PDCOVIDNet: A Parallel-Dilated Convolutional Neural Network Architecture for Detecting COVID-19 from Chest X-Ray Images

Nihad Karim Chowdhury
Department of Computer Science and Engineering
University of Chittagong
Bangladesh
nihad@cu.ac.bd

Md. Muhtadir Rahman
Department of Computer Science and Engineering
University of Chittagong
Bangladesh
muhtadirrahman@gmail.com

Muhammad Ashad Kabir
School of Computing and Mathematics
Charles Sturt University,NSW
Australia
akabir@csu.edu.au

July 15, 2020

Abstract

The COVID-19 pandemic continues to severely undermine the prosperity of the global health system. To combat this pandemic, effective screening techniques for infected patients are indispensable. There is no doubt that the use of chest X-ray images for radiological assessment is one of the essential screening techniques. Some of the early studies revealed that the patient's chest X-ray images showed abnormalities, which is natural for patients infected with COVID-19. In this paper, we proposed a parallel-dilated convolutional neural network (CNN) based COVID-19 detection system from chest x-ray images, named as Parallel-Dilated COVIDNet (PDCOVIDNet). First, the publicly available chest X-ray collection fully preloaded and enhanced, and then classified by the proposed method. Differing convolution dilation rate in a parallel form demonstrates the proof-of-principle for using PDCOVIDNet to extract radiological features for COVID-19 detection. Accordingly, we have assisted our method with two visualization methods, which are specifically designed to increase understanding of the key components associated with COVID-19 infection. Both visualization methods compute gradients for a given image category related to feature maps of the last convolutional layer to create a class-discriminative region. In our experiment, we used a total of 2,905 chest X-ray images, comprising three cases (such as COVID-19, normal, and viral pneumonia), and empirical evaluations revealed that the proposed method extracted more significant features expeditiously related to suspected disease. The experimental results demonstrate that our proposed method significantly improves performance metrics: the accuracy, precision, recall and F1 scores reach 96.58%, 96.58%, 96.59% and 96.58%, respectively, which is comparable or enhanced compared with the state-of-the-art methods. We believe that our contribution can support resistance to COVID-19, and will adopt for COVID-19 screening in AI-based systems.

1 Introduction

Coronavirus or covid-19 is a contagious disease that was caused by the SARS-CoV-2 (Severe Acute Respiratory Syndrome Coronavirus 2). The disease was first discovered and became prevalent in Wuhan, Hubei Province, China, and has since spread around the world. As we know, on March 11, 2020, the World Health Association (WHO) proclaimed the flare-up of coronavirus pandemic [1]. Until now, coronavirus pandemic has devastatingly affected the prosperity of the world economy and health system. As of July 12, 2020, more than 12,401,262 confirmed cases of COVID-19 and 559,047 confirmed deaths due to the disease [2]. Contemporaneous observations detail that COVID-19 infection causes lung illness, hence, making it troublesome to breathe. As well, a person infected with coronavirus spread the virus among others primarily through coughs or sneezes as tiny droplets, therefore, a legitimate social framework for a disengagement is obligatory to delimit the disease. While attempting to deal with growing numbers of COVID-19 infected cases, all nations try to increase testing to reduce the pandemic ahead of time. At present, it is vital to extend an effective screening strategy within reach to distinguish COVID-19 cases and segregate infected individuals from others while hospitals are striving with scaling up abilities to meet the increasing number of patients [3]. An ultimate goal against COVID-19 battle is to provide prompt treatment to infected individuals using a viable screening technique. As we all know, reverse transcriptase-polymerase chain reaction (RT-PCR) is a reliable and well-documented screening technology, which has been accepted by researchers and clinical staff, but the flexibility and strict necessity of conditions at the clinical laboratory will greatly delay the accurate detection of suspicious patients [4]. Moreover, starting the results of RT-PCR testing in China indicated moderately poor sensitivity, and this issue occurred due to the variable characteristics of testing [3]. Further, subsequent revelations illustrated a noteworthy factor that positive rate declined with time after symptom incipience at once relied on how the specimen collected [5].

On the other hand, the limitations of RT-PCR testing have prompted researchers to find a rapid and definitive method for diagnosing COVID-19 infection. Even though COVID-19 pandemic will significantly affect the global healthcare system, a radiography testing such as a chest X-ray would be a discretionary screening technique that has as well utilized for COVID-19 diagnosis. Certainly, chest X-ray is beneficial in emergency diagnosis and treatment considering, the way that this system is quick and simple to operate, and radiologists can yet recognize. In the prior observations, researchers observed that patients exhibit inconsistencies in chest X-ray images that are typical for those infected with COVID-19 [6]. However, a few British hospitals are presently turning out artificial intelligence (AI) based disease recognition systems to look for potential symptoms of Covid-19 infection in chest X-ray images, such as patterns of opacity in the lungs [7]. Significant diagnostic accuracy in medical imaging, which is an indispensable objective of automatic disease detection, has already achieved in recent years using CNN, as the core advancement of the rising AI. Inspired by this, recently, a lot of research has already accomplished by utilizing CNN modalities to detect COVID-19 using chest X-ray images, and has obtained promising results [3, 8–11].

The purpose of this research is to extend the curative effect using chest X-ray images showing COVID-19 infection. In this regard, we consider the CNN-based framework as it is renowned for its excellent recognition performance in object classification. In CNN, convolution is one of the core operations to regulate the architecture through its dilation rate. The traditional discrete convolution performs the convolution process at a dilation rate of 1, while the dilated convolution [12] uses a different dilation rate to perform the convolution process, and the proposed PDCOVIDNet varies dilation rate in a parallel stack of convolution layer in CNN, thus reflecting more distinctive features. In our experimental evaluation, 219 chest X-ray images of COVID-19, 1341 normal, and 1345 viral pneumonia were collected and processed and used to train, validate, and test the proposed method. The benchmark data set is publicly accessible [13], and the authors of this benchmark generated data from three different open access data repositories containing chest X-ray images [14–16]. The pipeline of PDCOVIDNet starts with the data augmentation strategy, and then optimizes and fine-tunes the settings to train CNN modalities that expand in parallel, through generating dominant features at different scale shifting receptive fields. Next, the generated features are fused into a neural network system to produce the final prediction. Also, we use two gradient-weighted class activation maps (such as Grad-CAM [17] and Grad-CAM++ [18]) to aid our system. These maps provide predictive explanations, and can identify important features related to COVID-19 infection. From experimental evaluation, it shows that the proposed method can identify important features related to COVID-19 disease, and the best accuracy achieved is 96.58%. The contributions of this paper are as follows:

- We propose and develop a novel CNN framework called PDCOVIDNet to detect COVID-19 from chest X-ray images. Our proposed framework uses a dilated convolution in the parallel stack of convolutional blocks that can capture and propagate important features in parallel over the network which enhances detection accuracy significantly.

- We visualize the X-ray images to analyze the COVID and non-COVID cases, and further investigate the incorrect classification.

- Finally, we empirically evaluate our approach with the state-of-the-art approaches to highlight the effectiveness of PDCOVIDNet in detecting COVID-19.

We organize the rest of this paper in the following ways. Section 2 reviews the state-of-the-art models used in detecting COVID-19 using chest X-ray images. The benchmark dataset and the augmentation strategy are described in Section 3. Next, Section 4 explains the main details of the proposed model and its adjustment to the detection of COVID-19 cases. In Section 5, we provide the experimental results and show the comparison between PDCOVIDNet and other models. Observation on visualization techniques and incorrect classification results are illustrated in Section 6 and Section 7, respectively. Finally, Section 8 provides the conclusion of this paper with the future research direction.

