# A Transfer Learning Method for Covid-19 and Pneumonia Diagnosis Based on Chest Radiograph Classification

Yuxuan Sun
*School of Computer Science and Technology*
*Harbin Institute of Technology*
Weihai, China
yuxuan_eric_sun@outlook.com

Linxu Guo
*School of Ocean Engineering*
*Harbin Institute of Technology*
Weihai, China
linxuguo@gmail.com

Teik-Toe Teoh
*NTU Business AI Lab*
*Nanyang Technological University*
Singapore
ttteoh@ntu.edu.sg

2023 15th International Conference on Computer Research and Development (ICCRD) | 978-1-6654-8750-4/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICCRD56364.2023.10080197

***Abstract*—Pneumonia has been a tough and dangerous human illness for a history-long time, notably since the COVID-19 pandemic outbreak. Many pathogens, including bacteria or viruses like COVID-19, can cause pneumonia, leading to inflammation in patients' alveoli. A corresponding symptom is the appearance of lung opacities, which are vague white clouds in the lungs' darkness in chest radiographs. Modern medicine has indicated that pneumonia-associated opacities are distinguishable and can be seen as fine-grained labels, which make it possible to use deep learning to classify chest radiographs as a supplementary aid for disease diagnosis and performing pre-screening. However, deep learning-based medical imaging solutions, including convolutional neural networks, often encounter a performance bottleneck when encountering a new disease due to the dataset's limited size or class imbalance. This study proposes a deep learning-based approach using transfer learning and weighted loss to overcome this problem. The contributions of it are three-fold. First, we propose an image classification model based on pre-trained Densely Connected Convolutional Networks using Weighted Cross Entropy. Second, we test the effect of masking non-lung regions on the classification performance of chest radiographs. Finally, we summarize a generic practical paradigm for medical image classification based on transfer learning. Using our method, we demonstrate that pre-training on the COVID-19 dataset effectively improves the model's performance on the non-COVID Pneumonia dataset. Overall, the proposed model achieves excellent performance with 95.75% testing accuracy on a multiclass classification for the COVID-19 dataset and 98.29% on a binary classification for the Pneumonia dataset.**

***Keywords—Artificial Intelligence, Deep Learning, Transfer Learning, Image Processing, Pneumonia diagnosis***

## I. INTRODUCTION

The World Health Organization (WHO) estimates that pneumonia kills about 2 million young children under five every year, which has been the main cause of children It has been the main cause of child mortality [1]. Mcluckie [2] also suggested that bacterial and viral infections are the two primary culprits. More specifically, viral pneumonia has caused extreme concern due to the COVID-19 epidemic in recent years, which is responsible for over 6.5 million deaths [3]. As shown in the work of Kermany, Daniel S., et al. [4], the different lung opacity patterns caused by different pneumonia types make it possible to use neural networks for disease diagnosis, i.e., image classification. Furthermore, it is intuitive to hypothesize that using lung-area images (excluding the non-lung area) as input can offer sufficient feature information for the classifier.

When applying image-based deep learning methods to medical images, an effective and useful technique for it is transfer learning, especially for tasks with limited data due to privacy restrictions or lack of expert ground truth [5]-[7]. Generally speaking, instead of training a new network from scratch, it is better to pre-train a Convolutional Neural Network (CNN or ConvNet) on a large dataset in other domains like ImageNet. Additionally, the greater the similarity between the original task and the new task, the greater the transferability gap, but even features pre-trained on distant tasks are superior to random weight, according to Yosinski, Jason, et al. [7]. This conclusion has been proved in [4], where weights transferred from ImageNet help the model achieve better performance on the pneumonia classification task. Therefore, theoretically speaking, as COVID-19 lung radiographs are comparable to those of other viral pneumonia, a network pre-trained on ImageNet and the COVID dataset could achieve similar accuracy on the (non-COVID) pneumonia dataset.

Another factor that affects the performance of CNN is how we deal with the class imbalance. Uneven distribution of data categories is common in medical imaging tasks, which is likely to hinder the performance of classifiers [8]. The common strategies to overcome it include data augmentation [9] (e.g., image manipulations and GAN-based methods) and weighted loss function [10]. It is tricky to augment the dataset without introducing a new domain gap so that the weighted loss function could be more generic and time-saving.

