Informatics in Medicine Unlocked 28 (2022) 100835

Contents lists available at ScienceDirect

Informatics in Medicine Unlocked

journal homepage: www.elsevier.com/locate/imu

Indirect supervision applied to COVID-19 and pneumonia classification

Viacheslav V. Danilov [a,b,*], Alex Proutski [c], Alex Karpovsky [d], Alexander Kirpich [e], Diana Litmanovich [f], Dato Nefaridze [c], Oleg Talalov [c], Semyon Semyonov [c], Vladimir Koniukhovskii [g], Vladimir Shvartc [g], Yuriy Gankin [c,**]

[a] *Tomsk Polytechnic University, Tomsk, Russia*
[b] *Research Institute for Complex Issues of Cardiovascular Diseases, Kemerovo, Russia*
[c] *Quantori, Cambridge, MA, United States*
[d] *Kanda Software, Newton, MA, United States*
[e] *Georgia State University, Atlanta, GA, United States*
[f] *Beth Israel Deaconess Medical Center, Boston, MA, United States*
[g] *EPAM Systems, Saint Petersburg, Russia*

ARTICLE INFO

Keywords:
COVID-19
Pneumonia
Classification
Indirect supervision
Deep learning
Transfer learning

ABSTRACT

The novel coronavirus 19 (COVID-19) continues to have a devastating effect around the globe, leading many scientists and clinicians to actively seek to develop new techniques to assist with the tackling of this disease. Modern machine learning methods have shown promise in their adoption to assist the healthcare industry through their data and analytics-driven decision making, inspiring researchers to develop new angles to fight the virus. In this paper, we aim to develop a CNN-based method for the detection of COVID-19 by utilizing patients' chest X-ray images. Developing upon the inclusion of convolutional units, the proposed method makes use of indirect supervision based on Grad-CAM. This technique is used in the training process where Grad-CAM's attention heatmaps support the network's predictions. Despite recent progress, scarcity of data has thus far limited the development of a robust solution. We extend upon existing work by combining publicly available data across 5 different sources and carefully annotate the comprising images across three categories: normal, pneumonia, and COVID-19. To achieve a high classification accuracy, we propose a training pipeline based on indirect supervision of traditional classification networks, where the guidance is directed by an external algorithm. With this method, we observed that the widely used, standard networks can achieve an accuracy comparable to tailor-made models, specifically for COVID-19, with one network in particular, VGG-16, outperforming the best of the tailor-made models.

1. Introduction

Since its introduction into the human population in late 2019, COVID-19 continues to have a devastating effect on the global populace with the number of infected individuals steadily rising [1]. With widely available treatments still outstanding and the continued strain placed on many healthcare systems across the world, efficient screening of suspected COVID-19 patients and their subsequent isolation is of paramount importance to mitigate further spread of the virus. Presently, the accepted gold standard for patient screening is reverse transcriptase-polymerase chain reaction (RT-PCR) where the presence of COVID-19 is inferred from the analysis of respiratory samples [2]. Despite its success, RT-PCR is a highly involved manual process with slow turnaround times, with results becoming available up to several days after the test is performed. Furthermore, its variable sensitivity, lack of standardized reporting, and a widely ranging total positive rate [3–5] calls for alternative screening methods.

Chest radiography imaging (such as X-ray or computed tomography (CT) imaging) has gained traction as a powerful alternative, where the diagnosis is administered by expert radiologists who analyze the resulting images and infer the presence of COVID-19 through subtle visual cues [6–10]. Of the two imaging methods studied, X-ray imaging has distinct advantages with regards to accessibility, availability, and rate of testing [11]. Furthermore, the existence of portable X-ray

* Corresponding author. Tomsk Polytechnic University, Tomsk, Russia.
** Corresponding author. Quantori, Cambridge, MA, United States.
E-mail addresses: viacheslav.v.danilov@gmail.com (V.V. Danilov), yuriy.gankin@quantori.com (Y. Gankin).

https://doi.org/10.1016/j.imu.2021.100835
Received 2 November 2021; Received in revised form 11 December 2021; Accepted 23 December 2021
Available online 28 December 2021

imaging systems does not require patient transportation or physical contact between healthcare professionals and suspected infected individuals, thus allowing for efficient virus isolation and a safer testing methodology. Despite its obvious promise, the main challenge facing radiography examination is the scarcity of trained experts that could conduct the analysis at a time when the number of possible patients continues to rise. As such, a computer system that could accurately analyze and interpret chest X-ray images could significantly alleviate the burden placed on expert radiologists and further streamline patient care. Image identification techniques are readily adopted in Artificial Intelligence (AI) and could prove to be a powerful solution to the problem at hand.

Deep learning models, such as convolutional neural networks (CNNs), have gained traction in the field of medical imaging [12,13] and here we train 10 promising CNNs for the purpose of COVID-19 classification in chest X-ray images. To assist the models, we utilize a purpose-built extraction of a soft mask as part of a three-stage procedure. To better quantify the performance of our proposed framework we benchmark our results against recently developed COVID-Net models [14]. To ensure consistency, we utilize our dataset to output predictions across an array of different COVID-Net models.

The structure of the rest of this paper is as follows: section “Related Work” briefly discusses some of the existing work used to diagnose COVID-19 in radiographic imaging; section “Data” summarizes the data collected from the 5 most studied datasets; section “Methods” describes the proposed three-stage workflow using an indirect attention mechanism; section “Results” displays the results obtained during all 3 stages, outlines further improvements of the proposed workflow, its advantages over other models and showcases possible implementations; section “Conclusion” represents a synthesis of key points of the developed model based on the indirect attention mechanism.

2. Related Work

The necessity for faster turnaround times to interpret radiographic images has led to a substantial effort to adopt CNN-based techniques, with a concentrated effort on distinguishing COVID-19 infected patients with the aid of both CT [15–21] and X-ray [14,22–34] imaging. Several overviews into the application of CNN techniques to aid in COVID-19 diagnosis have been conducted and we refer the reader to Refs. [35–37] for more details.

The authors in Ref. [34] propose DeepCOVID-XR, an ensemble of CNNs, to detect the presence of COVID-19 on frontal chest radiographs with an accuracy of 82% reported on a test set of 300 images (194 of which were from COVID-19 infected patients). Studying 5,090 images (1,979 of which were COVID-19 positive), the authors in Ref. [33] were able to achieve a binary classification accuracy of 99.5% by making use of HOG + CNN architecture for feature extraction and VGG for classification. In Ref. [32], pre-trained CNN models VGG-16, VGG-19, MobileNet, and Inception ResNet V2, are used to achieve a classification accuracy of at least 90.8% across 545 images (181 of which are COVID-19 positive).

Patients diagnosed with COVID-19 present symptoms consistent with pneumonia in their X-ray images, necessitating the ability to distinguish between COVID-19 and non-COVID-19 based pneumonia findings. Mahmud et al. [29] introduced CovXNet, a CNN-based model that makes use of depthwise convolution with varying dilation rates. The model is trained in two stages, first, on images corresponding to normal and viral/bacterial pneumonia. The model is then trained to distinguish COVID-19 from other forms of pneumonia, with a multi-class accuracy of 90.2% when trained (second stage) on 305 images in each class. Abbas et al. [31] developed a deep CNN called DeTrac to achieve an accuracy of 93.1% when detecting COVID-19 in 196 images across three categories: normal, severe acute respiratory syndrome (SARS), and COVID-19. Toraman et al. [38] developed Convolutional CapsNet (capsule neural network) to distinguish COVID-19 from normal and pneumonia X-ray images. The authors reported an accuracy of 97.2% and 84.2% for binary and multi-class classification, respectively, when making use of 2,331 images (231 of which were COVID-19 positive). Mansour et al. [39] introduced an unsupervised deep-learning-based variational autoencoder model for COVID-19 prediction, with resultant accuracies of 98.7% and 99.2% for binary and multi-class classification respectively. The authors tested their model against the X-ray dataset found in Ref. [40], split across normal, COVID-19, SARS, and ARDS classes. Khan et al. [41] developed CoroNet, a CNN model based on the Xception architecture. When tasked with classifying X-ray images as either normal, COVID-19, bacterial pneumonia, or viral pneumonia, the model achieved an accuracy of 89.6%, based on a dataset consisting of 1,251 images (284 of which belonged to COVID-19 positive cases). Chandra et al. [42] introduced an automatic COVID-19 screening system that uses a two-phase classification approach (normal vs abnormal and then COVID-19 vs pneumonia). The implemented classifier ensemble makes use of majority voting across five benchmark classification algorithms. By making use of 2,346 X-ray images (782 were COVID-19 positive), the authors report accuracies of 98.1% and 91.3% for each phase respectively. Ozturk et al. [30] developed DarkCovidNet, a model that obtained an accuracy of 87.0% when distinguishing between COVID-19, normal, and pneumonia in 1,127 images (127 of which are from COVID-19 positive patients). Wang et al. [14], developed a state-of-the-art model, called COVID-Net, that attains an accuracy of 93.3% when classifying a patient’s image across three categories: normal, pneumonia, and COVID-19.

Despite recent progress in the development of CNN-based algorithms, several fundamental challenges remain: the scarcity of publicly available data, overfitting of models, and model sizes that make their adoption within a healthcare setting cumbersome. We extend upon existing works by combining various publicly available data sources and carefully annotate the images across three classes: normal, pneumonia, and COVID-19. The data is then divided into training, validation, and testing subsets with an 8:1:1 split respectively, with a strict class balance maintained across all sets. Furthermore, we make use of widely adopted CNNs whose size is a fraction of some purpose-built models.

3. Data

We collected data from different publicly available sources to train a high-precision classifier and to estimate its generalization properties. At the time of publication, we identified the following five datasets; COVID Chest X-Ray Dataset (CCXD) [40,43], Actualmed COVID-19 Chest X-Ray Dataset (ACCD) [44], Figure 1 COVID-19 Chest X-Ray Dataset (FCCD) [45], COVID-19 Radiography Database (CRD) [46,47], and RSNA Pneumonia Detection Dataset (RSNA) [48]. Since the datasets include different labels for their findings, we reassigned the labels to maintain consistency across the global dataset. We assigned viral and bacterial cases of pneumonia to the “Pneumonia” label; SARS, MERS-CoV, COVID-19, and COVID-19 (ARDS) to the “COVID-19” label; “no findings” and “normal” diagnosis to the “Normal” label. Table 1 summarizes the statistical information of the study dataset.

It should be noted that the RSNA dataset includes only normal and pneumonia cases. Originally, this dataset consisted of 20,672 normal

Table 1
Statistical information of the dataset used in the study.

Dataset	Diagnosis			Total
	Normal	Pneumonia	COVID-19	
CCXD	18	165	504	687 (26%)
ACCD	127	–	58	185 (7%)
FCCD	3	2	35	40 (2%)
CRD	–	–	219	219 (8%)
RSNA	800	700	–	1500 (57%)
Total	948 (36%)	867 (33%)	816 (31%)	2631 (100%)

cases and 9,555 cases of pneumonia. In order to keep class balance in our dataset, we incorporated a total of 800 normal and 700 pneumonia cases. It is worth noting that normal and pneumonia cases from the CRD dataset were excluded because they duplicated images from the CCXD dataset.

The final dataset includes images acquired from the anterior-posterior (AP) and posteroanterior (PA) directions only. Lateral CXR has no clinical applicability to distinguish COVID-19 patients [49]. During network training, validation, and testing, the dataset was split in an 8:1:1 ratio i.e. the training subset includes 2,122 images (80%), the validation subset – 242 images (10%), and the testing subset – 267 images (10%). The split of data within training, validation, and testing phases was performed according to the distribution shown in Table 2.