2 Related Works

With the rapid spread of COVID-19 in many countries around the world, imaging technology can quickly detect COVID-19, which helps to control the spread of disease. Chest X-ray is a promising imaging technology with a historical prospect of an image diagnosis system. It can be fully explored through various feature extraction methods, thereby playing an important role in the diagnosis of COVID-19 disease.

Due to the need for a faster interpretation of chest X-ray images, a CNN-based AI system provides [19] a comprehensive overview of the latest application areas of AI for COVID-19, which mentions medical imaging for diagnosis. Halgurd et al. [20] proposed a customized CNN and pre-trained AlexNet [21] model through chest X-ray images, the results showed very promising accuracy in detecting patients infected either COVID-19 or normal with 98% and 94.1% accuracy respectively. By using the pre-trained ResNet50 [22] model, Narin et al. [8] obtained 98% accuracy. The authors also evaluated several CNN architectures for COVID-19 detection. The Inception [23] transfer learning model modified to extract radiological features for accurate COVID-19 diagnosis, and then experienced internal and external validation with an accuracy of 89.5% [24]. Also, the CNN architecture based on transfer learning, named as COVID-Net [3], used to classify chest X-ray images into three categories: normal, non-COVID and COVID-19 infections. It showed that COVID-Net achieved the best accuracy by achieving a test accuracy of 93.3%. In [10], transfer learning adopted in state-of-the-art convolutional neural network architectures, results demonstrated the proof-of-principle for using CNN with transfer learning to extract radiological features from small data sets. Biraja et al. [25] investigated the uncertainty in the CNN solution based on Bayesian Convolutional Neural Network (BCNN) to improve the diagnostic performance of COVID-19. Another CNN based on ResNet50 used by Bukhari et al. [26] to prove the usefulness and diagnostic accuracy of patients infected with COVID-19 having accuracy of 98.18%. The authors [27] of proposed a deep learning architecture for COVID-19 detection utilizing their benchmark in which images exhibit COVID-19 disease verified by radiologists, by fine-tuning four renowned pre-trained convolutional models (ResNet18 [22], ResNet50, SqueezeNet [28], and DenseNet121 [29]), thereby yielded promising results in terms of sensitivity and specificity. A deep learning framework based on EfficientNet [30] proposed by Eduardo et al. [31]. Contrasted with some popular baseline models (such as VGG16 [32] and ResNet50), the learning parameters have greatly reduced, and the COVID-19 case has a 100% positive prediction. Farooq et al. [33] proposed a method called COVID-ResNet that uses a three-step technique, which includes gradually adjusting the size, looping the learning rate search and discriminating the learning rate, and then fine-tuning the pre-trained ResNet50 architecture to improve model performance. A deep learning model called DarkCovidNet [34] proposed for the automatic detection of COVID-19 using chest X-ray images. The average classification accuracy of binary classification (such as COVID and No-Findings) was 98.08%, and the average classification accuracy of multi-class classification (such as COVID, No-Findings and pneumonia) was 87.02%. Finally, the author provided an intuitive explanation and evaluated by expert radiologists.

In a case of AI-based COVID-19 diagnostics system, it is obligatory to verify results with expert radiologists. Almost all methods have shown encouraging results. To the best of our knowledge, a few methods are different only if they have verified by radiologists. Several methods focus only on quantitative analysis, while other methods focus on qualitative analysis using visualization and localization techniques to prove that their analysis can be used for COVID-19 detection, which aids to allow for human-interpretable explanation. In literature reviews, there are a few model and prediction level integration strategies to mitigate variance errors and enhance the performance and generalization of CNN models. Finally, due to the inadequate number of COVID-19 cases, creating ample benchmarks is a major challenge in COVID-19 detection. Considering a small data set, running a large number of iterative CNN architectures can lead to overfitting.

3 Data Pre-processing

First, we will introduce the benchmark data set and the expansion strategy in detail to aid the training of the proposed model. Next, we will discuss in detail the proposed PDCOVIDNet architecture design method and the training strategy covering the optimal parameter adjustment. Finally, in order to make suspicious disease detection more convincing, we will integrate visualization techniques to highlight key issues with visual markers.

3.1 Chest X-ray image benchmark

The benchmark data set [13] used in our experimental evaluation consists of three main categories (such as COVID-19, Normal, and Viral Pneumonia), yielding 219 COVID-19 positive, 1341 normal and 1345 viral pneumonia chest X-ray images. In the case of accumulating COVID-19 positive images, the authors used two open-access repositories, such as the Italian Society of Medical and Interventional Radiology (SIRM) COVID-19 database [14], and the Novel Corona Virus 2019 dataset developed by Joseph et al. [15]. On the other hand, P. Mooney [16] created normal and viral pneumonia images from the chest X-ray image (pneumonia) database used for this benchmark [13]. Moreover, the benchmark is public, and metadata is distributed to provide appropriate document guidance to generate references to each image. Since resizing is one of the essential steps in data preprocessing, all images resized to 224×224 pixels. Figure 1 shows sample images from the benchmark dataset, including COVID-19, normal and viral pneumonia. As shown in the Table 1, we have trained, validated and tested the images in an appropriate ratio.
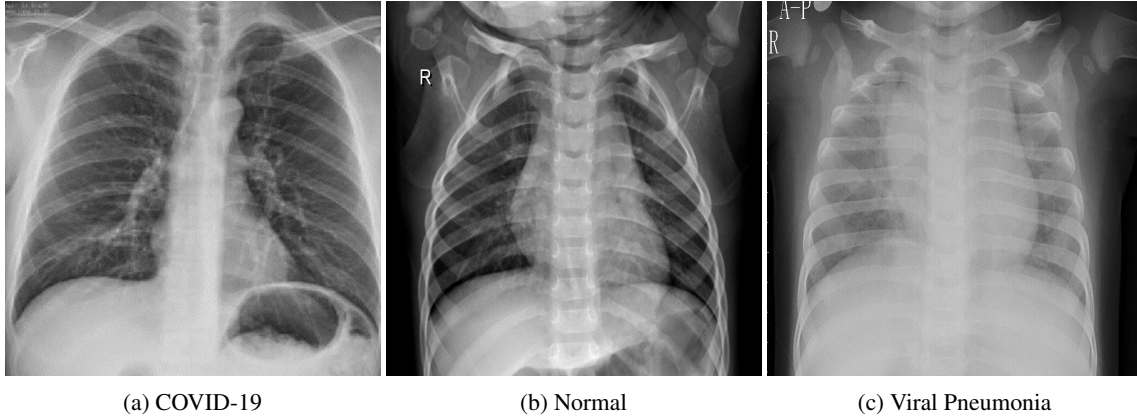

(a) COVID-19	(b) Normal	(c) Viral Pneumonia

Figure 1: Some image labels available in the benchmark dataset [13].

Table 1: Images partition in Training, Validation and Testing

Category	COVID-19	Normal	Viral Pneumonia	Total
Training	175	1072	1076	2323
Validation	21	134	134	289
Testing	23	135	135	293

3.2 Data Augmentation

In order to properly train the CNN model, it is often useful to manually increase the size of the data set using data enhancement that reduces noise and preserves the original quality. This process is executed just-in-time during the training process, so the performance of the model can be improved by solving the problem of overfitting. For image augmentation, we have many options to choose values of different scales, including horizontal flip, height and width offset, rotation, shearing, zoom, and fill modes. Each option has its ability to represent images in different ways to provide important features during the training phase and thus enhances the model's performance better. Table 2 shows the image augmentation settings used in our experiment.