In this study, in order to address the above three topics and related problems, we conduct experiments on two related datasets: the COVID-19 Radiography Database [11], [12] (COVID dataset) and Chest X-Ray Images (Pneumonia dataset) [13]. Both of them are chest x-ray images, while the former consists of 4 categories of images (3616 COVID-19 positive cases along with 10,192 Normal, 6012 Lung Opacity, and 1345 Viral Pneumonia images) and corresponding ground-truth lung

Dr. Teoh Teik Toe, the senior lecturer and academic director of the Business AI Lab at Nanyang Technological University, supported this study. (Correspondence: ttteoh@ntu.edu.sg; Tel.: 65-97905202).

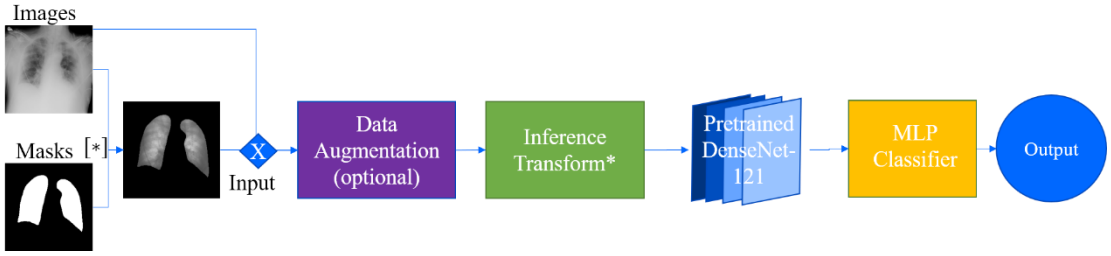


Fig. 1. Transfer Learning Paradigm (A modified version of Figure. 1 in [4]. In this study, we omit the part of data augmentation to emphasize the effect of Weighted Loss Function. * DenseNet121_Weights.IMAGENET1K_V1.transforms from torchvision package is used as the Inference Transform.)

masks. The latter dataset consists of 2 categories of images (4273 positive pneumonia cases (including Bacterial and viral) along with 1583 Normal) without masks.

**Contributions.** Our key contributions are listed as follows:

- We propose an image classification model based on pre-trained DenseNet-121 using Weighted Cross Entropy, which is fine-tuned on the COVID dataset and achieves excellent performance with a test accuracy of 95.75%. Fig. 1 presents an overview of our model, which takes chest radiographs (raw images or masked images) as inputs and predicts the corresponding labels.
- We tested the effect of masking non-lung regions on the classification performance of chest radiographs and find that using raw images as inputs get better performance.
- We have summarized a generic practical paradigm for medical image classification based on transfer learning. Using our method, we demonstrate that pre-training on the COVID dataset effectively improves the classifier performance on the (non-COVID) Pneumonia dataset (both as a fixed feature extractor or a pre-trained model to fine-tune). Overall, the pre-trained model, which was fine-tuned on the pneumonia dataset, can achieve a testing accuracy of 98.06%.

## II. Related Work

### A. AI screening Viral and COVID-19 Pneumonia

In [11], AI (Image classifiers based on CNN) was used to detect COVID-19-related pneumonia from chest X-ray images. Those authors created a public database consisting of 423 COVID-19, 1485 viral pneumonia, and 1579 normal chest X-ray images. The transfer learning method with data augmentation was used to train several pre-trained CNNs, which achieved 99.7% accuracy. This research has shown the feasibility of pneumonia diagnosis based on X-ray image classification. Also, DenseNet-201 (with data augmentation) using transfer learning techniques achieved its best score, which inspired us to design a similar architecture for our comparable tasks.

### B. Transfer Learning

As mentioned in [4], [7], [11], transfer learning is the core technique for most classification tasks. There are three main scenarios of Transfer Learning, including using CNN as a fixed feature extractor, fine-tuning the CNN and using pre-trained models. PyTorch Image Models (timm) [14] is a PyTorch package built by Ross Wightman, which is a collection of image models, pre-trained weights, and other useful tools. In this paper, our image models use models from timm, which were pre-trained on ImageNet. Moreover, when dealing with a medical imaging task, datasets consisting of grayscale images (e.g., radiographs) are quite common, which means that inputs are training images with a single channel instead of RGB images with 3 channels. However, most pre-trained models were trained on colorful datasets (e.g., ImageNet). Previous studies [15] have shown that using color or grayscale images is a factor that affects model performance. In this study, we tested and compared the performance disparities between two naive methods without changing the weights pre-trained on color images.