4. Methods

The proposed workflow in this study is divided into three stages. First, we utilized the transfer learning approach based on 10 industry-standard networks including MobileNet V2, DenseNet-121, EfficientNet B0, EfficientNet B1, EfficientNet B3, EfficientNet B5, VGG-16, ResNet-50 V2, Inception V3, and Inception ResNet V2. The weights of feature extractors (networks bodies) were frozen and only the classifier heads were trained. During the second stage, we chose the 4 most accurate networks to advance to full training. Here, the weights of the whole network were unfrozen, such that both the feature extractor and the classifier were trained. Finally, the networks were trained with an indirect attention mechanism. Such an indirect supervision mechanism is based on the adoption of the Grad-CAM approach [50], where the output is used to focus the classifier on the lung area of an image. Indirect supervision is used in the training process since Grad-CAM's attention heatmaps reflect the areas of an input image supporting the network's prediction. In this regard, the prediction is based on the areas on which we expect the network to focus, while indirect supervision forces networks to focus on the desired object in the image rather than its surroundings. The training workflow of the model is shown in Fig. 1 below. All three stages are described in the paragraph Description of the workflow stages in more detail. It should also be noted that different COVID-Net models [14] are considered in this study. To date, COVID-Net models are state-of-the-art models used for distinguishing COVID-19 and pneumonia cases. All COVID-Net models are abbreviated to CXR in the remainder of the paper.

4.1. Description of the workflow stages

As mentioned previously, 10 deep learning networks were selected to determine which network architectures are most effective in recognizing COVID-19 and pneumonia. All networks vary in the number of weights, architecture topology, data processing, etc. Additionally, CXR models are used for comparison purposes. In order to compare the investigated networks, we provide an overview of the networks used during the first stage in Table 3.

Table 2
Description of the data distribution across training, validation, and testing subsets.

Dataset	Diagnosis	Training	Validation	Testing	Total
CCXD	Normal	14	2	2	18
	Pneumonia	133	15	17	165
	COVID-19	407	46	51	504
ACCD	Normal	102	12	13	127
	Pneumonia	0	0	0	0
	COVID-19	46	6	6	58
FCCD	Normal	1	1	1	3
	Pneumonia	0	1	1	2
	COVID-19	27	4	4	35
CRD	Normal	0	0	0	0
	Pneumonia	0	0	0	0
	COVID-19	177	20	22	219
RSNA	Normal	648	72	80	800
	Pneumonia	567	63	70	700
	COVID-19	0	0	0	0
Total		**2122 (80%)**	**242 (10%)**	**267 (10%)**	**2631 (100%)**

To train the aforementioned networks, we used bodies of these networks with frozen ImageNet weights. The most optimal version of each model was obtained through a series of training jobs performed on the collected dataset through the utilization of Amazon SageMaker. Having performed hyperparameter tuning based on a Bayesian optimization strategy, a set of hyperparameter values for the best performing model, given by the validation accuracy, was found. We chose the following pool of hyperparameters for the investigation:

- The number of blocks, where each block is constructed of densely connected, activation, and dropout layers, was chosen to vary from 1 to 5.
- The number of neurons for each densely-connected layer was varied from 64 to 512 with an increment of 8.
- Optimizers were chosen from a set of Adam, SGD, RMSprop, FTRL, and Rectified Adam.
- The learning rate was continuously varied on a log scale from 10^{-1} to 10^{-5}.
- Activation functions were chosen from a set of ReLU, ELU, Leaky ReLU, and SELU.
- The dropout rate was varied from 0.00 to 0.50 with a step of 0.05.

It is worth noticing that the architecture including 3 densely connected and 2 dropout layers was an optimal solution for all networks. However, the number of neurons varied slightly from network to network. The optimal number of neurons for the first and second densely-connected layers varied from 112 to 136 and from 56 to 72 respectively. A similar situation was observed for the dropout rate which varied from 0.05 to 0.15 for the first dropout layer, and from 0.05 to 0.10 for the second dropout layer. In this regard, we chose the optimal architecture of all network classifiers consisting of the following layers:

- Densely-connected layer with 128 neurons and ELU activation;
- Dropout layer with dropout rate equal to 0.10;
- Densely-connected layer with 64 neurons and ELU activation;
- Dropout layer with dropout rate equal to 0.05;
- Densely-connected layer with 3 neurons;
- Softmax activation layer.

It is important to note that for the first stage, only the classification heads were trained with the body weights frozen. According to the results of the hyperparameter tuning procedure, the gradient descent optimizer SGD with a learning rate equal to 10^{-4} proved to be optimal. Having trained several state-of-the-art networks, we found that most of them diverged. As a result, L2-regularization with λ of 0.001 was applied to all training networks. All networks were trained with a batch size equal to 32. To avoid overfitting during network training, we applied Early Stopping regularization, monitoring validation loss with a patience equal to 10 epochs. For training networks in both first and second stages, we used cross-entropy, calculated as follows:

$$L_{cls} = -\sum_{i=1}^{c} y_i * log(p_i + \varepsilon) \tag{1}$$

where c is the number of classes (3 in our study), y_i is the ground-truth label (ternary indicator), p_i is the softmax probability for the c-th class, ε is a small positive constant used for avoiding an undefined case of $log(0)$.

During the second stage, we took the four best performing networks with their trained heads from the first stage, namely MobileNet V2, EfficientNet B1, EfficientNet B3, VGG16, unfroze their body weights

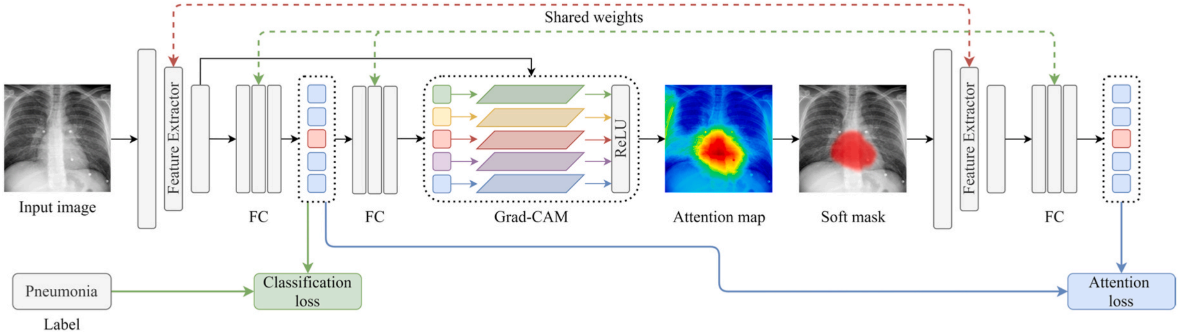


Fig. 1. The workflow for classification with the indirect supervision mechanism.

(weights of feature extractors) and retrained them using the SGD optimizer whose learning rate was 10^{-5}. As seen, we decreased the learning rate by a factor of 10 compared to that used in the first stage. It is important to lower the learning rate at this stage since a larger model with more unfrozen weights is trained, and this requires the readaptation of the pre-trained weights. Otherwise, unfreezing all weights without changes in the training policy may lead to quick model overfitting.

Once the performance and accuracy metrics of all networks were estimated, four networks that showed the best results during the first stage were chosen for fine-tuning. Besides training both bodies and heads of the networks, we introduced an indirect supervision mechanism for the considered networks. We were inspired by Ref. [59], where the authors proposed a framework that provides guidance on the attention maps generated by a weakly supervised deep learning neural network. The attention block in our pipeline is based on the usage of Grad-CAM preceded by a classification block. Usually, attention maps only cover small discriminative regions of the object of interest when the network is purely supervised by the classification loss. In order to overcome this issue and use attention maps as more reliable priors, both classification and attention blocks share weights between each other. The latter acts as a regularizer, imposing constraints on the attention maps. While the classification block is targeted to search for regions used in the recognition of classes, the attention block ensures that all regions that can contribute to the classification decision will be included in the network's attention. Such an iterative process aids both classification and attention blocks in finding reliable priors and making a correct decision.

With the usage of the indirect supervision mechanism, the network learns to extend the focus area of an input image contributing to the recognition of the target class as much as possible, such that the attention maps are tailored towards the task of interest. In this regard, during network training in Stage III, the loss differs from that of Stage I and Stage II and is calculated as follows:

$$L_{total} = \alpha L_{cls} + \beta L_{attn} \tag{2}$$

where L_{cls} is the classification loss i.e. cross-entropy loss defined in Eq. (1), L_{attn} is the attention loss, α and β are the coefficients used to scale the total loss and both components. In order to obtain the attention map and compute the attention loss L_{attn} for a given image I, we compute the neuron importance weights $w^c_{l,k} = GAP\left(\frac{\partial s^c}{\partial f_{l,k}}\right)$ using an application of the global average pooling operation (GAP) to the gradient of the score s^c with respect to activation maps $f_{l,k}$. Once $w^c_{l,k}$ are computed on the backward pass, the networks are not updated. Since $w^c_{l,k}$ represents the importance of the activation map $f_{l,k}$ (activation of unit k on the l-th layer) assisting in prediction of class c, the indirect mechanism uses the weights matrix w^c and applies a two-dimensional convolution over activation maps f_l, integrating all of them. Then the *ReLU* operation allows us to obtain the attention map A^c computed as follows:

$$A^c = ReLU(conv(f_l,\ w^c)) \tag{3}$$

where l is the representation from the last convolutional layer whose features have the best compromise between high-level semantics and detailed spatial information. The attention map A^c has the same size as the convolutional feature maps (see the column with the size of the output feature matrix in Table 3).

Using the trainable attention map A^c we generate a soft mask that is applied to an input image. This procedure allows us to obtain regions I^{*c} which are beyond the network's current attention for class c and are calculated as follows:

$$I^{*c} = I - (T(A^c) \odot I) \tag{4}$$

$$T(A^c) = \frac{1}{1 + e^{-\omega(A^c - M^\sigma)}} \tag{5}$$

where I is an input image, $T(A^c)$ is a masking function that is based on the thresholding operation, and $\odot$ denotes element-wise multiplication. Since standard thresholding is not derivable, $T(A^c)$ is approximated using a sigmoid function, where M^σ is the thresholding matrix filled with σ values, ω is a scale parameter, ensuring $T(A^c)_{i,j}$ *is equal to* 1, when $A^c_{i,j}$ is larger than σ or equal to 0 otherwise.

Having obtained a soft mask I^{*c}, the attention block of the pipeline uses it to compute the prediction scores s^c for all classes. Since the indirect supervision mechanism is used to guide the network to focus its attention on all parts of a given class, I^{*c} has to contain as little features belonging to the target class as possible because regions beyond the high-responding area on the attention map area should not include single-pixel areas that can trigger the network to recognize the object of class c. From the perspective of the attention loss function, it is designed to minimize the prediction score s^c of I^{*c} and is calculated as follows:

$$L_{attn} = \frac{1}{n}\sum_c s^c(I^{*c}) \tag{6}$$

where n is the number of ground-truth class labels for an input image I.

4.2. Visual model validation

While modern neural networks enable superior performance, their lack of decomposability into intuitive and understandable components makes them hard to interpret. In this regard, an achievement of the model transparency is useful to explain their predictions. Class Activation Map (CAM) is a modern-day technique used for model interpretation [60]. Though CAM is a good technique to demystify the working of CNNs, it suffers from several drawbacks. For example, CAM requires feature maps to directly precede the softmax layers, so it applies to a particular kind of network architecture that performs global average pooling over convolutional maps immediately before prediction. Such

Table 3
Description of the models used during the first stage.