Table 2: Images augmentation settings

Option	Value
rotation range	30
height shift	0.15
width shift	0.15
shear range	0.10
zoom range	0.10
fill mode	nearest

4 PDCOVIDNet Architecture

In this section, we will briefly describe our proposed PDCOVIDNet architecture. In our proposed model, we have three main components, such as feature extraction, detection, and visualization. First of all, our proposed PDCOVIDNet is a parallel stack of convolutional layers, activation layer and max-Pooling layer. Then, we add parallel layers at the feature level, and perform a convolution again with activation on resulting feature maps. Afterward, the flattened features provide into two layers of Multi-Layer Perceptrons (MLP), but an adjustment needs to determine the proportion of neurons at each layer that drop, to avoid overfitting. Finally, the last layer with softmax activation function performs the classification task, and then generates a class activation map, which acts as an interpreter of classification, merged with the last convolution layer. Figure 2 shows the overall system architecture of the proposed PDCOVIDNet. We split the workflow into two parts: the feature extraction phase and the classification and visualization phase. In the next section, we will briefly explain the feature extraction process.
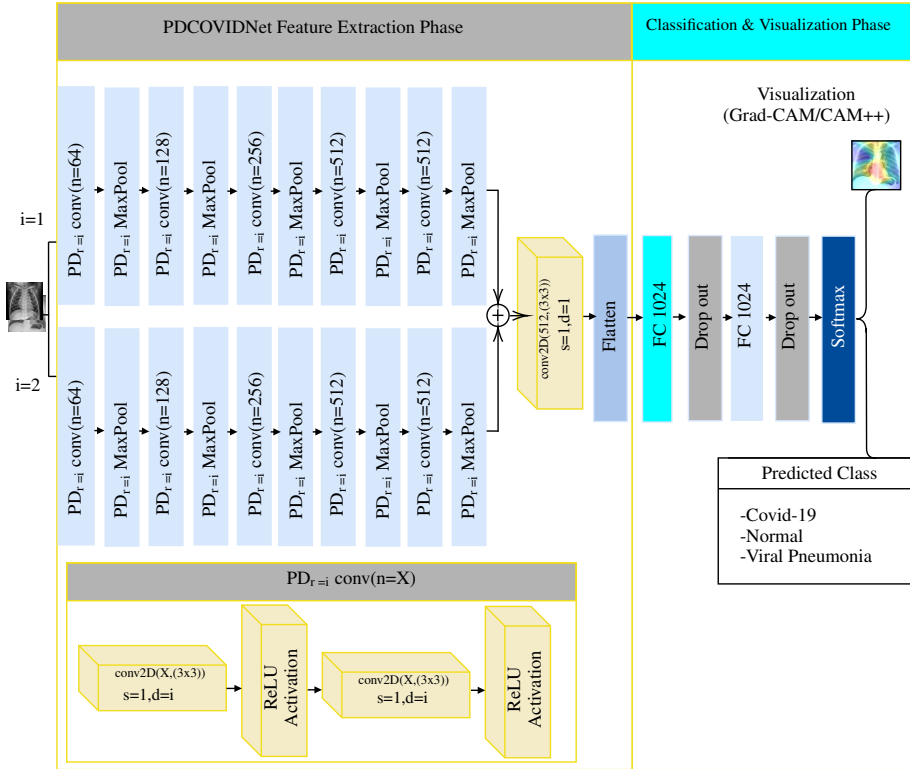
4.1 Feature Extraction

Figure 2: An overall system architecture of PDCOVIDNet.

To obtain a suitable network architecture, different numbers of filters in each convolution layer, filter sizes, different layers in MLP and different hyper-parameters have experimented. In the first stage, PDCOVIDNet consists of five dilated convolutional blocks, expressed as $PD_{r=i}conv(n=X)$, that are alternately max-pooled. Figure 3 illustrates how dilated convolution is incorporated into our proposed model. As shown in Figure 3, the input image provides in two $PD_{r=i}conv(n=X)$ blocks in parallel, only changing the dilation rate, such as $d = 1, ..., N$. A convolution with a dilation rate of 1 is equivalent to a standard convolution, while a convolution with a dilation rate of greater than 1 expands the receptive field when processing input at a higher resolution, thereby achieving fine details of the image. The receptive field refers to the portion of the image where a filter extracts feature without change filter size, and is simply an input with a fixed gap, i.e., if there is dilation of d, then each input skips of $d - 1$ pixels. According to this definition, considering that our input is a $2D$ image, the dilation rate of 1 is a standard convolution, and dilation rate of 2 means that each input skips a pixel. To understand the relation between the dilation rate of d and the receptive field size of rf,
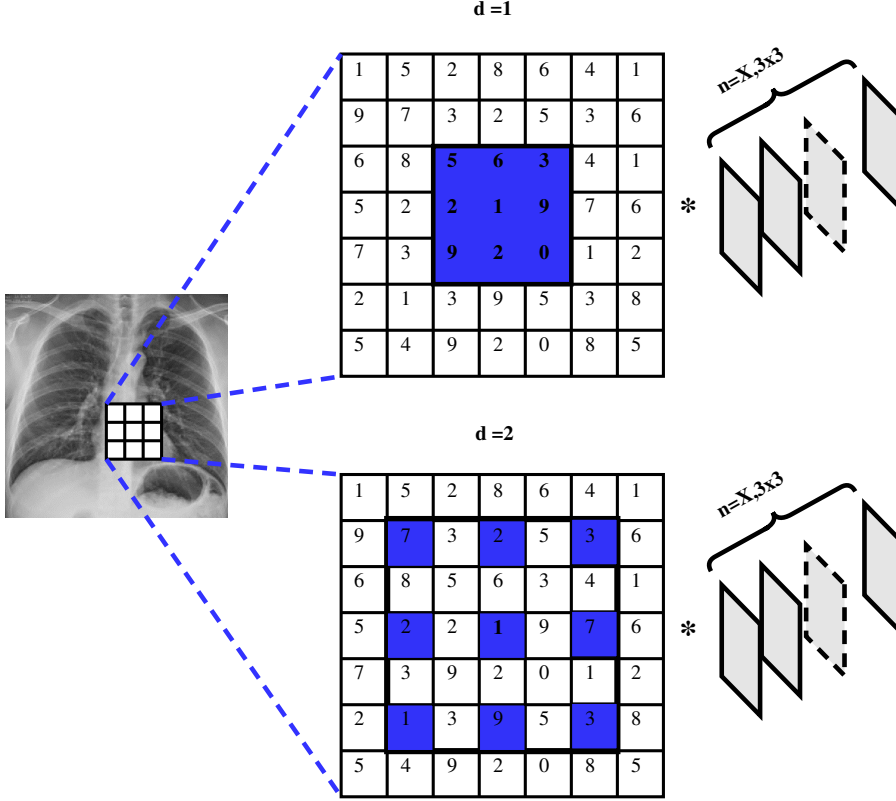
Figure 3: Dilated convolution and its activation in PDCOVIDNet. At the top, we have a dilated convolution with a dilation rate of 1, and the corresponding receive field size is 3×3, which is equivalent to the standard convolution. It has a 5×5 receiving field size at the bottom and a dilation rate of 2. All dilated convolutions have a kernel size of 3×3 and n filters.

it is often useful to understand the effect of d on rf when the kernel is fixed in size. Equation 1 [35] depicts the form of a receptive field size where the kernel of size k is dilated by the factor d.

$$rf = d(k - 1) + 1. \tag{1}$$

Equation 1 refers to form the following equation that renders the size of the output o, where $m \times m$ input with a dilation factor, padding and stride of d, p and s respectively.

