 9, no. 1, pp. 1–12, 2020.

[18] S. A. Harmon *et al.*, "Artificial intelligence for the detection of COVID-19 pneumonia on chest CT using multinational datasets," *Nature Commun.*, vol. 11, no. 1, pp. 1–7, 2020.

[19] J. Wang *et al.*, "Prior-attention residual learning for more discriminative COVID-19 screening in CT images," *IEEE Trans. Med. Imag.*, vol. 39, no. 8, pp. 2572–2583, Aug. 2020.

[20] M. Owais, Y. W. Lee, T. Mahmood, A. Haider, H. Sultan, and K. R. Park, "Multilevel deep-aggregated boosted network to recognize COVID-19 infection from large-scale heterogeneous radiographic data," *IEEE J. Biomed. Health Informat.*, vol. 25, no. 6, pp. 1881–1891, Jun. 2021.

[21] Y.-H. Wu *et al.*, "JCS: An explainable COVID-19 diagnosis system by joint classification and segmentation," *IEEE Trans. Image Process.*, vol. 30, pp. 3113–3126, 2021.

[22] A. I. Aviles-Rivero, P. Sellars, C.-B. Schönlie, and N. Papadakis, "GraphX-COVID: Explainable deep graph diffusion pseudo-labelling for identifying COVID-19 on chest x-rays," *Pattern Recognit.*, vol. 122, 2022, Art. no. 108274.

[23] D.-P. Fan *et al.*, "Inf-net: Automatic COVID-19 lung infection segmentation from CT images," *IEEE Trans. Med. Imag.*, vol. 39, no. 8, pp. 2626–2637, Aug. 2020.

[24] A. Haque *et al.*, "Generalized multi-task learning from substantially unlabeled multi-source medical image data," *Mach. Learn. Biomed. Imag.*, vol. 1, pp. 1–10, 2021.

[25] S. Hu *et al.*, "Weakly supervised deep learning for COVID-19 infection detection and classification from CT images," *IEEE Access*, vol. 8, pp. 118869–118883, 2020.

[26] S. Tabik *et al.*, "COVIDGR dataset and COVID-SDNet methodology for predicting COVID-19 based on chest x-ray images," *IEEE J. Biomed. health Informat.*, vol. 24, no. 12, pp. 3595–3605, Dec. 2020.

[27] G. Yang, Q. Ye, and J. Xia, "Unbox the black-box for the medical explainable ai via multi-modal and multi-centre data fusion: A mini-review, two showcases and beyond," *Inf. Fusion*, vol. 77, pp. 29–52, 2022.

[28] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 2921–2929.

[29] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: Visual explanations from deep networks via gradient-based localization," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 618–626.

[30] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 818–833.

[31] Y. Zhu, Y. Zhou, Q. Ye, Q. Qiu, and J. Jiao, "Soft proposal networks for weakly supervised object localization," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 1841–1850.

[32] P.-J. Kindermans *et al.*, "The (un)reliability of Saliency Methods," in *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning*. Berlin, Germany: Springer, 2019, pp. 267–280.

[33] K. Baek, M. Lee, and H. Shim, "Psynet: Self-supervised approach to object localization using point symmetric transformation," in *Proc. AAAI Conf. Artif. Intell.*, 2020, vol. 34, no. 07, pp. 10451–10459.

[34] C. Wang, J. Xiao, Y. Han, Q. Yang, S. Song, and G. Huang, "Towards learning spatially discriminative feature representations," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 1326–1335.

[35] Q. Ye, J. Xia, and G. Yang, "Explainable AI for COVID-19 CT classifiers: An initial comparison study," in *Proc. IEEE 34th Int. Symp. Comput.-Based Med. Syst.*, 2021, pp. 521–526.

[36] J. Yang *et al.*, "Autosegmentation for thoracic radiation treatment planning: A grand challenge at AAPM 2017," *Med. Phys.*, vol. 45, no. 10, pp. 4568–4581, 2018.

[37] K. Simonyan, A. Vedaldi, and A. Zisserman, "Deep inside convolutional networks: Visualising image classification models and saliency maps," in *Proc. Workshop Int. Conf. Learn. Representations.*, 2014.

[38] F. Pan *et al.*, "Time course of lung changes at chest CT during recovery from coronavirus disease 2019 (COVID-19)," *Radiology*, vol. 295, no. 3, pp. 715–721, 2020.

[39] H. Shi *et al.*, "Radiological findings from 81 patients with COVID-19 pneumonia in wuhan, China: A descriptive study," *Lancet Infect. Dis.*, vol. 20, no. 4, pp. 425–434, 2020.

[40] S. Dong *et al.*, "Holistic and deep feature pyramids for saliency detection," in *Proc. BMVC*, 2018, p. 67.

[41] K. Sohn *et al.*, "Fixmatch: Simplifying semi-supervised learning with consistency and confidence," *Adv. Neural Inf. Process. Syst.*, vol. 33, pp. 596–608, 2020.

[42] Y. Zou *et al.*, "PseudoSeg: Designing pseudo labels for semantic segmentation," in *Proc. Int. Conf. Learn. Representations*, 2021.

[43] M. Sundararajan, A. Taly, and Q. Yan, "Axiomatic attribution for deep networks," in *Proc. Int. Conf. Mach. Learn.*, 2017, pp. 3319–3328.

[44] D. Berthelot, N. Carlini, I. Goodfellow, N. Papernot, A. Oliver, and C. A. Raffel, "Mixmatch: A holistic approach to semi-supervised learning," *Adv. Neural Inf. Process. Syst.*, vol. 32, 2019.

[45] Q. Xie, Z. Dai, E. Hovy, T. Luong, and Q. Le, "Unsupervised data augmentation for consistency training," *Adv. Neural Inf. Process. Syst.*, vol. 33, pp. 6256–6268, 2020.

[46] J. Ma *et al.*, "COVID-19 CT lung and infection segmentation dataset," in Zenodo, 2020, doi: 10.5281/zenodo.3757476.

[47] S.-X. Wang *et al.*, "Diagnosis and treatment of novel coronavirus pneumonia based on the theory of traditional chinese medicine," *J. Integrative Med.*, vol. 18, no. 4, pp. 275–283, 2020.

[48] D.-H. Lee, "Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks," *Proc. Workshop Challenges Representation Learn. ICML*, vol. 3, no. 2, p. 896, 2013.

[49] Y. Liu, Yu-H. Wu, Y. Ban, H. Wang, and M.-M. Cheng, "Rethinking computer-aided tuberculosis diagnosis," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 2646–2655.