Model	Size of an input image	Size of an output feature matrix	Parameters, millions	Depth	Size, Mb	Source
MobileNet V2	224 × 224 × 3	7 × 7 × 1280	3.5	88	14	[51]
DenseNet-121	224 × 224 × 3	7 × 7 × 1024	8.0	121	33	[52]
EfficientNet B0	224 × 224 × 3	7 × 7 × 1280	5.3	–	29	[53]
EfficientNet B1	240 × 240 × 3	8 × 8 × 1280	7.9	–	31	[53]
EfficientNet B3	300 × 300 × 3	10 × 10 × 1536	12.3	–	48	[53]
EfficientNet B5	456 × 456 × 3	15 × 15 × 2048	30.6	–	75	[53]
VGG-16	224 × 224 × 3	7 × 7 × 512	138.4	23	528	[54]
ResNet-50 V2	224 × 224 × 3	7 × 7 × 2048	25.6	50	98	[55]
InceptionV3	299 × 299 × 3	8 × 8 × 2048	23.9	159	92	[56]
Inception ResNet V2	299 × 299 × 3	5 × 5 × 1536	55.9	572	215	[57]
CXR Small	224 × 224 × 3	7 × 7 × 2048	117.4	–	1448	[58]
CXR Large	224 × 224 × 3	7 × 7 × 2048	127.4	–	1486	[58]
CXR-3A	480 × 480 × 3	13 × 13 × 1536	40.2	–	617	[58]
CXR-3B	480 × 480 × 3	15 × 15 × 2048	11.7	–	293	[58]
CXR-3C	480 × 480 × 3	15 × 15 × 2048	9.2	–	210	[58]
CXR-4A	480 × 480 × 3	13 × 13 × 1536	40.2	–	617	[58]
CXR-4B	480 × 480 × 3	15 × 15 × 2048	11.7	–	293	[58]
CXR-4C	480 × 480 × 3	15 × 15 × 2048	9.2	–	210	[58]

architectures may achieve inferior accuracies compared to general networks on some tasks or simply be inapplicable to new tasks. De facto deeper representations of a CNN capture the best high-level features. Furthermore, CNNs naturally retrain spatial information which is lost in fully connected layers, so we expect the last convolutional layer to have the best tradeoff between high-level semantics and detailed spatial information. In this regard, a popular technique, known as Grad-CAM and published in Ref. [50], aims to improve the shortcomings of CAM and claims to be compatible with any kind of architecture. The technique does not require any modifications to the existing model architecture, and this allows its application to any CNN-based architecture. Unlike CAM, Grad-CAM uses the gradient information flowing into the last convolutional layer of a CNN to understand each neuron for a decision of interest. Grad-CAM improves on its predecessor, provides better localization and clear class discriminative saliency maps. As such, we created heatmap images using the following equations:

$$\alpha_k^c = \frac{1}{Z}\sum_i \sum_j \frac{\partial y^c}{\partial A_{ij}^k} \tag{7}$$

$$L_{Grad-CAM}^c = ReLU\left(\sum_k \alpha_k^c A^k\right) \tag{8}$$

where the algorithm takes gradient of the output y^c with respect to a feature map A^k , then it averages the result to get a weight of each feature map α_k^c. Finally, Grad-CAM takes a linear combination of weights α_k^c and feature maps A^k, which gives us heatmaps.

5. Results

5.1. *Stage I*

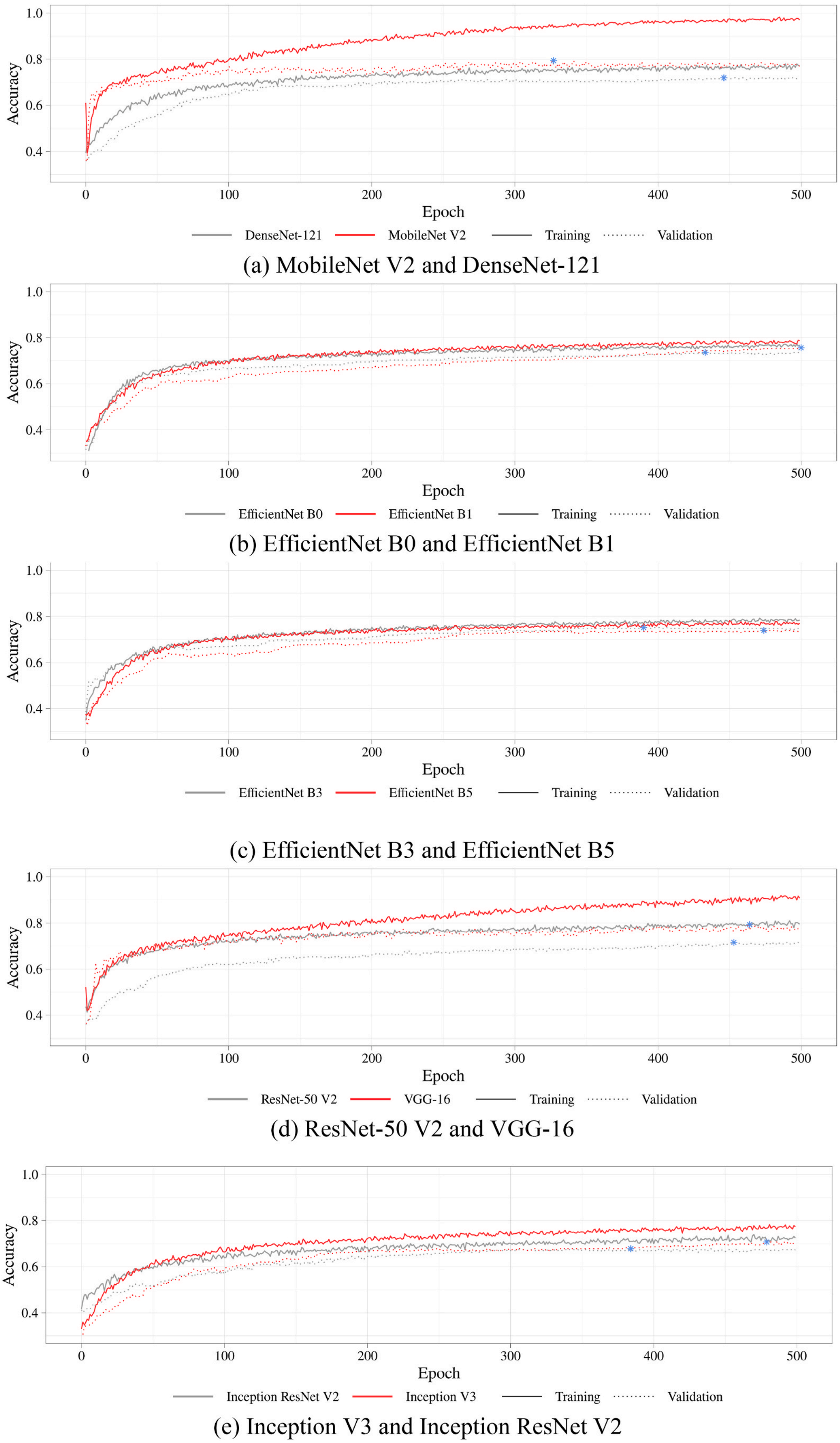
Having trained 10 neural networks, we found that two networks tend to overfit more than others. This is likely connected with their normalization layers. Networks such as MobileNet V2 and VGG-16 do not have Batch/Instance/Layer/Group Normalization layers in their architecture. In this regard, these networks start overfitting (MobileNet V2) or hit a validation loss/accuracy plateau (VGG-16) after approximately 100 epochs, while the training accuracy keeps increasing. Popular regularization techniques such as Lasso Regression (L1 Regularization), Ridge Regression (L2 regularization), ElasticNet (L1-L2 regularization), Dropout, and Early Stopping may help to avoid this problem. In this regard, we applied Ridge Regression, Dropout layers, and Early Stopping in our training pipeline. As for the remaining networks, they did not suffer from overfitting; however, they could not reach better validation loss/accuracy values. When a given model reached its best validation loss, we saved the associated model weights using a saving callback. Fig. 2 demonstrates how the accuracy dynamics of the networks evolved during the first training stage. Blue asterisks reflect the best value of the accuracy on the validation subsets.

Since loss is poorly interpreted, we compared commonly used network metrics such as accuracy and F1-score. Table 4 and Table 5 summarize these metrics estimated during the first stage. As seen, MobileNet V2, EfficientNet B1, EfficientNet B3, and VGG-16 achieved better results than other networks. Additionally, we provide all obtained metrics (Accuracy, F1-score, Precision, and Recall), computed over different subsets, classes, and stages inAppendix A.

5.2. *Stage II*

Based on the results of the first stage, MobileNet V2, EfficientNet B1, EfficientNet B3, and VGG-16 demonstrated their ability to distinct COVID-19 and pneumonia on X-ray images much better than other networks. During the second stage, we chose these four most accurate networks to advance to full training. The weights of each network were unfrozen, such that both the feature extractor and the classifier were trained. Having obtained the accuracy dynamics, we compare, in Fig. 3, how fully-trained networks differ from the networks fine-tuned in the first stage. The blue asterisks in this figure reflect the best value of the accuracy reached on the validation subset.

Having compared the accuracy and F1-score values obtained in the first (Tables 4 and 5) and second stages (Table 6 and Table 7), we can state that MobileNet V2 and VGG-16 have a larger boost in accuracy over EfficientNet models. Once full training was performed, MobileNet V2 and VGG-16 got a +6% and +9% accuracy change on the validation subset and a +1% and +4% accuracy change on the testing subset. On the other hand, EfficientNet B1 and EfficientNet B3 displayed a +2% and +3% accuracy change on the validation subset and a −1% and +6%



(a) MobileNet V2 and DenseNet-121

(b) EfficientNet B0 and EfficientNet B1

(c) EfficientNet B3 and EfficientNet B5

(d) ResNet-50 V2 and VGG-16

(e) Inception V3 and Inception ResNet V2

Fig. 2. Accuracy dynamics over the training during the first stage.

Table 4
Performance metrics within different subsets obtained after the first stage.

Model	Accuracy			F1-score		
	Training	Validation	Testing	Training	Validation	Testing
MobileNet V2	0.95	0.79	0.77	0.95	0.80	0.78
DenseNet-121	0.76	0.72	0.74	0.76	0.72	0.75
EfficientNet B0	0.95	0.79	0.70	0.95	0.80	0.70
EfficientNet B1	0.79	0.76	0.74	0.79	0.76	0.75
EfficientNet B3	0.77	0.75	0.71	0.78	0.75	0.72
EfficientNet B5	0.77	0.74	0.70	0.77	0.74	0.70
VGG-16	0.90	0.79	0.78	0.90	0.80	0.79
ResNet-50 V2	0.80	0.71	0.69	0.80	0.71	0.70
Inception V3	0.77	0.71	0.73	0.77	0.71	0.74
Inception ResNet V2	0.71	0.68	0.70	0.71	0.67	0.70

Table 5
Performance metrics within different classes obtained after the first stage.

Model	Accuracy			F1-score		
	Normal	Pneumonia	COVID-19	Normal	Pneumonia	COVID-19
MobileNet V2	0.70	0.78	0.83	0.74	0.75	0.83
DenseNet-121	0.75	0.82	0.63	0.76	0.73	0.73
EfficientNet B0	0.74	0.69	0.66	0.71	0.66	0.72
EfficientNet B1	0.73	0.73	0.75	0.74	0.69	0.79
EfficientNet B3	0.70	0.72	0.72	0.70	0.69	0.75
EfficientNet B5	0.66	0.75	0.67	0.68	0.68	0.73
VGG-16	0.80	0.76	0.78	0.77	0.75	0.82
ResNet-50 V2	0.68	0.70	0.68	0.69	0.65	0.74
Inception V3	0.74	0.77	0.68	0.75	0.71	0.75
Inception ResNet V2	0.70	0.76	0.61	0.70	0.68	0.70

accuracy change on the testing subset. It should also be noted, that the largest boost in the classification of COVID-19 was achieved by VGG-16. This network had an +11% boost, while MobileNet V2, EfficientNet B1, and EfficientNet B3 could reach the level of +2%, 0%, and +6%, respectively.