$$o = \left\lfloor \frac{m + 2p - rf}{s} \right\rfloor + 1. \tag{2}$$

After using two receptive fields of different sizes, it captures important features in the observation area at different scales. In our proposed model, each $PD_{r=i}conv(n=X)$ block includes two convolution layers and activation layers, where each block consists of a different number of filter response (i.e. X=[64, 128, 256, 512, 512]) of 3×3 filter size having stride and dilation rate with 1 and d, respectively. We define a convolution layer with filters $\mathbf{F} \in \mathbb{R}^{1 \times n}$ given as

$$\mathbf{F} = \begin{bmatrix} f_1^{k \times k} & f_2^{k \times k} & \cdots & f_{n=X}^{k \times k} \end{bmatrix}, \tag{3}$$

where $k \times k$ is filter size and X is total number of filters. For a dilated convolution with a dilation rate of d on $m \times m$ input feature maps at layer l, the convolution generates feature maps from the input, denoted as $\mathbf{X}_l^{m \times m}$, and calculated by

$$\mathbf{X}_l^{o \times o} = \mathbf{X}_{l-1, d=i}^{m \times m} * \mathbf{F}_l + \mathbf{B}_l, \tag{4}$$

where $\mathbf{B}_l$ is a bias of the l-layer, and $\mathbf{F}_l$ is the l-layer filter of size $k \times k$. The features of layer l are generated at the dilation rate of d with the feature map in layer $(l - 1)$. After the convolution layer, we introduce a nonlinear layer with an activation function that uses the features generated at an earlier stage to create a new feature map as output. In the case of activation, we prefer the rectified linear unit (ReLU) [36] because it can integrate the nonlinear layer

6

and the rectification layer in CNN. ReLU has several advantages, and most importantly, it can effectively propagate gradients. Therefore, if the initial weight takes into account the unique characteristics of CNN, the possibility of gradient disappearance can be reduced. Note that the activation function performs element-by-element operations on the input feature map, so the output is the same size as the input. Assuming that the layer l is the active layer of the n-th filter, it obtains the input feature $\mathbf{X}_{(l-1),n}^{m \times m}$ with the feature map $m \times m$ from the previous convolution layer, and generates the same number of features defined as:

$$\mathbf{X}_{l,n}^{m \times m} = max(\mathbf{X}_{(l-1),n}^{m \times m}, 0) \tag{5}$$

where $\mathbf{X}_{l,n}^{m \times m}$ maps negative values to zero.

The proposed model has a PD$_{r=i}$conv(n=X) block, followed by a max-Pooling layer, and is set five times in parallel successive form. The proposed model has a PD$_{r=i}$conv(n=X) block, followed by a max-Pooling layer five times in parallel successive form. Max-Pooling, which takes the maximum value in each window, is an efficient approach to downscale the filtered image, because when using a filter size of 2×2 with a stride of 2, three-fourths of generated features are ignored in each layer substantially it reduces the computational complexity for the next layer. The max-pooling window used in our experiment was 2×2 and the stride was 2, because as reported in earlier studies [37], the overlapping window did not improve significantly over the non-overlapping window. Then, in Figure 2, we see that the features generated from the parallel branches are concatenated and provided to the next convolution layer. The inspiration behind this concatenation-convolution operation is that each branch generates features from images at different layers of CNN have different properties, so we concatenate low-level features of parallel branches to explore feature relationship of dilated convolution hence final convolution layer might detect dominant features for classification. In the last convolution layer, a total of 512 filters with a filter size of $k \times k$ and a dilation rate of 1 are applied to create final low-level features followed by ReLU activation. After that, we inaugurate a flatten layer, which converts the square feature map into a one-dimensional feature vector and prepares it for the next phase, which is finally a classification task. Our final task is the classification and visualization phase, which will briefly illustrate in the next section.

4.2 Classification and Visualization

At this stage, a two-layer MLP (often called a fully connected (FC) layer) feeds the results of the flatten layer through two neural layers to perform the classification task. It attempts to render the activation from the previous FC layers into class scores (i.e., in classification). In addition, we include a Dropout cite JMLR: v15: srivastava14a layer after each FC layer. This layer can randomly discard some FC layer weights during training to reduce overfitting. The number of randomly selected weight drops is defined by the dropout limit, which ranges from 0 to 100%. Indeed, the best adjustment is to determine the optimal number of weights to use in each layer and the dropout ratio to avoid overfitting at the same time, making the network more robust. In this study, we chose a dropout size of 0.3, two FC layers of size 1024 and 1024, respectively, and used the softmax activation function to determine the classes of the input chest X-ray images as COVID-19, normal and viral pneumonia. Finally, the layer details of the proposed model are shown in Table 3.

Although CNN models are powerful in producing impressive results, there are still many questions about why and how to produce such good results. Owing to its black-box nature, it is sometimes challenging to adopt it in a real-life application (such as a medical diagnosis system) where we need an interpretable model. However, early studies [17, 18, 38] focused their attention on visualizing the behavior of CNN models, and various visualization methods emphasized the importance of distinguishing classes, so they could execute models with interpretability. In our proposed model, we use Grad-CAM and Grad-CAM++ to highlight the important regions that are class-discriminative saliency maps, where the model emphasizes a gradient-based approach that computes the gradients for a target image class on the feature maps of the final convolution layer in a CNN model. For a given image, let $A_{i,j}^k$ denotes the activation map at a spatial location (i, j) for the k-th filter. The class-discriminative saliency map L^c for the target image class c is then computed as [17]:

$$L_{i,j}^c = ReLU(\sum_k w_k^c A_{i,j}^k) \tag{6}$$

In Eq.6, the role of ReLU is to capture features that have a positive impact on the target class. Then, in the case of Grad-CAM, gradients that are flowing back to the final convolutional layer are globally averaged to calculate the target class weights of the k-th filter, as described in Eq.7. Here, Z is the total number of pixels in the activation map, and Y^c is the probability that the target category is classified as c.

$$w_k^c = \frac{1}{Z} \sum_i \sum_j \frac{\partial Y^c}{\partial A_{i,j}^k} \tag{7}$$

On the other hand, Grad-CAM++ contributes to the weighted average of pixel-level gradients rather than the global average of gradients, so the pixel weights in a particular feature map contribute to the overall decision of detection.

Table 3: PDCOVIDNet layer details

Layer	filter size/X, Stride	PDCOVIDNet (Output Size)	
		Branch	
Input	-	$PD_{r=1}conv(n=X)$	$PD_{r=2}conv(n=X)$
Conv2D	$3 \times 3/64, 1$	$224 \times 224 \times 64$	$224 \times 224 \times 64$
Conv2D	$3 \times 3/64, 1$	$224 \times 224 \times 64$	$224 \times 224 \times 64$
MaxPooling2D	$2 \times 2/64, 2$	$112 \times 112 \times 64$	$112 \times 112 \times 64$
Conv2D	$3 \times 3/128, 1$	$112 \times 112 \times 128$	$112 \times 112 \times 128$
Conv2D	$3 \times 3/128, 1$	$112 \times 112 \times 128$	$112 \times 112 \times 128$
MaxPooling2D	$2 \times 2/128, 2$	$56 \times 56 \times 128$	$56 \times 56 \times 128$
Conv2D	$3 \times 3/256, 1$	$56 \times 56 \times 256$	$56 \times 56 \times 256$
Conv2D	$3 \times 3/256, 1$	$56 \times 56 \times 256$	$56 \times 56 \times 256$
MaxPooling2D	$2 \times 2/256, 2$	$28 \times 28 \times 256$	$28 \times 28 \times 256$
Conv2D	$3 \times 3/512, 1$	$28 \times 28 \times 512$	$28 \times 28 \times 512$
Conv2D	$3 \times 3/512, 1$	$28 \times 28 \times 512$	$28 \times 28 \times 512$
MaxPooling2D	$2 \times 2/512, 2$	$14 \times 14 \times 512$	$14 \times 14 \times 512$
Conv2D	$3 \times 3/512, 1$	$14 \times 14 \times 512$	$14 \times 14 \times 512$
Conv2D	$3 \times 3/512, 1$	$14 \times 14 \times 512$	$14 \times 14 \times 512$
MaxPooling2D	$2 \times 2/512, 2$	$7 \times 7 \times 512$	$7 \times 7 \times 512$
Add	-	$7 \times 7 \times 512$	
Conv2D	$3 \times 3/512, 1$	$7 \times 7 \times 512$	
Flatten	-	25088	
FC	-	1024	
FC	-	1024	
Softmax	-	3	