### C. Dense Convolutional Network

As ResNet significantly changed the parameterization of deep network functions, the Dense convolutional Network (DenseNet) is, in a sense, its logical extension [16]. Composed of dense blocks (the modified "batch normalization, activation, and convolution" structure used by ResNet) and transition layers, DenseNet can be considerably deeper, more expeditious and more accurate to train because of the shorter connections it contains between layers close to the input and those adjacent to the output. In [11], DenseNet-201, namely the DenseNet with 201 layers, is used to acquire the best score. However, it is possible and more accessible for an uncomplicated classification task to train a DenseNet with fewer layers for real application. Hence a DenseNet-121 is chosen as the baseline after fundamental experiments. Combining previous work [17] with our experiments, we designed the baseline model using Adam with Nesterov Momentum (NAdam) as an optimizer with determined hyper-parameters.

### D. Weighted Cross Entropy Loss

The original Cross-entropy loss function for image classification cannot focus on the frequency of classes during training and thus results in potentially lower accuracy on datasets with class-imbalance problems [10]. A straightforward solution is using weighted loss functions (Weighted Cross Entropy, WCE), including Balanced Cross Entropy, Inverse Class Frequency, and Focal Loss [18]. Substantial research has proven their effectiveness qualities for image classification [19].

## III. Methodology

### A. Generic Practical Paradigm

According to our experiments, we have summarized a generic practical paradigm to solve medical image classification tasks, which also has been formalized with mathematics.

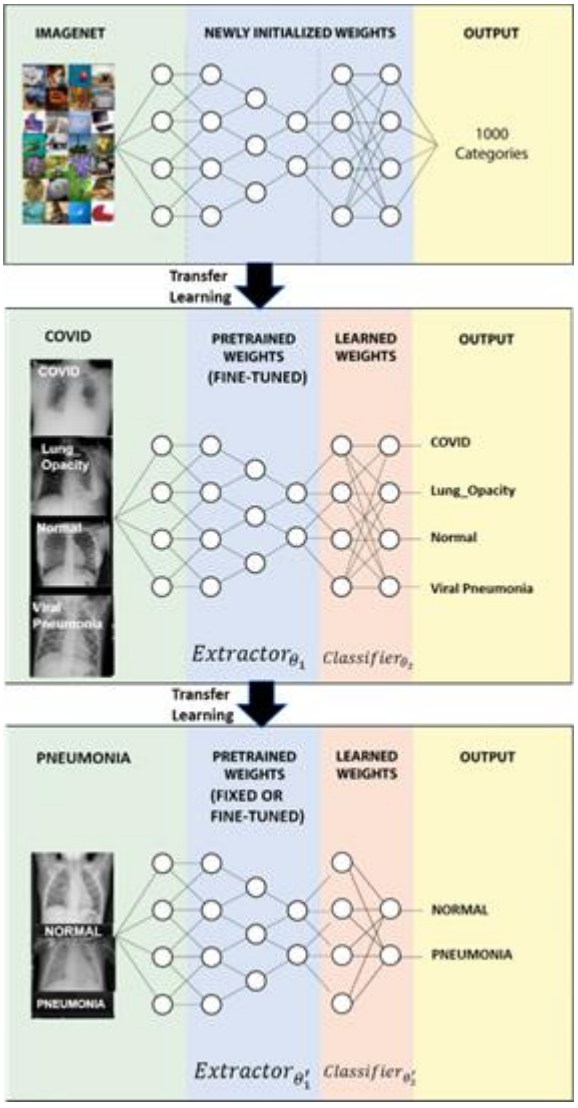


Fig. 2. Transfer Learning Paradigm (*a modified version of [4, Fig. 1]*)

For instance, take one example as input with the shape of $h \times w \times c$, where $h$ denotes height, $w$ denotes width, and $c$ denotes channel. $\mathcal{H}$ denotes the hypothesis class, namely the proposed classification model.

The overall problem is to devise such a $\mathcal{H}$ to achieve a map as (1):

$$\mathcal{H}: R^{h \times w \times c} \to R^{k} \tag{1}$$

where $k$ denotes the number of image categories.