5.3. Stage III

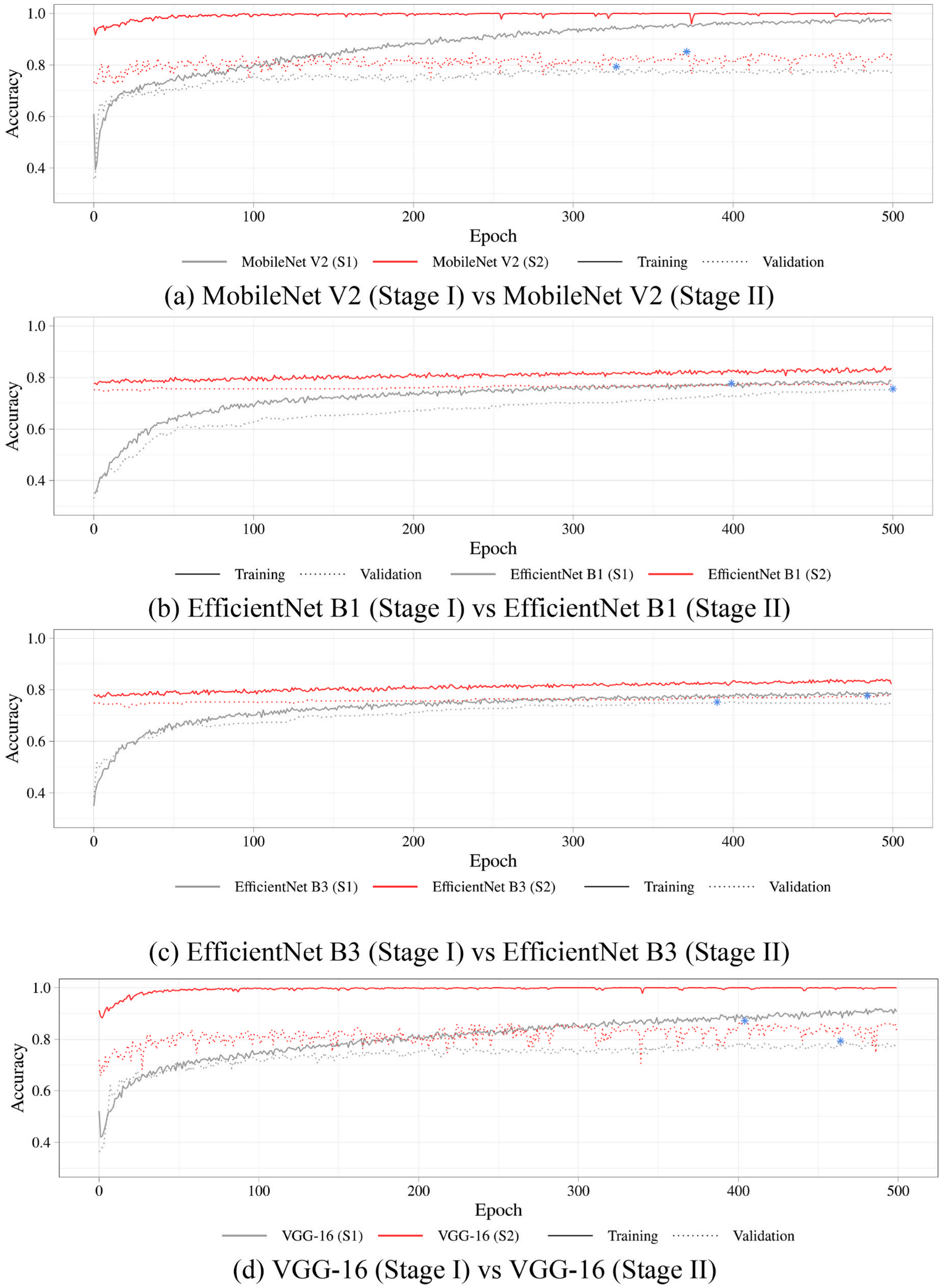
Once the networks are fine-tuned and fully trained, we then train those best four networks using the proposed pipeline based on indirect supervision. Having trained the chosen networks according to our pipeline described in Description of the workflow stages, we compared them on the validation and testing subsets, reflected in Fig. 4 and Appendix B. Based on the obtained results, we established that the proposed pipeline allows for boosting of the model accuracy. VGG-16 and MobileNet V2 showed the best accuracy on the validation and testing subsets. It is worth noticing that the VGG-16 network outperformed the best CXR model (CXR-4A) on these subsets. The performance of other CXR models is additionally shown inAppendix A. It is observed that the VGG-16 (S3) network trained based on the proposed pipeline has a +9% and +1% of accuracy boost on the validation subset compared to VGG-16 (S1) and VGG-16 (S2) respectively. Similar positive dynamics of using our pipeline are observed for other models as well. It should be noted that the CXR-4A and lightweight MobileNet V2 have almost the same accuracy, while the complexity of the latter is 11-time lower. The MobileNet V2 network includes 3.5 mln. weights, while CXR-4A includes 40.2 mln. weights.

In general, the network that produces the best results is VGG16, having consistently high values in every metric. We assume that VGG-16 could achieve such a high accuracy because of the high complexity and a large number of parameters (138.4 mln.) as compared to other studied networks. Additionally, we found that the plain network architecture is more suitable for the classification of indistinctive lung areas such as COVID-19 and pneumonia-affected regions. Both VGG-16 and MobileNet V2 are based on straight-line architecture, including, at most, a few skip-connections. Whilst the EfficientNet, ResNet, Inception, and Inception ResNet network families are based on complex architectures, including a wide variety of skip-connections such as identity/projection shortcuts (ResNet and Inception ResNet) and inception modules (Inception and Inception ResNet). It is worth noting that networks such as Inception V3 and Inception ResNet V2 integrate multiple kernels of different sizes (1 × 1, 3 × 3, and 5 × 5) which should assist in detecting area-specific features. However, 3 × 3 convolutional kernels, integrated to VGG-16 and MobileNet V2, turned out to provide a better solution, allowing for the network's better generalization ability and its ability to distinguish healthy patients from those diagnosed with COVID-19 or pneumonia.

5.4. Model validation using Grad-CAM

As we mentioned in Section "Visual model validation", despite deep learning models having facilitated unprecedented accuracy in image classification, one of their biggest drawbacks is model interpretability, representing a core component in understanding and debugging a model. We used the Grad-CAM technique to validate the models and their ability for making predictions and to verify which series of neurons activated in the forward-pass during the prediction. For the sake of visualization, we choose several patients with different findings: pneumonia, and COVID-19. Source images of these findings with their ground truth heatmaps and the heatmap dynamics over three stages are shown in Fig. 5 and Fig. 6.

Using Grad-CAM, we validated where our four best networks (MobileNet V2, EfficientNet B1, EfficientNet B3, VGG-16) are focusing, verifying that they are properly looking at the correct patterns in the image and activating around those patterns. The Grad-CAM technique uses the gradients flowing into the final convolutional layer to produce a coarse localization heatmap, highlighting the important regions in the image for predicting the target concept i.e. COVID-19 or pneumonia areas. However, the localization heatmaps may differ from the traditional localization techniques such as segmentation masks or bounding boxes. In this regard, these heatmaps are used for the sake of approximate localization.



(a) MobileNet V2 (Stage I) vs MobileNet V2 (Stage II)

(b) EfficientNet B1 (Stage I) vs EfficientNet B1 (Stage II)

(c) EfficientNet B3 (Stage I) vs EfficientNet B3 (Stage II)

(d) VGG-16 (Stage I) vs VGG-16 (Stage II)

Fig. 3. Accuracy dynamics over the training during the second stage.

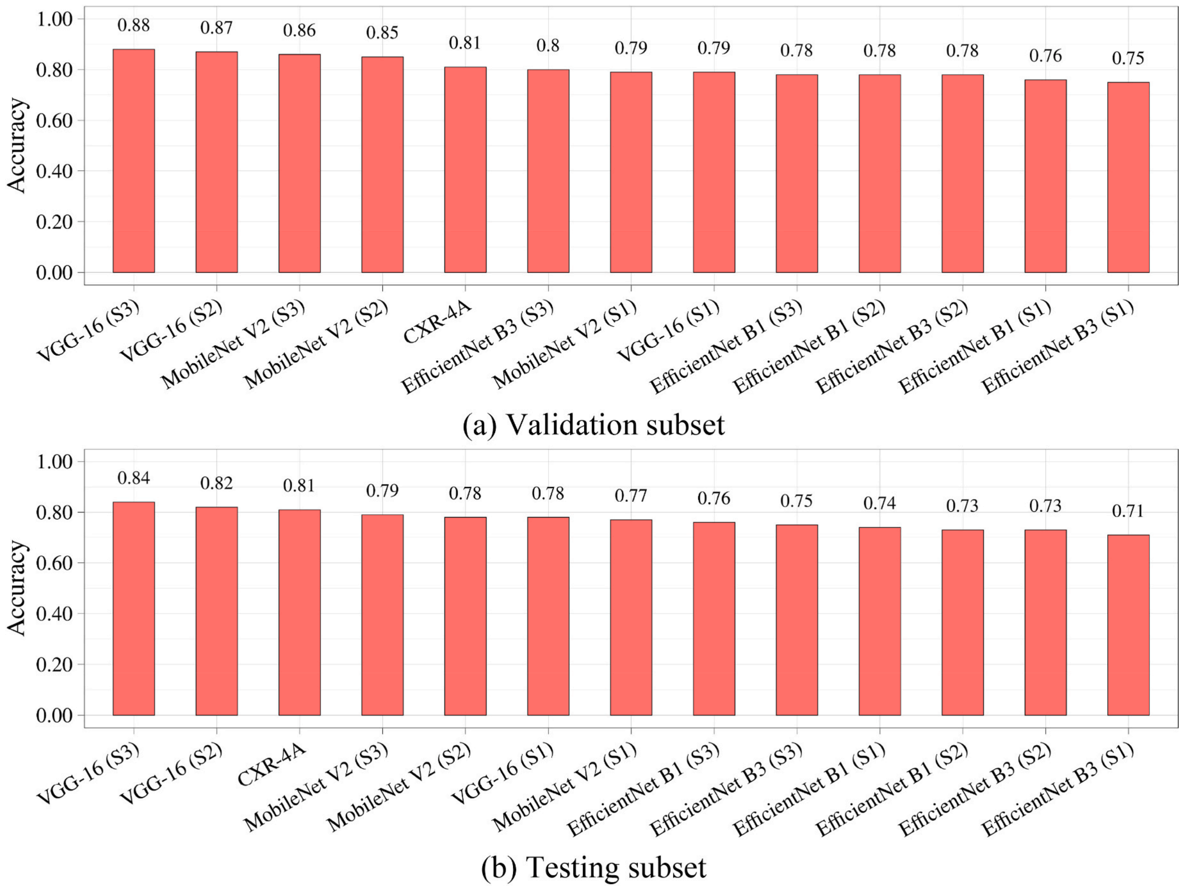
In order to interpret the models, Figs. 5 and 6 reflect the visualization of gradient class activation maps. Additional cases of the networks' heatmaps are shown in Appendix C and Appendix D. Due to the nature of the task at hand, we utilize Grad-CAM for training and visualization purposes only. As we do not segment the COVID-19 affected regions, we have insufficient image information to compute associated metrics such as the Dice coefficient or the Jaccard distance. However, based on the obtained results, we may state that the training of the models using soft masks obtained by the indirect supervision mechanism (Stage III) has a positive effect on the search for correct patterns by the models.

Table 6
Performance metrics within different subsets obtained after the second stage.

Model	Accuracy			F1-score		
	Training	Validation	Testing	Training	Validation	Testing
MobileNet V2	1.00	0.85	0.78	1.00	0.85	0.79
EfficientNet B1	0.83	0.78	0.73	0.83	0.78	0.74
EfficientNet B3	0.83	0.78	0.77	0.83	0.78	0.77
VGG-16	1.00	0.87	0.82	1.00	0.87	0.83

Table 7
Performance metrics within different classes obtained after the second stage.