Grad-CAM++ redevelops Eq.7 to ensure that their contribution to the weighted average of the gradients remains unchanged without losing generality, i.e.,

$$w_k^c = \sum_i \sum_j \alpha_{i,j}^{kc}.ReLU\left(\frac{\partial Y^c}{\partial A_{i,j}^k}\right) \tag{8}$$

$$\alpha_{i,j}^{kc} = \frac{\frac{\partial^2 Y^c}{(\partial A_{i,j}^k)^2}}{2\frac{\partial^2 Y^c}{(\partial A_{i,j}^k)^2} + \sum_a \sum_b A_{a,b}^k \frac{\partial^3 Y^c}{(\partial A_{i,j}^k)^3}} \tag{9}$$

In Eq.9, (i,j) and (a,b) are iterators over the same activation map A^k [18].

5 Experimental Evaluation

In this section, we will present the performance of our proposed model to classify chest X-ray images, broadly categorized into three classes: COVID-19, Normal and Viral Pneumonia. In Section 4, a brief description of the benchmark data set used in the experiment and the augmentation approach are discussed. The experiment sets the training, validation, and test ratios to 80%, 10%, and 10%, respectively. We compared our proposed PDCOVIDNet with VGG16, ResNet50, InceptionV3 [39] and DenseNet121, and did not use any pre-trained weights (such as ImageNet) since ImageNet weights come from images of general objects, not chest X-ray images. All of our experiments are executed in Keras with the TensorFlow backend.

5.1 Hyper-parameters Tuning

Hyper-parameters become critical because they directly control the behavior of the model, so fine-tuned hyper-parameters have a huge impact on the performance of the model. We used the Adam [40] optimizer to train 50 epochs for each model with a learning rate of $1e - 4$, with a batch size of 32. In addition, we applied the categorical cross-entropy loss function to the training, which measures the loss between the probability of the class predicted from the softmax activation function and the true probability of the category.

5.2 Performance Evaluation Metrics

For experimental evaluations, we utilized several evaluation metrics such as Accuracy, Precision, Recall, and F1 score, i.e.,

$$Accuracy = \frac{TP + TN}{Total\ Samples} \tag{10}$$

$$Precision = \frac{TP}{TP + FP} \tag{11}$$

$$Recall = \frac{TP}{TP + FN} \tag{12}$$

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{13}$$

where TP stands for true positive, while TN, FP, and FN stand for true negative, false positive, and false negative, respectively. The $F1$ score may be a more reliable measure because the benchmark dataset is unbalanced, such as COVID-19 with 219 images and non-COVID with 2686 images. Subsequently, we used the ROC (Receiver Operating Characteristics) curve to display the results and measured the area under the ROC curve (often called AUC (Area Under the Curve)) to provide information about the effectiveness of the model.

5.3 Evaluation of individual model

The overall results are shown in Table 4 and Table 5, where Table 4 describes the class-wise classification results on different evaluation metrics, and Table 5 shows the weighted average results. From the table 4, we can see that almost all models tend to enhance the classification of most categories (such as normal and viral pneumonia) because they have more training weights than the COVID-19 case. For COVID-19, the highest performance belongs to PDCOVIDNet, whose precision, recall, and F1 scores are 95.45%, 91.30%, and 93.33%, respectively. Our model provides consistent results for the precision, recall, and F1 under normal cases, with each performance index being 97.04%, and the recall for ResNet50 is 97.78%, slightly higher than PDCOVIDNet. Also, the precision and F1 scores of DenseNet121 are 95.52% and 95.17%, respectively, which is comparable to PDCOVIDNet. Next, in the case of viral pneumonia, the precision, recall, and F1 score of PDCOVIDNet are 96.32%, 97.04%, and 96.68%, respectively. In particular, for cases of viral pneumonia, ResNet50 is only 1% better precision than PDCOVIDNet. In the Table 4, the accuracy of all evaluation models is summarized, and it can be seen that PDCOVIDNet is superior to other models. At the same time, it is evident that PDCOVIDNet has the ability to resist class imbalances since COVID-19 cases are smaller than normal or viral pneumonia cases. However, the more structured residual blocks of the model, the worse the classification performance (e.g., ResNet50). As shown in Table 5, considering the weighted average of all performance evaluation indicators, the best results are obtained by using PDCOVIDNet. In the weighted average comparison, PDCOVIDNet's results are much better than other models, which can be explained by the fact that the proposed model can extract feature maps at different scales from chest X-ray images. In particular, compared with PDCOVIDNet, Densenet121 is missing 2% in each evaluation indicator. Although ResNet50 provides the best performance for normal and viral pneumonia, unexpectedly, it fails to achieve the most successful model in the performance measurement.

It is often hard to measure the performance of the model using precision, recall and accuracy, so we need to look at the ROC curve which allows a false positive rate since it plots the true positive rate against a false positive rate. In Figure 4, ROC curves show the micro and macro average and class-wise AUC scores achieved with the PDCOVIDNet, and show consistent AUC scores across all classes, indicating stable predictions of the proposed model. In ROC curves, we obtained AUC scores of 0.9918, 0.9927, and 0.9897 for COVID-19, normal and viral pneumonia, respectively. We can see that the area under the curve of all classes is relatively similar, but normal's AUC is slightly higher than other classes.

Figure 5 shows the confusion matrices for all evaluated models. In Figure 5, it is clear that of the 23 test images, two of the COVID-19 images are classified as normal and viral pneumonia, and of the 135 images, only one image of viral pneumonia is related to COVID-19, but none of the normal images belong to COVID-19. One of the reasons may be that COVID-19 is an especial case of viral pneumonia, so they have common features that mislead the PDCOVIDNet model. It becomes clear that the VGG16 model has the same recall for the classification of normal and viral pneumonia, although it shows a significant decline in the positive prediction of COVID-19 cases. However, ResNet50 shows the ability to detect normal images, but in the case of COVID-19 detection, it shows almost the same performance as the VGG16 model, although it shows inadequate performance when predicting viral pneumonia images. As shown in the confusion matrix (Figure 5(d)), we can say that InceptionV3 can correctly classify more cases of viral pneumonia than COVID-19 and normal cases. Next, DenseNet121 demonstrates the same performance as InceptionV3 in detecting

Table 4: Class-wise classification results of individual model

Method	Class	Precision	Recall	F1	Accuracy
PDCOVIDNet	COVID-19	95.45	91.30	93.33	**96.58**
	Normal	97.04	97.04	**97.04**	
	Viral Pnemunia	96.32	97.04	96.68	
VGG16	COVID-19	90.48	82.61	86.36	93.86
	Normal	93.43	94.81	94.12	
	Viral Pnemunia	94.81	94.81	94.81	
ResNet50	COVID-19	94.74	78.26	85.71	92.15
	Normal	87.42	**97.78**	92.31	
	Viral Pnemunia	**97.56**	88.89	93.02	
InceptionV3	COVID-19	83.83	86.96	85.11	93.51
	Normal	96.15	92.59	94.34	
	Viral Pnemunia	92.81	95.56	94.16	
DenseNet121	COVID-19	95.24	86.96	90.91	94.54
	Normal	95.52	94.81	95.17	
	Viral Pnemunia	93.48	95.56	94.51	