More specifically, using $\mathcal{H}^{(1)}$ denotes the transferred network, and $\mathcal{H}^{(2)}$ denotes the target task, we can represent those two processes as follows:

$$\mathcal{H}^{(1)}: R^{h_1 \times w_1 \times c_1} \to R^{c_1^{'} \times h_1^{'} \times w_1^{'}} \to R^{k_1} \tag{2}$$

$$\mathcal{H}^{(2)}: R^{h_2 \times w_2 \times c_2} \to R^{c_2^{'} \times h_2^{'} \times w_2^{'}} \to R^{k_2} \tag{3}$$

Afterward, if we roughly divide the model into two ingredients: the first one is the feature extractor (e.g., in our model, it is the DenseNet, in which its global average pool and the fully-connected layer have been removed), and the other one is the classification layer (e.g., in our model, it consists of a global average pool along with an MLP)

$$\mathcal{H}_{\theta}^{(1)}(X) = Classifier_{\theta_2}\left(Extractor_{\theta_1}(X)\right) \tag{4}$$

$$\mathcal{H}_{\theta}^{(2)}(X) = Classifier_{\theta_2^{'}}\left(Extractor_{\theta_1^{'}}(X)\right) \tag{5}$$

where:

$$Extractor: R^{h \times w \times c} \to R^{n} \tag{6}$$

$$Classifier: R^{n} \to R^{k_i} \tag{7}$$

In our proposed model, $X$ denotes a general tensor, and its indexing mechanism (e.g., $x_{ijk}$ and $[X]_{1,2i-1,3}$) is a natural extension of matrix indexing. Each image is delivered as a $3^{rd}$-order tensor whose axes correspond to its height, width, and channel. However, in real coding, the input is not a single $3^{rd}$-order tensor but a $4^{th}$-order tensor because of using Mini-Batch, where separate photos are indexed along the first axis. The number of channels is whether 1 or 3. $w_1^{'} = h_1^{'} = w_2^{'} = h_2^{'} = 224$, $k_1 = 4$, $k_2 = 2$. The process of pre-training on ImageNet can be absorbed in (2).

While Fig. 2 shows our paradigm briefly with visualization, the whole idea of transfer learning can also be generalized as follows:

*1) Pre-training*

Training $\theta_1$ on the larger and more fine-grained datasets (e.g on ImageNet firstly and COVID dataset secondly in this study).

*2) Initialization*

Let $\theta_1^{'} := \theta_1$ as the parameter initialization for $\theta_1^{'}$.

*3) Optimization*

$$\text{minimize}_{\theta} \frac{1}{m} \sum_{i=1}^{m} l_{ce}\left(h_{\theta}(X), y^{(i)}\right) \tag{8}$$

## B. Dataset Description

Two datasets are chosen for our experiments, which both consist of chest radiographs. The different patterns of lung opacity existing in different types of images make it possible to classify [4]. In addition, there is an obvious feature of them in common should be emphasized. Both of them contain class-imbalance problems.

*1) COVID-19 Radiography Database*

COVID-19 RADIOGRAPHY DATABASE (COVID dataset) [11], [12] consists of 3616 photos labelled as COVID photos, 6012 as Lung_Opacity, 10192 as Normal, and 1345 as Viral Pneumonia with a significant class imbalance in the dataset. As a so-called COVID dataset, only about 17% of its images are labeled COVID. Despite it, the advantage of this

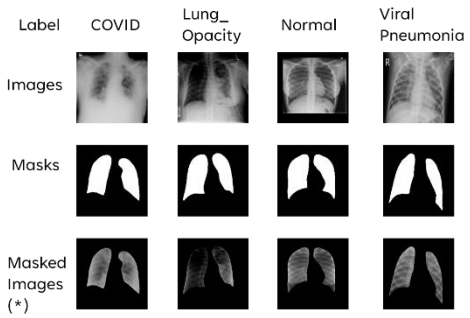


Fig. 3. Some samples of COVID Dataset and corresponding masked images (* *Masked images are generated with method in*

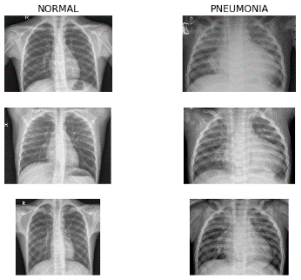


Fig. 4. Some samples of Pneumonia Dataset

dataset is that each image has a corresponding lung mask, which professional doctors have labeled.