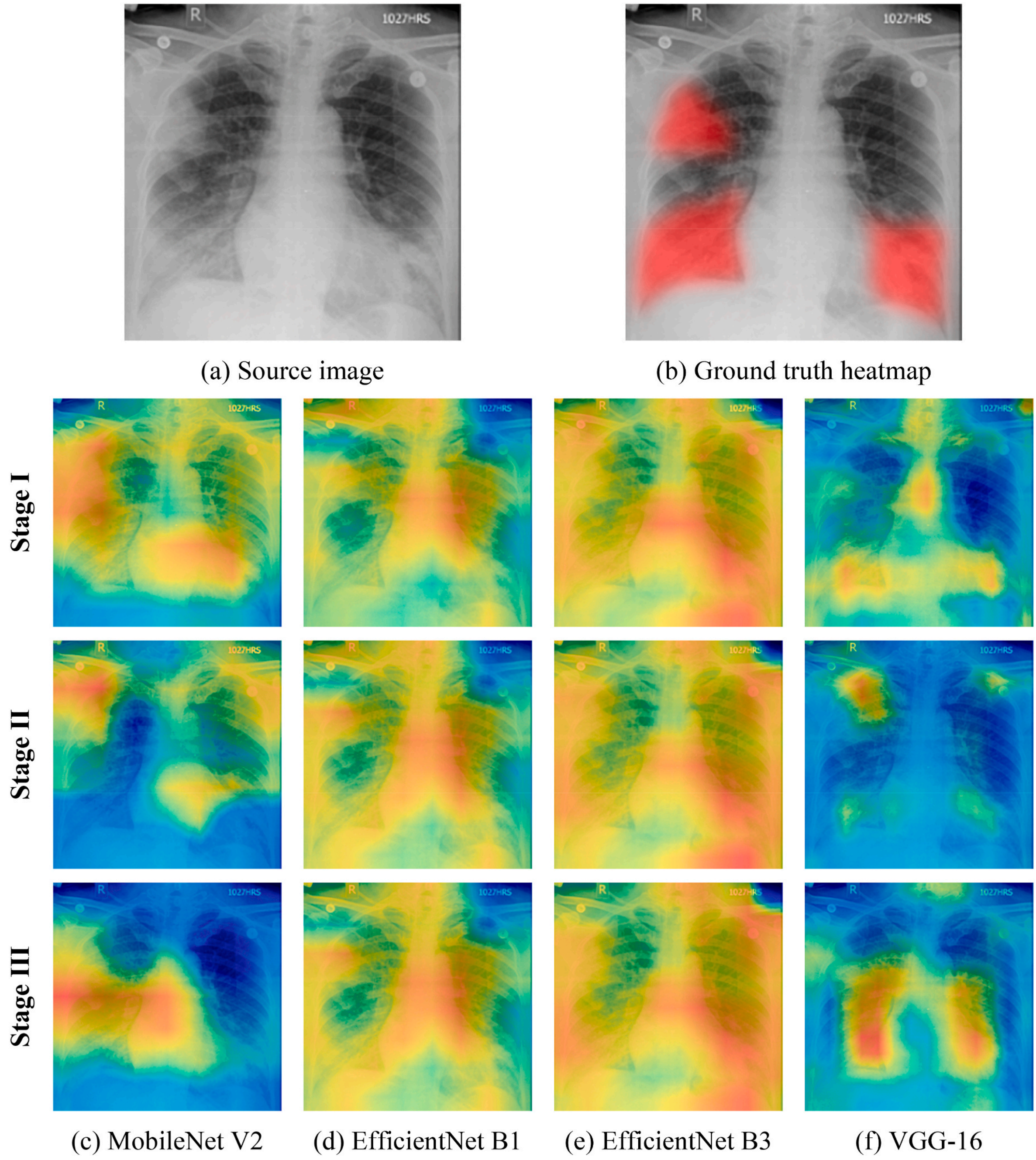
Model	Accuracy			F1-score		
	Normal	Pneumonia	COVID-19	Normal	Pneumonia	COVID-19
MobileNet V2	0.74	0.75	0.85	0.75	0.74	0.85
EfficientNet B1	0.70	0.74	0.75	0.72	0.71	0.78
EfficientNet B3	0.77	0.75	0.78	0.76	0.74	0.81
VGG-16	0.81	0.78	0.89	0.80	0.79	0.89



(a) Validation subset

(b) Testing subset

Fig. 4. Comparison of the networks' accuracy over different subsets and stages.



(a) Source image

(b) Ground truth heatmap

(c) MobileNet V2

(d) EfficientNet B1

(e) EfficientNet B3

(f) VGG-16

Fig. 5. Visualization of network heatmaps for a COVID-19 finding.

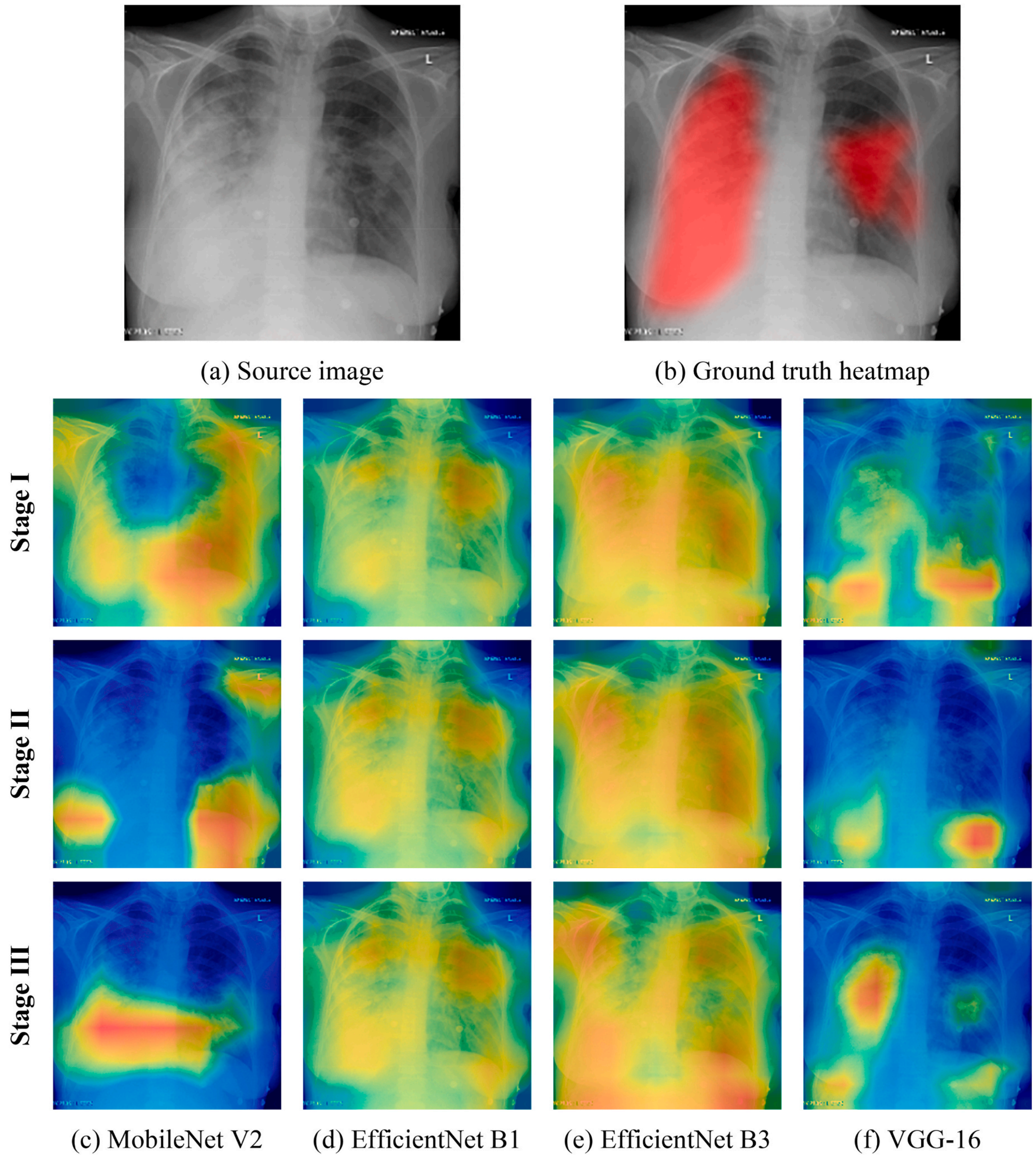
Networks such as MobileNet V2 (Figs. 5c and 6) and VGG-16 (Fig. 5f and 6f) identify affected areas correctly, despite the inaccuracies in the location of the heatmaps. On the other hand, interpretation of the EfficientNet networks showed that they are not activating around the proper patterns of the image. This allows us to assume that EfficientNet B1 and EfficientNet B3 have not properly learned the underlying patterns in our dataset and/or we may need to collect additional data for more complex training.

6. Conclusion

In this study, we demonstrated a training pipeline based on indirect supervision for neural networks. This supervision forces the neural networks to pay attention to the areas obtained by the external algorithm. Having trained a set of deep learning models, we found that the proposed pipeline allows for an increased classification accuracy. This pipeline was used for the detection of COVID-19 and distinguishing its presence from that of pneumonia. Of the obtained results, MobileNet V2 performed comparably to the tailor-made CXR model CXR-4A, despite being 11 times less complex. According to the performed experiments, the networks trained based on the proposed pipeline perform comparably to practicing radiologists when it comes to the classification of multiple thoracic pathologies in chest X-ray radiographs. Our pipeline may have the potential to improve healthcare delivery and increase access to chest radiograph expertise for the detection of a variety of acute diseases.

Author contributions

Y.G., S.S., and O.T. conceived the idea of the study. V.D., Y.G., and O.


(a) Source image

(b) Ground truth heatmap

(c) MobileNet V2

(d) EfficientNet B1

(e) EfficientNet B3

(f) VGG-16

Fig. 6. Visualization of network heatmaps for a pneumonia finding.

T. developed the plan of execution. V.D. and O.T. collected and annotated the data. V.D. and D.N. developed, trained, and analyzed the performance of deep learning networks on the collected data. A.P. tested the performance of purpose-built deep learning networks (COVID-Nets) on the collected data. V.D. and A.P. wrote the manuscript with input from all co-authors. A.K., A.K., V.K., V.S., and D.L. assisted in study direction and data quality discussions. Y.G. supervised the project.

Funding

Viacheslav V. Danilov was supported in part by the Ministry of Science and Higher Education of the Russian Federation, fundamental research project No. 0419-2021-001 “Novel anti-atherosclerotic therapies and machine learning solutions for automated diagnosis and prognostication of cardiovascular disease”.

Declaration of competing interest

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

Appendix A. Model metrics

Table 1
Model metrics computed over different subsets

STAGE I												
Model	Accuracy			F1-score			Precision			Recall		
	Train	Val	Test	Train	Val	Test	Train	Val	Test	Train	Val	Test
MobileNet V2	0.95	0.79	0.77	0.95	0.80	0.78	0.95	0.80	0.78	0.95	0.79	0.78
DenseNet-121	0.76	0.72	0.74	0.76	0.72	0.75	0.76	0.74	0.77	0.76	0.72	0.74
EfficientNet B0	0.95	0.79	0.70	0.95	0.80	0.70	0.95	0.80	0.71	0.95	0.79	0.70
EfficientNet B1	0.79	0.76	0.74	0.79	0.76	0.75	0.79	0.76	0.75	0.79	0.76	0.74
EfficientNet B3	0.77	0.75	0.71	0.78	0.75	0.72	0.78	0.75	0.73	0.78	0.75	0.72
EfficientNet B5	0.77	0.74	0.70	0.77	0.74	0.70	0.77	0.74	0.71	0.77	0.74	0.70
VGG-16	0.90	0.79	0.78	0.90	0.80	0.79	0.90	0.80	0.79	0.90	0.79	0.78
ResNet-50 V2	0.80	0.71	0.69	0.80	0.71	0.70	0.80	0.73	0.71	0.80	0.71	0.69
Inception V3	0.77	0.71	0.73	0.77	0.71	0.74	0.77	0.72	0.75	0.77	0.70	0.74
Inception ResNet V2	0.71	0.68	0.70	0.71	0.67	0.70	0.71	0.71	0.72	0.71	0.67	0.70
STAGE II												
Model	**Accuracy**			**F1-score**			**Precision**			**Recall**		
	Train	**Val**	**Test**	**Train**	**Val**	**Test**	**Train**	**Val**	**Test**	**Train**	**Val**	**Test**
MobileNet V2	1.00	0.85	0.78	1.00	0.85	0.79	1.00	0.85	0.79	1.00	0.85	0.78
EfficientNet B1	0.83	0.78	0.73	0.83	0.78	0.74	0.83	0.78	0.75	0.83	0.78	0.74
EfficientNet B3	0.83	0.78	0.77	0.83	0.78	0.77	0.83	0.78	0.78	0.83	0.78	0.77
VGG16	1.00	0.87	0.82	1.00	0.87	0.83	1.00	0.87	0.83	1.00	0.87	0.83
STAGE III												
Model	**Accuracy**			**F1-score**			**Precision**			**Recall**		
	Train	**Val**	**Test**	**Train**	**Val**	**Test**	**Train**	**Val**	**Test**	**Train**	**Val**	**Test**
MobileNet V2	1.00	0.86	0.79	1.00	0.86	0.78	1.00	0.86	0.78	1.00	0.86	0.78
EfficientNet B1	0.84	0.78	0.76	0.85	0.78	0.76	0.85	0.79	0.76	0.85	0.78	0.76
EfficientNet B3	0.89	0.80	0.75	0.89	0.80	0.76	0.89	0.80	0.76	0.89	0.80	0.75
VGG16	1.00	0.88	0.84	1.00	0.87	0.83	1.00	0.87	0.82	1.00	0.87	0.83
Covid-Net												
Model	**Accuracy**			**F1-score**			**Precision**			**Recall**		
	Train	**Val**	**Test**	**Train**	**Val**	**Test**	**Train**	**Val**	**Test**	**Train**	**Val**	**Test**
CXR Small	0.77	0.80	0.79	0.77	0.80	0.79	0.77	0.80	0.79	0.78	0.81	0.79
CXR Large	0.79	0.78	0.79	0.79	0.78	0.79	0.79	0.78	0.79	0.80	0.78	0.80
CXR-3A	0.79	0.79	0.78	0.79	0.79	0.78	0.79	0.79	0.78	0.79	0.79	0.79
CXR-3B	0.79	0.78	0.78	0.79	0.78	0.78	0.79	0.78	0.78	0.79	0.78	0.79
CXR-3C	0.78	0.79	0.79	0.78	0.79	0.79	0.78	0.79	0.79	0.79	0.79	0.80
CXR-4A	0.81	0.81	0.81	0.81	0.81	0.81	0.81	0.81	0.81	0.81	0.82	0.82
CXR-4B	0.79	0.79	0.78	0.79	0.79	0.78	0.79	0.79	0.78	0.79	0.79	0.79
CXR-4C	0.79	0.80	0.80	0.79	0.80	0.80	0.79	0.80	0.80	0.79	0.80	0.81

Abbreviations: Train – training subset, Val – validation subset, Test – testing subset.