Table 5: Weighted average results of each model

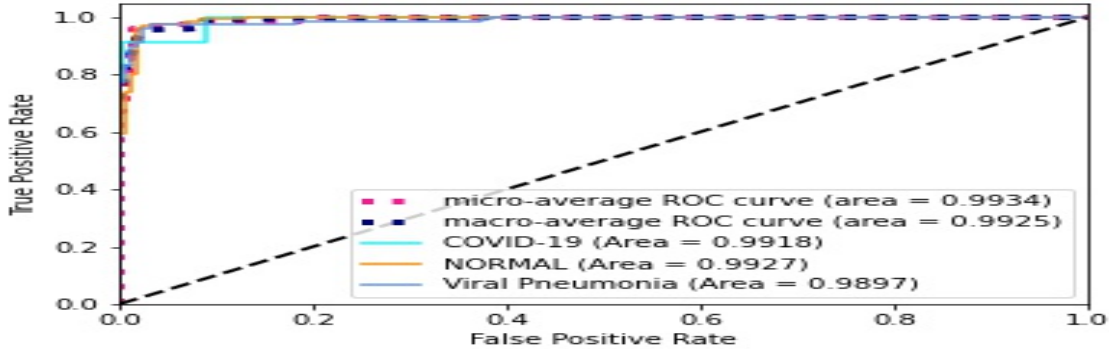| Method	Precision	Recall	F1
PDCOVIDNet	**96.58**	**96.59**	**96.58**
VGG16	93.84	93.86	93.83
ResNet50	92.67	92.15	92.12
InceptionV3	93.60	93.52	93.53
DenseNet121	94.54	94.54	94.53

Figure 4: Comparison of the ROC curve for COVID-19, Normal and Viral Pneumonia using PDCOVIDNet

COVID-19, and shows nearly the same performance with VGG16 in detecting viral pneumonia and normal cases. Finally, we can claim that PDCOVIDNet is powerful in detecting COVID-19 cases from chest X-ray images. For this reason, we believe that the proposed model focuses on discriminating features that can help distinguish between other types (e.g., normal and viral pneumonia).

6 Visualization using Grad-CAM and Grad-CAM++

In our evaluation, we used Grad-CAM and Grad-CAM++ visualization methods to visually represent the salient regions where PDCOVIDNet insisted on making the final classification decision on chest X-ray images. Accurate and decisive salient region detection is important for the interpretation of classification, while also ensuring the reliability of the results. In this regard, a two-dimensional heat map is generated from feature weights with different brightness, which corresponds to the importance of the feature. The heat map is overlaid on the input image to locate the salient region. Figure 6 shows the visualization results of Grad-CAM and Grad-CAM++ using PDCOVIDNet to locate salient regions
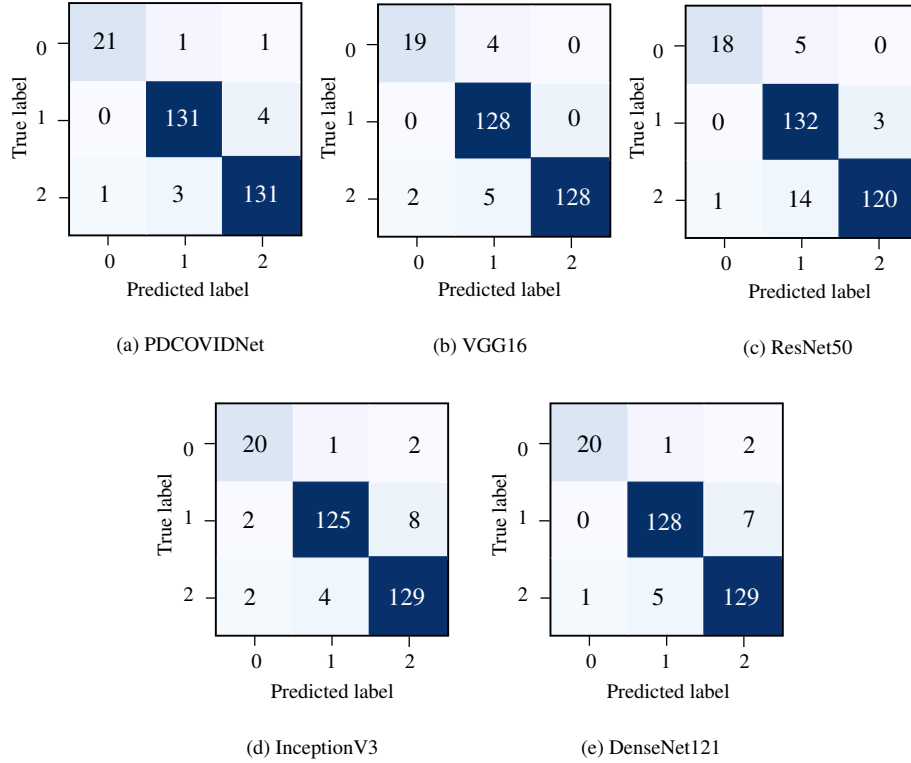
Figure 5: Confusion matrix of all evaluated models with test set. Classes 0, 1, and 2 represent COVID-19, Normal, and Viral Pneumonia.

when the input image is classified as COVID-19 or normal or viral pneumonia, where the regions distinguishing the classes in the lung have been localized. For COVID 19, both Grad-CAM and Grad-CAM++ generate seemingly the same results, so for the detection of critical areas, the overlapping positions of the heat maps can be considered. In the case of viral pneumonia, the salient regions detected using Grad-CAM and Grad-CAM++ are undifferentiated, while under normal class there are differences, and seems to fail to detect the salient regions as the heat map highlights outside X-ray than inside the lung. To help AI-based systems, it is certainly effective to provide the system with some human-understandable numerical measures (such as probability) as shown in Figure 6.

7 Investigation on the incorrect classification

In this section, we will further investigate the incorrect classification caused by the use of PDCOVIDNet. The total number of incorrectly classified images is 10, as shown in Figure 7. Two COVID-19 images are classified as normal and viral pneumonia, and in both cases, COVID-19 is far behind the prediction as the probability of COVID-19 prediction is very low compared with others. Normal images are not classified as COVID-19, but among the four images with normal classification errors, one prediction is on the edge of viral pneumonia, while the other predictions are very different. Correspondingly, among the four incorrectly classified images of viral pneumonia, one image belongs to COVID-19 and the other images are normal, and one of the predictions is very close.