Our first experiment tested the classifier's performance using masked images (without non-lung details) as inputs. Each mask image consists of white pixels (lung area) and black pixels (non-lung area). To generate the masked images, we adjust the values of the white and black areas in masks to 0 and 1, respectively, and apply elementwise multiplying for images and masks after resizing them correctly. The operation can be presented as (9):

$$\mathbf{C} := \mathbf{A} \odot \frac{1}{255}\mathbf{B} = \frac{1}{255}\begin{bmatrix} a_{11} \times \max\{0, b_{11}\} & a_{12} \times \max\{0, b_{12}\} & \dots & a_{1n} \times \max\{0, b_{1n}\} \\ a_{21} \times \max\{0, b_{21}\} & a_{22} \times \max\{0, b_{22}\} & \dots & a_{2n} \times \max\{0, b_{2n}\} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} \times \max\{0, b_{m1}\} & a_{m2} \times \max\{0, b_{m2}\} & \dots & a_{mn} \times \max\{0, b_{mn}\} \end{bmatrix} \quad (9)$$

where C stands for the masked image, A and B for its raw image and mask, respectively. Fig. 3 shows some samples of the COVID dataset.

We reshuffle all images and divide them in the ratio of 7:1.5:1.5 into training, validation, and test data sets.

*2) Chest X-Ray Images*

As shown in Fig. 4, these chest radiographs (Pneumonia Dataset) were chosen from a group of patients at the Guangzhou Women and Children's Medical Center [13]. The dataset includes 1583 Normal images and 4273 Pneumonia images with a class imbalance.

Note that the images labeled as PNEUMONIA contain both viral and bacterial pneumonia, which means that this study uses a coarse-grained binary classification on this dataset.

We reshuffle all images and divide them into training, validation, and test sets in the ratio of 7:2:1.

The model fine-tuned on the COVID dataset will be transferred to this dataset as a fixed feature extractor (or continue to fine-tune) to train a binary classifier.

### C. CNN Model Selection

According to works in [11], [17], DenseNet has been proven effective in COVID-19 detection tasks based on chest radiographs. In addition, considering the trade-off between model accuracy and computational scale, we finally chose DenseNet-121 as the baseline.

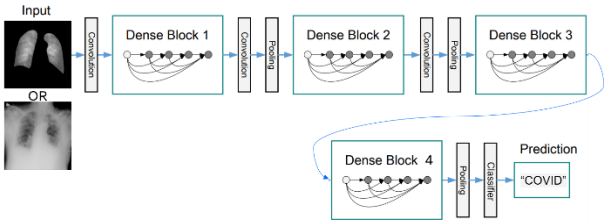


Fig. 5. How Dense Blocks makes up DenseNet [16, Fig.2]

TABLE I. PROPOSED MODEL ARCHITECTURE (A MODIFIED VERSION OF [16, TABLE. 1])

| Layers | Output Size | DenseNet-121 |
|---|---|---|
| Convolution | 112 × 112 | 7 × 7 conv, stride 2 |
| Polling | 56 × 56 | 3 × 3 max pool, stride 2 |
| Dense Block (1) | 56 × 56 | $\begin{bmatrix} 1 \times 1\ conv \\ 3 \times 3\ conv \end{bmatrix} \times 6$ |
| Transition Layer (1) | 56 × 56 | 1 × 1 conv |
| | 28 × 28 | 2 × 2 average pool, stride 2 |
| Dense Block (2) | 28 × 28 | $\begin{bmatrix} 1 \times 1\ conv \\ 3 \times 3\ conv \end{bmatrix} \times 12$ |
| Transition Layer (2) | 28 × 28 | 1 × 1 conv |
| | 14 × 14 | 2 × 2 average pool, stride 2 |
| Dense Block (3) | 14 × 14 | $\begin{bmatrix} 1 \times 1\ conv \\ 3 \times 3\ conv \end{bmatrix} \times 24$ |
| Transition Layer (3) | 14 × 14 | 1 × 1 conv |
| | 7 × 7 | 2 × 2 average pool, stride 2 |
| Dense Block (4) | 7 × 7 | $\begin{bmatrix} 1 \times 1\ conv \\ 3 \times 3\ conv \end{bmatrix} \times 16$ |
| Classification Layer | 1 × 1 | 7 × 7 global average pool |
| | | 64D fully-connected, Mish |
| | | 4D fully-connected, Softmax |

Fig. 5 illustrates how a DenseNet is composed of multiple Dense Blocks, while a Dense Block is made up of several convolution blocks, all of which have the same number of output channels. [16]. In its PyTorch implementation [14], each Dense Block consists of a Conv2d layer and a BatchNormAct2d layer. DenseNet is distinguished by the connection pattern in which each layer connects to all its preceding levels and the concatenation operation (rather than the addition operator in ResNet) to preserve and reuse information from previous layers.