Table 2
Model metrics computed over different classes

STAGE I												
Model	Accuracy			F1-score			Precision			Recall		
	Norm	PNA	COV	Norm	PNA	COV	Norm	PNA	COV	Norm	PNA	COV
MobileNet V2	0.70	0.78	0.83	0.74	0.75	0.83	0.78	0.71	0.84	0.70	0.78	0.83
DenseNet-121	0.75	0.82	0.63	0.76	0.73	0.73	0.78	0.66	0.85	0.75	0.82	0.63
EfficientNet B0	0.74	0.69	0.66	0.71	0.66	0.72	0.69	0.64	0.79	0.74	0.69	0.66
EfficientNet B1	0.73	0.73	0.75	0.74	0.69	0.79	0.75	0.66	0.84	0.73	0.73	0.75
EfficientNet B3	0.70	0.72	0.72	0.70	0.69	0.75	0.70	0.66	0.80	0.70	0.72	0.72
EfficientNet B5	0.66	0.75	0.67	0.68	0.68	0.73	0.71	0.63	0.80	0.66	0.75	0.67
VGG-16	0.80	0.76	0.78	0.77	0.75	0.82	0.75	0.74	0.87	0.80	0.76	0.78
ResNet-50 V2	0.68	0.70	0.68	0.69	0.65	0.74	0.69	0.61	0.81	0.68	0.70	0.68
Inception V3	0.74	0.77	0.68	0.75	0.71	0.75	0.76	0.66	0.82	0.74	0.77	0.68
Inception ResNet V2	0.70	0.76	0.61	0.70	0.68	0.70	0.70	0.62	0.83	0.70	0.76	0.61
STAGE II												
Model	**Accuracy**			**F1-score**			**Precision**			**Recall**		
	Norm	**PNA**	**COV**	**Norm**	**PNA**	**COV**	**Norm**	**PNA**	**COV**	**Norm**	**PNA**	**COV**
MobileNet V2	0.74	0.75	0.85	0.75	0.74	0.85	0.77	0.74	0.84	0.74	0.75	0.85
EfficientNet B1	0.70	0.74	0.75	0.72	0.71	0.78	0.73	0.68	0.81	0.70	0.74	0.75
EfficientNet B3	0.77	0.75	0.78	0.76	0.74	0.81	0.75	0.73	0.84	0.77	0.75	0.78
VGG16	0.81	0.78	0.89	0.80	0.79	0.89	0.78	0.81	0.89	0.81	0.78	0.89
STAGE III												
Model	**Accuracy**			**F1-score**			**Precision**			**Recall**		
	Norm	**PNA**	**COV**	**Norm**	**PNA**	**COV**	**Norm**	**PNA**	**COV**	**Norm**	**PNA**	**COV**
MobileNet V2	0.74	0.76	0.84	0.75	0.76	0.83	0.76	0.76	0.82	0.74	0.76	0.84
EfficientNet B1	0.73	0.76	0.78	0.75	0.73	0.80	0.76	0.70	0.82	0.73	0.76	0.78
EfficientNet B3	0.72	0.78	0.76	0.74	0.73	0.80	0.77	0.68	0.84	0.72	0.78	0.76
VGG16	0.86	0.78	0.88	0.83	0.80	0.86	0.81	0.81	0.88	0.86	0.78	0.88
Covid-Net												
Model	**Accuracy**			**F1-score**			**Precision**			**Recall**		
	Norm	**PNA**	**COV**	**Norm**	**PNA**	**COV**	**Norm**	**PNA**	**COV**	**Norm**	**PNA**	**COV**
CXR Small	0.86	0.83	0.88	0.81	0.75	0.80	0.71	0.83	0.85	0.93	0.68	0.76
CXR Large	0.87	0.82	0.90	0.82	0.74	0.83	0.73	0.80	0.89	0.94	0.68	0.77
CXR-3A	0.85	0.85	0.87	0.77	0.78	0.80	0.74	0.84	0.77	0.80	0.73	0.82
CXR-3B	0.85	0.83	0.88	0.79	0.74	0.81	0.73	0.83	0.81	0.87	0.67	0.82
CXR-3C	0.86	0.84	0.89	0.80	0.75	0.83	0.73	0.86	0.81	0.88	0.67	0.84
CXR-4A	0.85	0.88	0.90	0.79	0.81	0.84	0.73	0.92	0.81	0.86	0.72	0.87
CXR-4B	0.85	0.82	0.90	0.79	0.73	0.84	0.71	0.85	0.82	0.88	0.63	0.86
CXR-4C	0.86	0.85	0.90	0.80	0.76	0.84	0.73	0.89	0.81	0.88	0.66	0.88

Abbreviations: Norm – normal (no findings), PNA – pneumonia, COV – COVID-19.

Appendix B. Overall network comparison at different stages

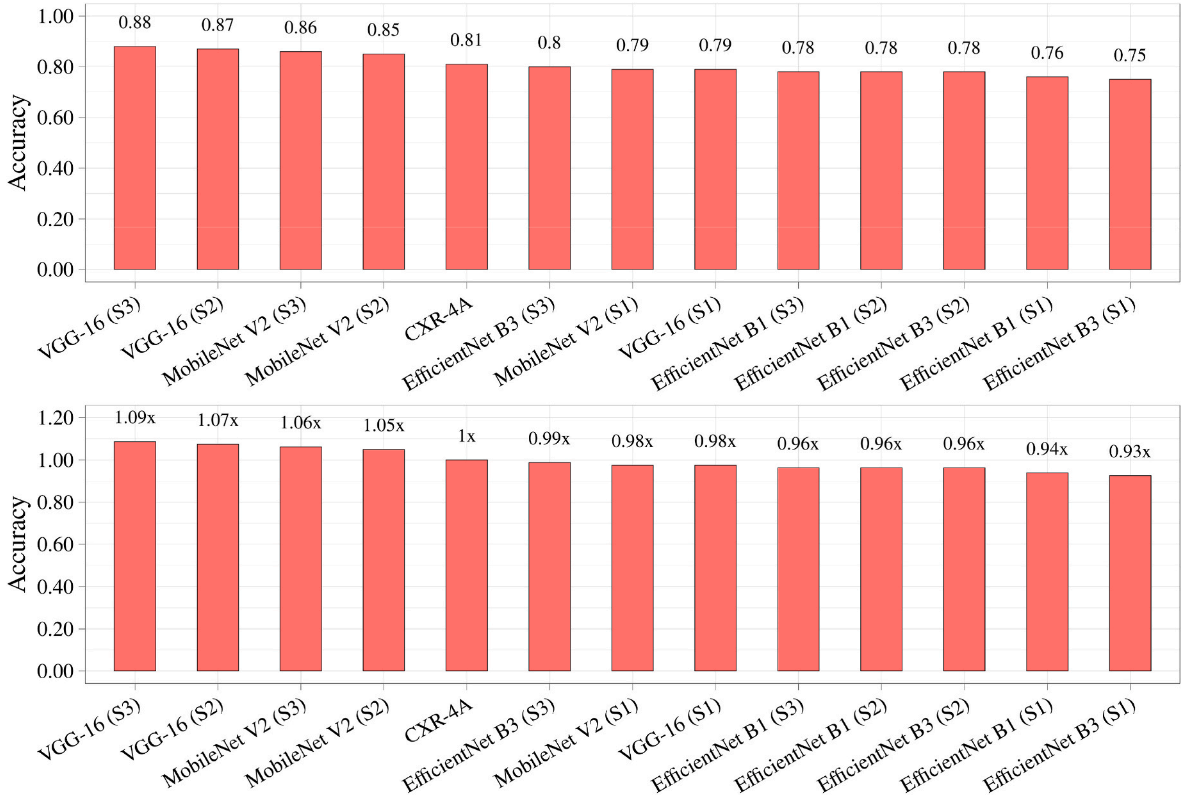


Fig. 1. Comparison of networks' accuracy based on the validation subset. The top chart is the comparison of absolute values, while the bottom chart is the comparison of relative values.