8 Conclusion and Future Work

In this paper, we proposed a CNN-based method, called PDCOVIDNet, for detecting COVID-19 from chest X-ray images. As we have seen, PDCOVIDNet can effectively capture COVID-19 features by dilated convolution in the parallel stack of convolution blocks, so it has an excellent classification performance compared to some well-known CNN architectures. The dataset used in the experiment has a limited number of COVID-19 images, and at once, it is still developing, but data augmentation techniques have able to surmount the challenge as CNN based architecture needs more data for effective training. Our experimental evaluation shows that PDCOVIDNet outperforms the state-of-the-art
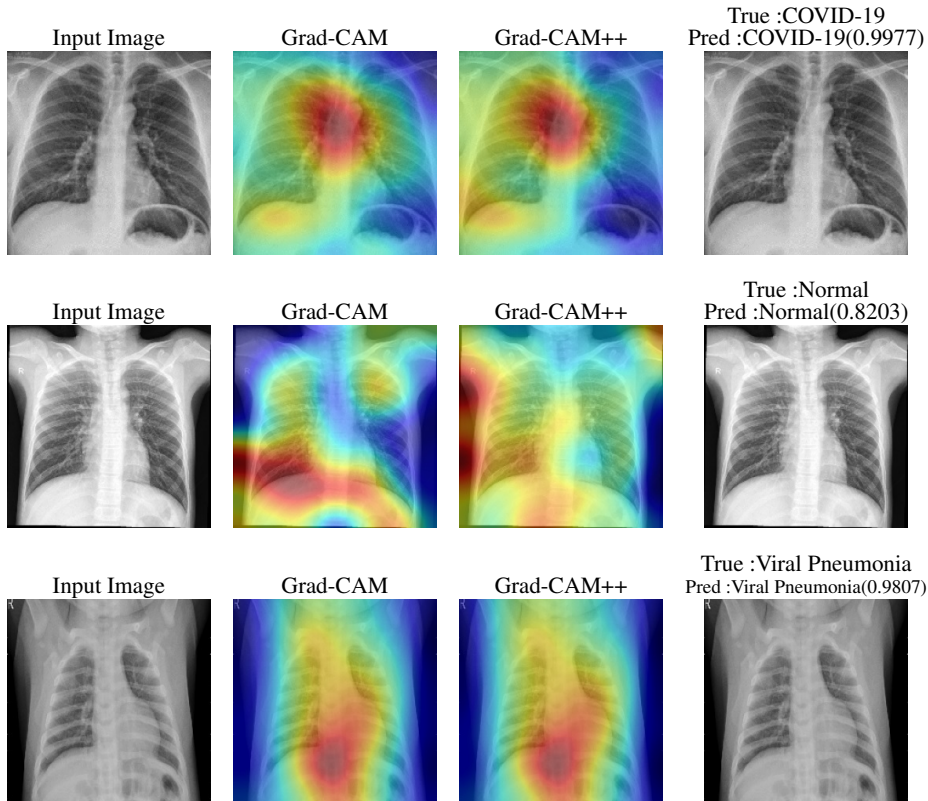
Figure 6: Input images and their Grad-CAM, Grad-CAM++, and human-understandable prediction with probability score according to the PDCOVIDNet. The first row provides input with COVID-19, as well as two types of visualization effects, and finally the prediction results.The second and third rows show the investigation of normal and viral pneumonia, respectively. Here, True means the actual class of the image, and Pred. means the predicted class.

models, with its precision and recall are 95.45% and 91.3%, respectively. As well, PDCOVIDNet demonstrates its potential through other performance metrics such as the weighted average of precision, recall and F1 scores, and finally the overall model accuracy. We apply the proposed model as well as two visualization techniques to identify the class-discriminative regions because they have a greater influence in classifying the input chest X-ray image into its anticipated classes. Finally, we believe that the current findings will hopefully overcome intellectual challenges to detect more cases of COVID-19 and use them to screen for COVID-19 cases in AI-based systems, especially in clinical practice.

As future work, we will explore and incorporate a diversified data set with more COVID-19 cases to make our proposed model more robust.

References

[1] World Health Organization. Covid-2019 situation reports.

[2] World Health Organization. Covid-19 pandemic.

[3] L Wang and A Wong. COVID-Net: A tailored deep convolutional neural network design for detection of covid-19 cases from chest x-ray images. *arXiv:2003.09871*, 2020.

[4] C Zheng, X Deng, Q Fu, Q Zhou, J Feng, H Ma, W Liu, and X Wang. Deep learning-based detection for covid-19 from chest ct using weak label. *medRXiv:10.1101/2020.03.12.20027185*, 2020.

[5] P Wikramaratna, R Paton, M Ghafari, and J lourençp. Estimating false-negative detection rate of sars-cov-2 by rt-pcr. *medRXiv:10.1101/2020.04.05.20053355*, 2020.

[6] M Mg, E Lee, J Yang, F Yang, X Li, H Wang, M Lui, C Lo, B S T Leung, P Khong, C Hui, K Yuen, and M Kuo. Imaging profile of the covid-19 infection: Radiologic findings and literature review. *Radiology: Cardiothoracic Imaging*, 2020. DOI:10.1148/ryct.2020200034.

[7] BBC. Bbc business reports.

[8] A Narin, K Ceren, and P Ziynet. Automatic detection of coronavirus disease (covid-19) using x-ray images and deep convolutional neural networks. *arXiv:2003.10849*, 2020.

[9] S Rajaraman, J Siegelman, O Alderson, P, S Folio, L, R Folio, L, and K Antani, S. Iteratively Pruned Deep Learning Ensembles for COVID-19 Detection in Chest X-rays. *arXiv:2004.08379*, 2020.

[10] D Apostolopoulos, I and A Mpesiana, T. Covid-19: automatic detection from x-ray images utilizing transfer learning with convolutional neural networks. *Physical and Engineering Sciences in Medicine*, 2020. DOI:10.1007/s13246-020-00865-4.

[11] K Md, R, D Till, R-S Dietrich, D Stefan, C Michael, and B Oya. Deepcovidexplainer: Explainable covid-19 predictions based on chest x-ray images. *arXiv:2004.04582*, 2020.

[12] Fisher Yu and Vladlen Koltun. Multi-scale context aggregation by dilated convolutions. In *International Conference on Learning Representations (ICLR)*, 2016.

[13] Muhammad E. H. Chowdhury, Tawsifur Rahman, Amith Khandakar, Rashid Mazhar, Muhammad Abdul Kadir, Zaid Bin Mahbub, Khandaker Reajul Islam, Muhammad Salman Khan, Atif Iqbal, Nasser Al-Emadi, and Mamun Bin Ibne Reaz. Can ai help in screening viral and covid-19 pneumonia? *arXiv:2003.13145*, 2020.

[14] SIRM. Covid-19 database.

[15] C Joseph, P, M Paul, and D Lan. Covid-19 image data collection. *arXiv:2003.11597*, 2020.

[16] P. Mooney. Chest x-ray images (pneumonia).

[17] 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 *IEEE International Conference on Computer Vision (ICCV)*, pages 618–626, 2017.

[18] A. Chattopadhay, A. Sarkar, P. Howlader, and V. N. Balasubramanian. Grad-CAM++: Generalized gradient-based visual explanations for deep convolutional networks. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 839–847, 2018.

[19] Joseph Bullock, Alexandra Luccioni, Katherine Hoffmann Pham, Cynthia Sin Nga Lam, and Miguel Luengo-Oroz. Mapping the landscape of artificial intelligence applications against covid-19. *arXiv:2003.11336*, 2020.

[20] Halgurd S. Maghdid, Aras T. Asaad, Kayhan Zrar Ghafoor, Ali Safaa Sadiq, and Muhammad Khurram Khan. Diagnosing covid-19 pneumonia from x-ray and ct images using deep learning and transfer learning algorithms. *arXiv:2004.00038*, 2020.

[21] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. In *Proceedings of the 25th International Conference on Neural Information Processing Systems*, volume 1, page 1097–1105, 2012.

[22] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[23] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and ZB Wojna. Rethinking the inception architecture for computer vision. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[24] Shuai Wang, Bo Kang, Jinlu Ma, Xianjun Zeng, Mingming Xiao, Jia Guo, Mengjiao Cai, Jingyi Yang, Yaodong Li, Xiangfei Meng, and Bo Xu. A deep learning algorithm using ct images to screen for corona virus disease (covid-19). *medRxiv 2020.02.14.20023028*, 2020. DOI:10.1101/2020.02.14.20023028.