While the details of the principle and implementation of DenseNet can be found in [16], we use the model as the feature extractor and modify the last Classification Layer, where we remain the global average pool but replace the 1000D fully-connected layer with a multilayer perceptron that consists of a 64D fully-connected layer, a Mish [20] as the activation function and a 4D fully-connected layer (or 2D fully-connected layer for the pneumonia dataset). The DenseNet-121 architecture we modified for the COVID dataset is shown in Table I.

In our model, the number of input channels in the first convolutional layer is either 1 or 3, depending on the processing method used in "D. Pre-processing", i.e., the number of channels in input images.

### D. Pre-processing

Firstly, we need to consider how we deal with the channel of input grayscale images. As we have already mentioned, most (including the one we used) pre-trained models are trained on color datasets, which means that the underlying features they acquired are associated with RGB channels [15], and the number of input channels is 3. However, our two datasets consist of grayscale images, which only include one channel. In the subsequent experiments, we tested two methods separately:

#### 1) Pseudo-RGB

Copy the single channel data of each image into three channels to make up a pseudo-RGB image, and the original model is directly used for training.

#### 2) Single-Channel

Change the number of input channels of the first convolutional layer in the model (parameter "in_channels" of the first "torch.nn.Conv2d") to 1, and single-channel images are used as inputs.

Afterward, we apply the inference transforms to input images, which perform the following operations as our preprocessing: Accepts batched images. All images are resized to $256 \times 256$ using bilinear interpolation, followed by a central crop which converts them into $224 \times 224$. Finally, the values are rescaled to [0.0, 1.0] and then normalized with mean values of [0.485, 0.456, 0.406] and standard deviation values of [0.229, 0.224, 0.225] (in method 2, they are [0.485] and [0.229], respectively).

### E. Weighted Loss Function

Considering the existence of class imbalance in the dataset, we tested the effect of different weighted loss functions on improving classification accuracy:

#### 1) Cross Entropy

Given the n-class classification task (where Class = {0 ,1, …, n}, Cross Entropy loss of the object which belongs to class i can be calculated as follows:

$$l_{CE}(i) = \sum_1 i - t_i \log\big(P(i)\big) \tag{10}$$

In (10), $t_i$ is the corresponding one-hot vector indicating the label. $P(i)$ is a vector of logits.

#### 2) Weighted Cross Entropy

$$l_{WCE}(i) = \sum_1 i - wt_i \log\big(P(i)\big) \tag{11}$$

In (11), $w$ is a weight vector whose value can be chosen class by class. Next, we will introduce the weighting methods we chose for our experiments:

*a) Weighted Cross Entropy using 1-Class Frequency*

$$M = \sum_{i=1}^{n} N_i \tag{12}$$

$$w = 1 - \frac{N_i}{M} \tag{13}$$

In (12), (13), $N_i$ denotes the number of samples of the particular class i (in the training set). $M$ denotes the total number of the (training) dataset.

*b) Weighted Cross Entropy Using Inverse Class Frequency*

$$w = \frac{M}{N_i} \tag{14}$$

Symbols in (14) have same meaning in *a)*.

*c) Focal Loss*

$$w = \alpha\big(1 - P(i)\big)^{\gamma} \tag{15}$$

Focal Loss [18], [22] can also be considered as a kind of weighted cross entropy loss function. However, its weights are adjusted automatically rather than fixed.

By using the confidence of classifier $w = \alpha\big(1 - P(i)\big)^{\gamma}$ as weights, where $\gamma$ is a positive hyperparameter, the classifier is less confident in minority-class objects. In other words, lower value of $P(i)$ leads to the correspondingly larger weight $w$ and automatically bring focus to itself in the training process.

### F. Performance Evaluation

The performance of models was evaluated using four main classification metrics: accuracy, recall (sensitivity), precision (PPV), and F1 score presented in (16)-(19). Furthermore, those values of positive classes (COVID and Pneumonia) on two datasets are computed with the confusion matrix.

$$\text{Accuracy}_{\text{class}_i} = \frac{\text{TP}_{\text{class}_i} + \text{TN}_{\text{class}_i}}{\text{TP}_{\text{class}_i} + \text{TN}_{\text{class}_i} + \text{FP}_{\text{class}_i} + \text{FN}_{\text{class}_i}} \tag{16}$$

$$\text{Precision}_{\text{class}_i} = \frac{\text{TP}_{\text{class}_i}}{\text{TP}_{\text{class}_i} + \text{FP}_{\text{class}_i}} \tag{17}$$

$$\text{Sensitivity}_{\text{class}_i} = \frac{\text{TP}_{\text{class}_i}}{\text{TP}_{\text{class}_i} + \text{FN}_{\text{class}_i}} \tag{18}$$

$$\text{F1}_{\text{class}_i} = \frac{2 \times \text{Precision}_{\text{class}_i} \times \text{Recall}_{\text{class}_i}}{\text{Precision}_{\text{class}_i} + \text{Recall}_{\text{class}_i}} \tag{19}$$

For the multiclass classification on the COVID dataset, $class_i \in \{COVID,\ Lung\_Opacity,\ Normal,\ Viral\ Pneumonia\}$. For the binary classification on the Pneumonia dataset, $class_i \in \{Normal, Pneumonia\}$.