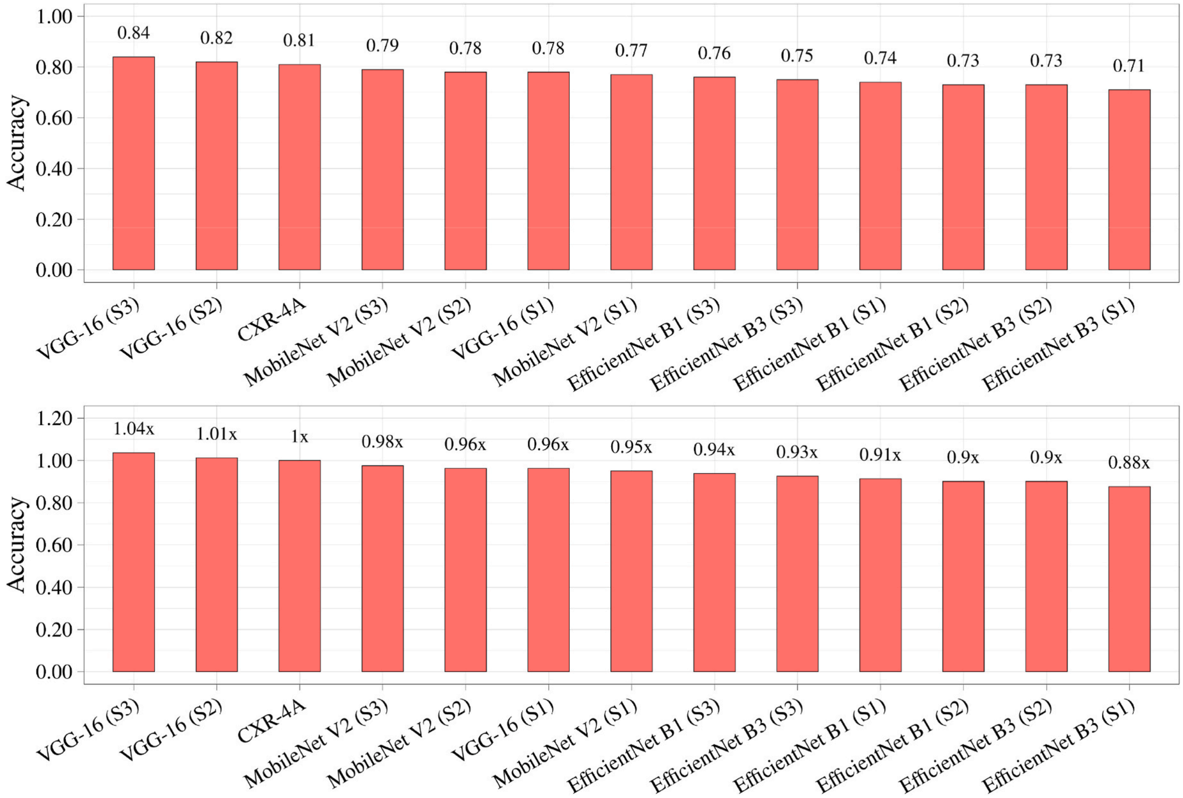
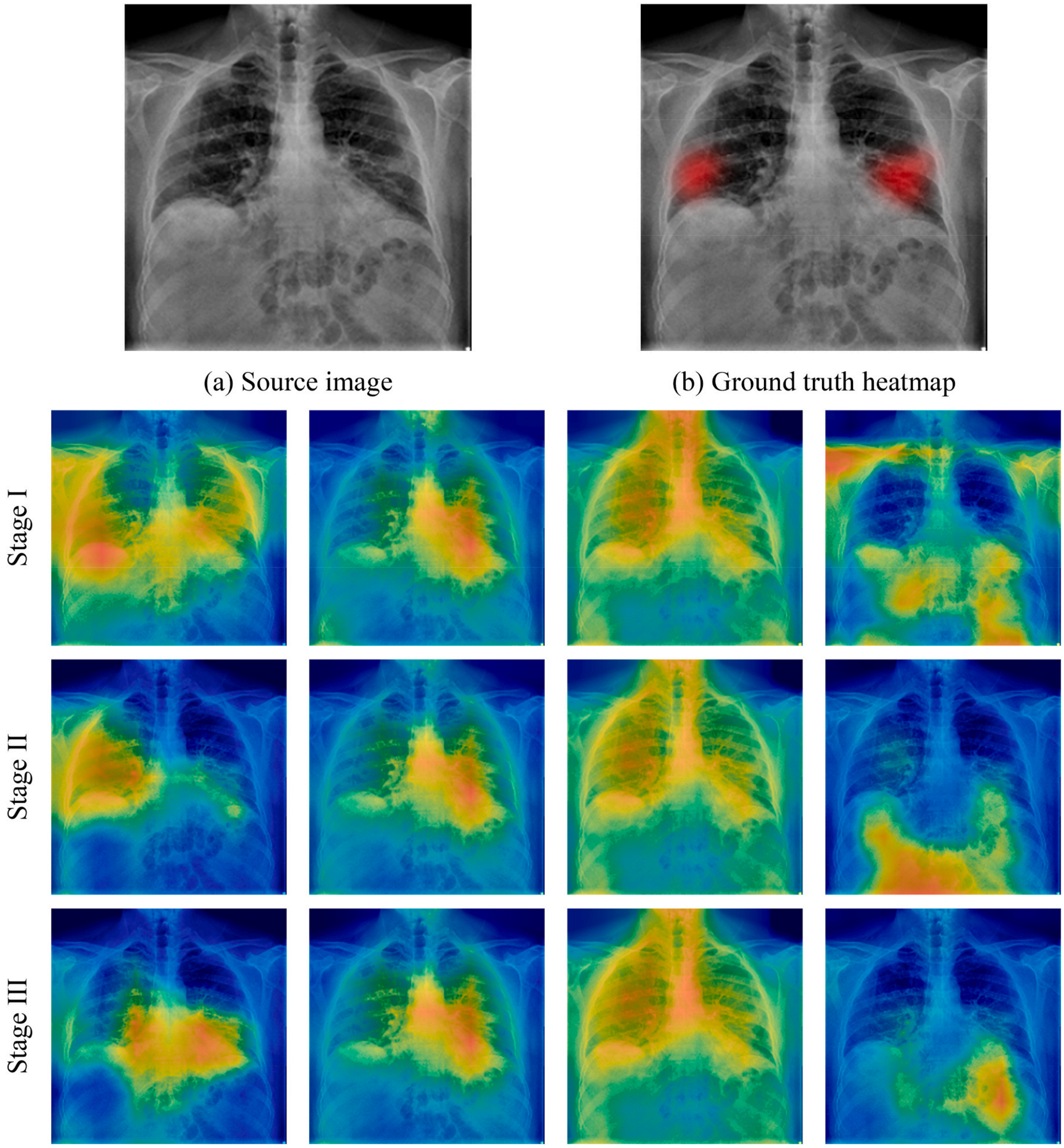


Fig. 2. Comparison of networks' accuracy based on the testing subset. The top chart is the comparison of absolute values, while the bottom chart is the comparison of relative values.

Appendix C. An additional case of COVID-19 visualization with Grad-CAM heatmaps



(a) Source image

(b) Ground truth heatmap

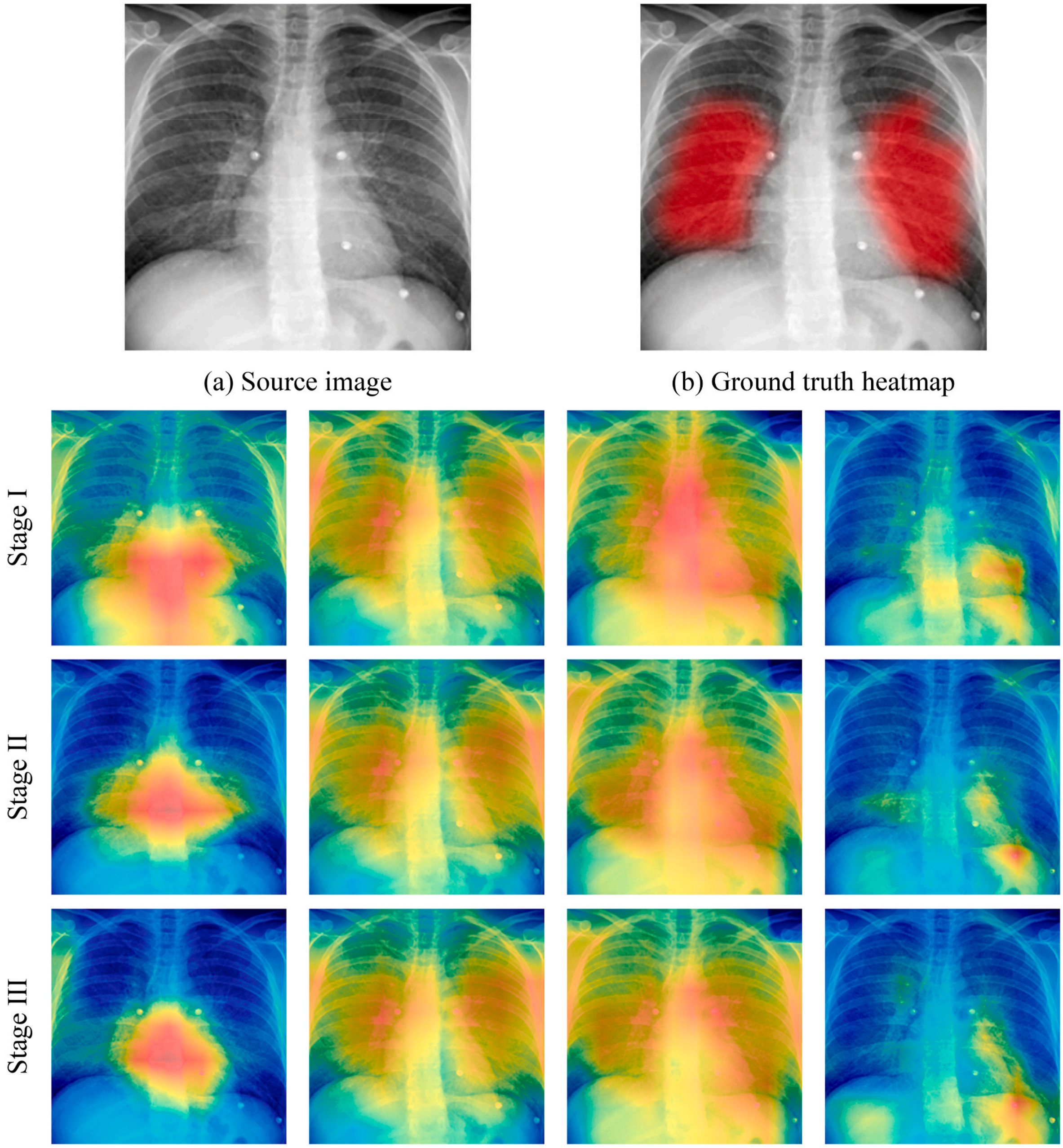
(c) MobileNet V2

(d) EfficientNet B1

(e) EfficientNet B3

(f) VGG-16

Appendix D. An additional case of pneumonia visualization with Grad-CAM heatmaps



(a) Source image (b) Ground truth heatmap

(c) MobileNet V2 (d) EfficientNet B1 (e) EfficientNet B3 (f) VGG-16

References

[1] COVID-19 virus pandemic - worldometer n.d. https://www.worldometers.info/coronavirus/. [Accessed 10 May 2021].

[2] Wang W, Xu Y, Gao R, Lu R, Han K, Wu G, et al. Detection of SARS-CoV-2 in different types of clinical specimens. JAMA, J Am Med Assoc 2020;323:1843–4. https://doi.org/10.1001/jama.2020.3786.

[3] Wikramaratna P, Paton R, Ghafari M, Lourenco J. Estimating false-negative detection rate of SARS-CoV-2 by RT-PCR. medRxiv 2020;2020. https://doi.org/10.1101/2020.04.05.20053355. 04.05.20053355.

[4] Yang Y, Yang M, Shen C, Wang F, Yuan J, Li J, et al. Evaluating the accuracy of different respiratory specimens in the laboratory diagnosis and monitoring the viral shedding of 2019-nCoV infections. medRxiv 2020:2020. https://doi.org/10.1101/2020.02.11.20021493. 02.11.20021493.

[5] Fang Y, Zhang H, Xie J, Lin M, Ying L, Pang P, et al. Sensitivity of chest CT for COVID-19: comparison to RT-PCR. Radiology 2020;296:E115–7. https://doi.org/10.1148/radiol.2020200432.

[6] Guan W, Ni Z, Hu Y, Liang W, Ou C, He J, et al. Clinical characteristics of coronavirus disease 2019 in China. N Engl J Med 2020;382:1708–20. https://doi.org/10.1056/nejmoa2002032.

[7] Huang C, Wang Y, Li X, Ren L, Zhao J, Hu Y, et al. Clinical features of patients infected with 2019 novel coronavirus in Wuhan, China. Lancet 2020;395:497–506. https://doi.org/10.1016/S0140-6736(20)30183-5.

[8] Ng M-Y, Lee EY, Yang J, Yang F, Li X, Wang H, et al. Imaging profile of the COVID-19 infection: radiologic findings and literature review. Radiol Cardiothorac Imaging 2020;2:e200034. https://doi.org/10.1148/ryct.2020200034.

[9] Kanne JP, Little BP, Chung JH, Elicker BM, Ketai LH. Essentials for radiologists on COVID-19: an update-radiology scientific expert panel. Radiology 2020;296: E113–4. https://doi.org/10.1148/radiol.2020200527.

[10] Ai T, Yang Z, Hou H, Zhan C, Chen C, Lv W, et al. Correlation of chest CT and RT-PCR testing for coronavirus disease 2019 (COVID-19) in China: a report of 1014 cases. Radiology 2020;296:E32–40. https://doi.org/10.1148/radiol.2020200642.

[11] Rubin GD, Ryerson CJ, Haramati LB, Sverzellati N, Kanne JP, Raoof S, et al. The role of chest imaging in patient management during the COVID-19 pandemic: a multinational consensus statement from the fleischner society. Chest 2020;158: 106–16. https://doi.org/10.1016/j.chest.2020.04.003.

[12] Baltruschat IM, Nickisch H, Grass M, Knopp T, Saalbach A. Comparison of deep learning approaches for multi-label chest X-ray classification. Sci Rep 2019;9:6381. https://doi.org/10.1038/s41598-019-42294-8.