[25] Biraja Ghoshal and Allan Tucker. Estimating uncertainty and interpretability in deep learning for coronavirus (covid-19) detection. *arXiv:2003.10769*, 2020.

[26] Syed Usama Khalid Bukhari, Syed Safwan Khalid Bukhari, Asmara Syed, and Syed Sajid Hussain Shah. The diagnostic evaluation of convolutional neural network (cnn) for the assessment of chest x-ray of patients infected with covid-19. *medRxiv 2020.03.26.20044610*, 2020.

[27] Shervin Minaee, Rahele Kafieh, Milan Sonka, Shakib Yazdani, and Ghazaleh Jamalipour Soufi. Deep-covid: Predicting covid-19 from chest x-ray images using deep transfer learning. *arXiv:2004.09363*, 2020.

[28] Forrest N. Iandola, Song Han, Matthew W. Moskewicz, Khalid Ashraf, William J. Dally, and Kurt Keutzer. Squeezenet: Alexnet-level accuracy with 50x fewer parameters and <0.5mb model size. *arXiv:1602.07360*, 2016.

[29] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q. Weinberger. Densely connected convolutional networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4700–4708, 2017.

[30] Mingxing Tan and Quoc V. Le. Efficientnet: Rethinking model scaling for convolutional neural networks. *arXiv:1905.11946*, 2019.

[31] Eduardo Luz, Pedro Lopes Silva, Rodrigo Silva, Ludmila Silva, Gladston Moreira, and David Menotti. Towards an effective and efficient deep learning model for covid-19 patterns detection in x-ray images. *arXiv:2004.05717*, 2020.

[32] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv:1409.1556*, 2014.

[33] Muhammad Farooq and Abdul Hafeez. Covid-resnet: A deep learning framework for screening of covid19 from radiographs. *arXiv:2003.14395*, 2020.

[34] Tulin Ozturk, Muhammed Talo, Eylul Azra Yildirim, Ulas Baran Baloglu, Ozal Yildirim, and U. Rajendra Acharyaf. Automated detection of covid-19 cases using deep neural networks with x-ray images. *Computers in Biology and Medicine 121 (2020) 103792*, 2020. DOI:10.1016/j.compbiomed.2020.103792.

[35] Vincent Dumoulin and Francesco Visin. A guide to convolution arithmetic for deep learning. *arXiv:1603.07285*, 2016.

[36] Vinod Nair and Geoffrey E. Hinton. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th International Conference on International Conference on Machine Learning*, pages 807–814, 2010.

[37] Dominik Scherer, Andreas Müller, and Sven Behnke. Evaluation of pooling operations in convolutional architectures for object recognition. In *20th International Conference on Artificial Neural Networks (ICANN)*, pages 92–101, 2010.

[38] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[39] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition,*, 2016.

[40] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv:1412.6980*, 2014.

Pred :Normal(0.54)
True :Viral Pneumonia(0.45)

Pred :Viral Pneumonia(0.60)
True :Normal(0.39)

Pred :Viral Pneumonia(0.88)
True :Normal(0.11)

Pred :Viral Pneumonia(0.93)
True :Normal(0.07)

Pred :Normal(0.93)
True :Viral Pneumonia(0.06)

Pred :Viral Pneumonia(0.96)
True :Normal(0.04)

Pred :Normal(0.95)
True :COVID-19(0.01)

Pred :Normal(0.98)
True :Viral Pneumonia(0.02)

Pred :COVID-19(0.98)
True :Viral Pneumonia(0.02)

Pred :Viral Pneumonia(0.98)
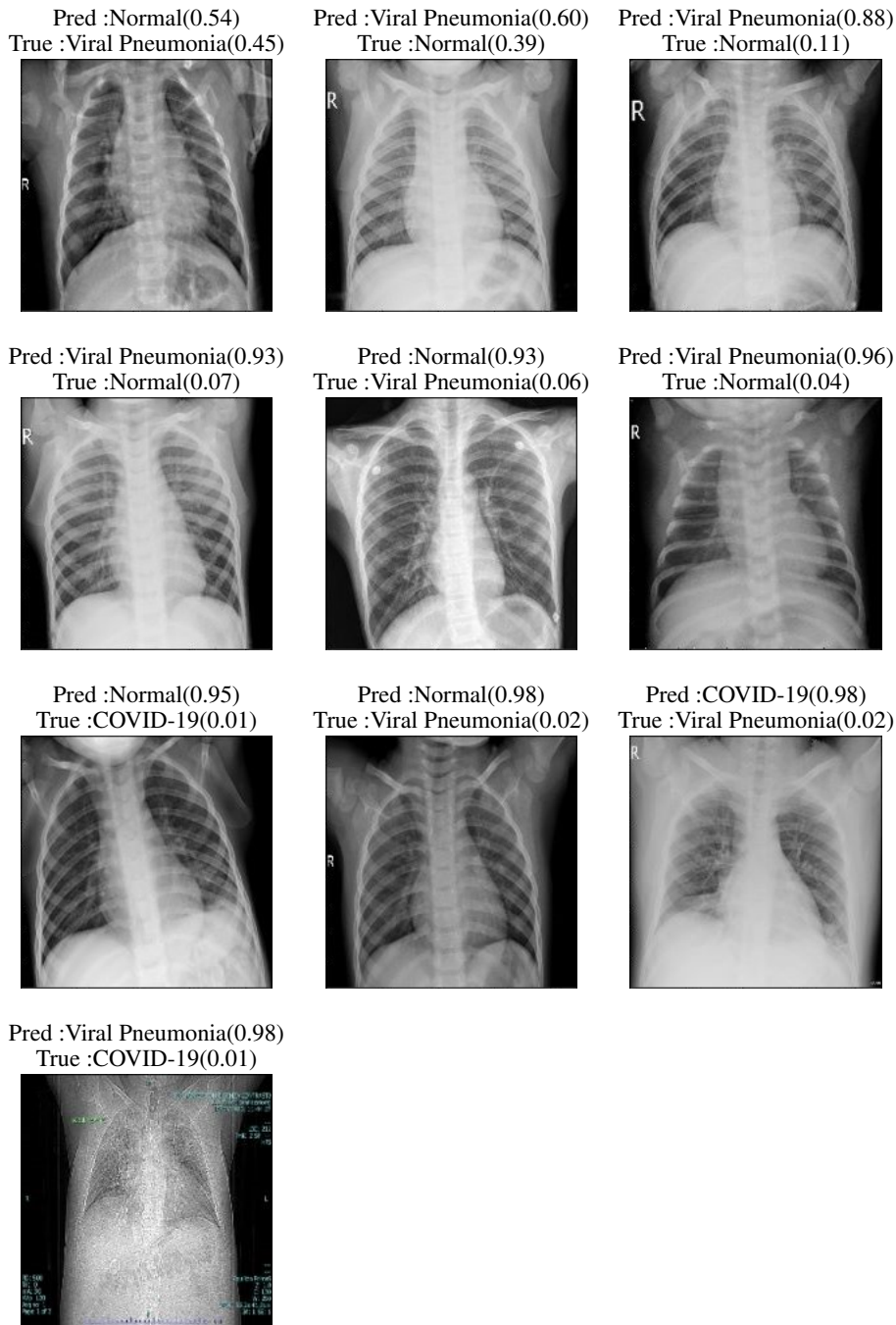True :COVID-19(0.01)

Figure 7: Investigation on the incorrectly classified images with the probability of predicted class and actual class