In real disease diagnosis, people tend to attach more importance to a certain positive class. For example, the recall of COVID is more important than the overall precision for pandemic prevention. We possibly would rather identify healthy samples or patients with viral pneumonia as COVID than miss the diagnosis. However, in this study, we use overall accuracy as the measure of model performance to simplify model tuning. In the training process, models are trained with 30 epochs, and a group of parameters with the highest accuracy on the validation set is selected as the final weights.

TABLE II. PERFORMANCE ON COVID CLASS USING DIFFERENT INPUTS AND PRE-PROCESSING

| Schemes | channels | Test COVID Precision | Test COVID Sensitivity (Recall) | Test COVID F1-Score |
|---|---|---|---|---|
| Raw Images | 1 | 97.08% | 96.57% | 96.83% |
| | 3 | **98.66%** | **97.36%** | **98.01%** |
| Masked Images | 1 | 93.20% | 85.81% | 89.89% |
| | 3 | 95.01% | 90.50% | 92.70% |

TABLE III. WEIGHTED AVERAGE PERFORMANCE USING DIFFERENT INPUTS AND PRE-PROCESSING

| Schemes | channels | Test Loss | Test Accuracy | Test Precision | Test Sensitivity (Recall) | Test F1-Score |
|---|---|---|---|---|---|---|
| Raw Images | 1 | 0.2685 | 95.32% | 95.61% | 96.03% | 95.81% |
| | 3 | 0.2296 | **95.75%** | **96.20%** | **96.71%** | **96.45%** |
| Masked Images | 1 | **0.2269** | 91.92% | 91.72% | 91.38% | 91.51% |
| | 3 | 0.2603 | 93.48% | 93.73% | 93.42% | 93.55% |

## IV. EXPERIMENTS AND RESULTS

**Common premise**: In the following experiments, we use NAdam [23] as the optimizer, MultiStepLR with milestones = [8, 15, 25] as the learning rate scheduler, and Cross Entropy Loss as the loss function for all validations and tests. We maintain the ratio of batch size to the learning rate to be 1.6e4. For example, when the batch size is set to 32, the corresponding learning rate is 2e-3.

### *A. Experiment 1. Masked Images as Inputs*

In this experiment, the batch size is set to 16, the learning rate is set to 2e-3, and weighted cross-entropy loss using 1-class frequency is used for training optimization, while cross-entropy loss without weights is used for validation and testing. All models are initialized with the same pre-trained weights. A total of four experimental groups were set up to test the difference in performance between the two kinds of input images (raw images and masked images) and the two types of image pre-processing (pseudo-RGB images and single-channel grayscale images).

Table. II shows the weighted average performance among all classes. Table. III shows the performance for the COVID images. The feature "channels" represents the corresponding pre-processing method ("1" denotes single-channel, "3" denotes pseudo-RGB). The results illustrate that using raw images with the non-lung area as inputs leads to a significantly better classification performance than using masked images without non-lung areas. Considering that, in theory, the non-lung area is supposed to be unhelpful for pneumonia diagnosis, these results are quite confusing. Whether this is the result of some medical facts or overfitting still needs further experiments and studies with the help of medical experts.

Additionally, using single-channel images as inputs leads to slightly lower accuracy, most likely the result of using parameters pre-trained on color datasets.

Since the lack of parameters pre-trained on grayscale ImageNet and the higher priority on accuracy over training speed in disease diagnosis, we decided to use pseudo-RGB (i.e., 3-channels) raw images as inputs for the following experiments.