[13] Jaiswal AK, Tiwari P, Kumar S, Gupta D, Khanna A, Rodrigues JJPC. Identifying pneumonia in chest X-rays: a deep learning approach. Measurement 2019;145: 511–8. https://doi.org/10.1016/J.MEASUREMENT.2019.05.076.

[14] Wang L, Lin ZQ, Wong A. COVID-Net: a tailored deep convolutional neural network design for detection of COVID-19 cases from chest X-ray images. Sci Rep 2020;10. https://doi.org/10.1038/s41598-020-76550-z.

[15] Jaiswal A, Gianchandani N, Singh D, Kumar V, Kaur M. Classification of the COVID-19 infected patients using DenseNet201 based deep transfer learning. J Biomol Struct Dyn 2021;39:5682–9. https://doi.org/10.1080/07391102.2020.1788642.

[16] Li L, Qin L, Xu Z, Yin Y, Wang X, Kong B, et al. Using artificial intelligence to detect COVID-19 and community-acquired pneumonia based on pulmonary CT: evaluation of the diagnostic accuracy. Radiology 2020;296:E65–72. https://doi.org/10.1148/radiol.2020200905.

[17] Gunraj H, Sabri A, Koff D, Wong A. COVID-net CT-2: enhanced deep neural networks for detection of COVID-19 from chest CT images through bigger, more diverse learning. ArXiv 2021.

[18] Javaheri T, Homayounfar M, Amoozgar Z, Reiazi R, Homayounieh F, Abbas E, et al. CovidCTNet: an open-source deep learning approach to identify covid-19 using CT image. ArXiv 2020.

[19] Gunraj H, Wang L, Wong A. COVIDNet-CT: a tailored deep convolutional neural network design for detection of COVID-19 cases from chest CT images. Front Med 2020;7:1025. https://doi.org/10.3389/fmed.2020.608525.

[20] Singh D, Kumar V, Vaishali, Kaur M. Classification of COVID-19 patients from chest CT images using multi-objective differential evolution–based convolutional neural networks. Eur J Clin Microbiol Infect Dis 2020;39:1379–89. https://doi.org/10.1007/S10096-020-03901-Z.

[21] Singh D, Kumar V, Kaur M. Densely connected convolutional networks-based COVID-19 screening model. Appl Intell 2021;51:3044–51. https://doi.org/10.1007/S10489-020-02149-6.

[22] Kaur M, Kumar V, Yadav V, Singh D, Kumar N, Das NN. Metaheuristic-based deep COVID-19 screening model from chest X-ray images. J Healthc Eng 2021;2021:1–9. https://doi.org/10.1155/2021/8829829.

[23] Chowdhury NK, Rahman MM, Kabir MA. PDCOVIDNet: a parallel-dilated convolutional neural network architecture for detecting COVID-19 from chest X-ray images. Health Inf Sci Syst 2020;8:27. https://doi.org/10.1007/s13755-020-00119-3.

[24] Moutounet-Cartan PGB. Deep convolutional neural networks to diagnose COVID-19 and other pneumonia diseases from posteroanterior chest X-rays. ArXiv 2020.

[25] Badawi A, Elgazzar K. Detecting coronavirus from chest X-rays using transfer learning. COVID 2021;1:403–15. https://doi.org/10.3390/covid1010034.

[26] Dash AK, Mohapatra P. A Fine-tuned deep convolutional neural network for chest radiography image classification on COVID-19 cases. Multimed Tool Appl 2021: 1–21. https://doi.org/10.1007/s11042-021-11388-9.

[27] Minaee S, Kafieh R, Sonka M, Yazdani S, Jamalipour Soufi G. Deep-COVID: predicting COVID-19 from chest X-ray images using deep transfer learning. Med Image Anal 2020;65. https://doi.org/10.1016/j.media.2020.101794.

[28] Farooq M, Hafeez A. COVID-ResNet: a deep learning framework for screening of COVID19 from radiographs. ArXiv 2020.

[29] Mahmud T, Rahman MA, Fattah SA. CovXNet: a multi-dilation convolutional neural network for automatic COVID-19 and other pneumonia detection from chest X-ray images with transferable multi-receptive feature optimization. Comput Biol Med 2020;122:103869. https://doi.org/10.1016/j.compbiomed.2020.103869.

[30] Ozturk T, Talo M, Yildirim EA, Baloglu UB, Yildirim O, Rajendra Acharya U. Automated detection of COVID-19 cases using deep neural networks with X-ray images. Comput Biol Med 2020;121:103792. https://doi.org/10.1016/j.compbiomed.2020.103792.

[31] Abbas A, Abdelsamea MM, Gaber MM. Classification of COVID-19 in chest X-ray images using DeTraC deep convolutional neural network. Appl Intell 2021;51: 854–64. https://doi.org/10.1007/s10489-020-01829-7.

[32] Mohammadi R, Salehi M, Ghaffari H, Rohani AA, Reiazi R. Transfer learning-based automatic detection of coronavirus disease 2019 (COVID-19) from chest X-ray images. J Biomed Phys Eng 2020;10:559–68. https://doi.org/10.31661/jbpe.v0i0.2008-1153.

[33] Alam N-A-A, Ahsan M, Based MA, Haider J, Kowalski M. COVID-19 detection from chest X-ray images using feature fusion and deep learning. Sensors 2021;21:1480. https://doi.org/10.3390/s21041480.

[34] Wehbe RM, Sheng J, Dutta S, Chai S, Dravid A, Barutcu S, et al. DeepCOVID-XR: an artificial intelligence algorithm to detect COVID-19 on chest radiographs trained and tested on a large U.S. Clinical data set. Radiology 2021;299:E167–76. https://doi.org/10.1148/radiol.2020203511.

[35] Bharati S, Podder P, Mondal MRH, Prasath VBS. Medical imaging with deep learning for COVID- 19 diagnosis: a comprehensive review. ArXiv 2021.

[36] Serena Low WC, Chuah JH, Tee CATH, Anis S, Shoaib MA, Faisal A, et al. An overview of deep learning techniques on chest X-ray and CT scan identification of COVID-19. Comput Math Methods Med 2021;2021:1–17. https://doi.org/10.1155/2021/5528144.

[37] Kumar V, Singh D, Kaur M, Damaševičius R. Overview of current state of research on the application of artificial intelligence techniques for COVID-19. PeerJ Comput Sci 2021;7:e564. https://doi.org/10.7717/peerj-cs.564.

[38] Toraman S, Alakus TB, Turkoglu I. Convolutional capsnet: a novel artificial neural network approach to detect COVID-19 disease from X-ray images using capsule networks. Chaos, Solit Fractals 2020;140:110122. https://doi.org/10.1016/J.CHAOS.2020.110122.

[39] Mansour RF, Escorcia-Gutierrez J, Gamarra M, Gupta D, Castillo O, Kumar S. Unsupervised deep learning based variational autoencoder model for COVID-19 diagnosis and classification. Pattern Recogn Lett 2021;151:267–74. https://doi.org/10.1016/J.PATREC.2021.08.018.

[40] Cohen JP, Morrison P, Dao L, Roth K, Duong TQ, Ghassemi M. COVID-19 image data collection: prospective predictions are the future. ArXiv 2020.

[41] Khan AI, Shah JL, Bhat MM. CoroNet: a deep neural network for detection and diagnosis of COVID-19 from chest x-ray images. Comput Methods Progr Biomed 2020;196:105581. https://doi.org/10.1016/J.CMPB.2020.105581.

[42] Chandra TB, Verma K, Singh BK, Jain D, Netam SS. Coronavirus disease (COVID-19) detection in Chest X-Ray images using majority voting based classifier ensemble. Expert Syst Appl 2021;165:113909. https://doi.org/10.1016/J.ESWA.2020.113909.

[43] Cohen JP, Morrison P, Dao L. COVID-19 image data collection. ArXiv 2020. https://github.com/ieee8023/covid-chestxray-dataset.

[44] Wang L, Wong A, Qiu ZL, McInnis P, Chung A, Gunraj H, et al. Actualmed COVID-19 chest X-ray dataset initiative 2020. https://github.com/agchung/Actualmed-COVID-chestxray-dataset.

[45] Wang L, Wong A, Qiu ZL, McInnis P, Chung A, Gunraj H, et al. Figure 1 COVID-19 chest X-ray dataset initiative 2020. https://github.com/agchung/Figure1-COVID-chestxray-dataset.

[46] COVID-19 radiography Database | kaggle n.d. https://www.kaggle.com/tawsifurrahman/covid19-radiography-database. [Accessed 10 May 2021].

[47] Chowdhury MEH, Rahman T, Khandakar A, Mazhar R, Kadir MA, Mahbub Z Bin, et al. Can AI help in screening viral and COVID-19 pneumonia? IEEE Access 2020; 8:132665–76. https://doi.org/10.1109/ACCESS.2020.3010287.

[48] RSNA pneumonia detection challenge | kaggle. Kaggle n.d. https://www.kaggle.com/c/rsna-pneumonia-detection-challenge. [Accessed 10 May 2021].

[49] Litmanovich DE, Chung M, Kirkbride RR, Kicska G, Kanne JP. Review of chest radiograph findings of COVID-19 pneumonia and suggested reporting language. J Thorac Imag 2020;35:354–60. https://doi.org/10.1097/RTI.0000000000000541.

[50] Selvaraju RR, Cogswell M, Das A, Vedantam R, Parikh D, Batra D. Grad-CAM: visual explanations from deep networks via gradient-based localization. In: Proc. IEEE int. Conf. Comput. Vis., vol. 2017- octob, institute of electrical and electronics engineers inc.; 2017. p. 618–26. https://doi.org/10.1109/ICCV.2017.74.

[51] Sandler M, Howard A, Zhu M, Zhmoginov A, Chen L-C. MobileNetV2: inverted residuals and linear bottlenecks. IEEE Comput Soc Conf Comput Vis Pattern Recogn 2018:4510–20.

[52] Huang G, Liu Z, Van Der Maaten L, Weinberger KQ. Densely connected convolutional networks. In: Proc. - 30th IEEE conf. Comput. Vis. Pattern recognition, CVPR 2017, vol. 2017- janua, institute of electrical and electronics engineers inc.; 2017. p. 2261–9. https://doi.org/10.1109/CVPR.2017.243.

[53] Tan M, Le QV. EfficientNet: rethinking model scaling for convolutional neural networks. In: 36th int conf mach learn ICML 2019 2019; 2019-June. 10691–700.

[54] Liu S, Deng W. Very deep convolutional neural network based image classification using small training sample size. In: Proc. - 3rd IAPR asian conf. Pattern recognition, ACPR 2015. Institute of Electrical and Electronics Engineers Inc.; 2016. p. 730–4. https://doi.org/10.1109/ACPR.2015.7486599.

[55] He K, Zhang X, Ren S, Sun J. Identity mappings in deep residual networks. Lect Notes Comput Sci 2016:630–45. 9908 LNCS.

[56] Szegedy C, Vanhoucke V, Ioffe S, Shlens J, Wojna Z. Rethinking the inception architecture for computer vision. In: Proc. IEEE comput. Soc. Conf. Comput. Vis. Pattern recognit., vol. 2016- decem, IEEE computer society; 2016. p. 2818–26. https://doi.org/10.1109/CVPR.2016.308.

[57] Szegedy C, Ioffe S, Vanhoucke V, Alemi AA. Inception-v4, inception-ResNet and the impact of residual connections on learning. In: 31st AAAI Conf. Artif. Intell. AAAI 2017, AAAI press; 2017. p. 4278–84.

[58] Wang L, Wong A, Qiu ZL, McInnis P, Chung A, Gunraj H, et al. COVID-net: COVID-net open source initiative n.d.

[59] Li K, Wu Z, Peng KC, Ernst J, Fu Y. Tell me where to look: guided attention inference network. In: Proc. IEEE comput. Soc. Conf. Comput. Vis. Pattern recognit., IEEE computer society; 2018. p. 9215–23. https://doi.org/10.1109/CVPR.2018.00960.

[60] Zhou B, Khosla A, Lapedriza A, Oliva A, Torralba A. Learning deep features for discriminative localization. In: Proc. IEEE comput. Soc. Conf. Comput. Vis. Pattern recognit., vol. 2016- decem, IEEE computer society; 2016. p. 2921–9. https://doi.org/10.1109/CVPR.2016.319.