### *B. Experiment 2. Comparing Different Weighted Loss Functions*

After experiment 1, we found that due to the limitation of the dataset size, 100% training accuracy can be achieved within 30 epochs using Cross Entropy alone. We believe that weighted loss functions may have little improvement in accuracy, and it is hard to say which one is better. Nevertheless, we believe that weighted loss can potentially reduce the time of model convergence. Therefore, we followed some hyperparameter settings during Experiment 1 and designed the experiments as shown in Table IV. In all 4 experiment groups, the batch size is set to 16, the learning rate is set to 2e-3, and pseudo-RGB raw images are used as inputs. All models were trained for 30 epochs, from which the group of parameters leading to the highest accuracy on the validation set was selected as the optimal parameters.

Table IV shows that using weighted cross-entropy loss can not only accelerate the training process but also improve classification accuracy

TABLE IV. PERFORMANCE USING DIFFERENT WEIGHTED LOSS

| | Cross Entropy Loss | $l_{WCE}$ using 1-CF | $l_{WCE}$ using Inverse CF | Focal Loss |
|---|---|---|---|---|
| Best Test Accuracy | 94.68% | **95.75%** | 95.13% | 94.51% |
| Epochs needed for training | 30 | 25 | 23 | **19** |

### C. *Experiment 3. Transfer Learning on Pneumonia Dataset*

According to the results of experiments 1 & 2, we suggest that the weighted cross entropy loss using 1-Class Frequency as weights is more suitable for this task. In the last experiment, we transferred the best model to the coarse-grained pneumonia dataset to test and compare its performance with other control groups. In this experiment, pseudo-RGB raw images are used as inputs, and the rest of the hyperparameter settings are the same as those in Experiment 2.

As shown in Table V and Table VI, within 30 epochs, our transfer learning method significantly improves the performance of the same model. More specifically, the fine-tuned model pre-trained on ImageNet and COVID has the best performance with the highest test accuracy and less training time needed compared with the same model without pre-training. Furthermore, using the model pretrained on ImageNet and COVID as the feature extractor also leads to better performance than its counterpart pretrained only on ImageNet.

Fig. 6 and Fig. 7 show the ROC curves of our best models on those two datasets. Fig. 8 shows the confusion matrices for corresponding tests.

Overall, our proposed model achieves excellent performance with a weighted average F1-score of 96.45% on the multiclass classification for the COVID-19 dataset and 98.29% on the binary classification for the Pneumonia dataset. Our method is accurate enough for both tasks, with a test precision of 98.66% for COVID and 98.30% for Pneumonia. Our related code is available in [24]

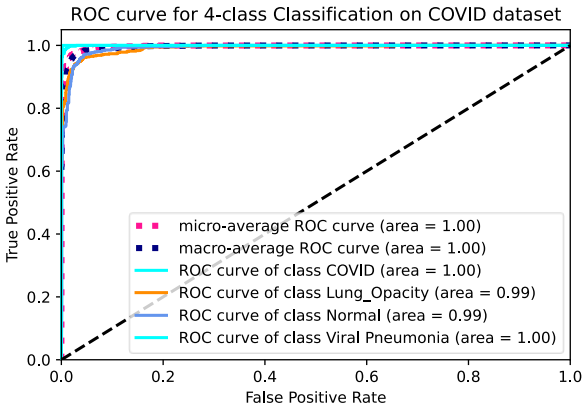


Fig. 6. ROC Curve for 4-Class Classification on COVID Dataset

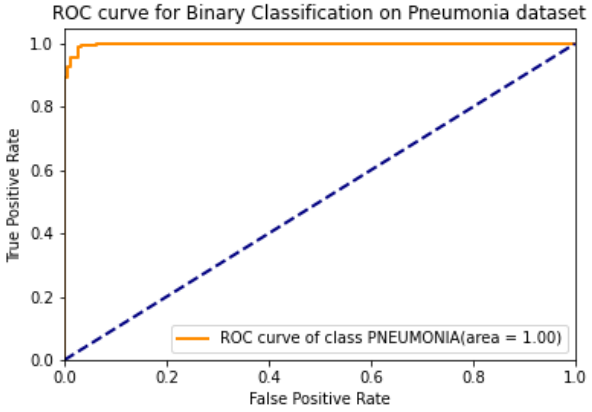


Fig. 7. ROC Curve for Binary Classification on Pneumonia Dataset

.

TABLE V. WEIGHTED AVERAGE PERFORMANCE ON PNEUMONIA DATASET

| Pre-trained Dataset | Schemes | Test Loss | Test Accuracy | Test WA precision | Test WA recall | Test WA f1-score | Epoch |
|---|---|---|---|---|---|---|---|
| None | Fine-tune | 0.0773 | 96.59% | 96.59% | 96.59% | 96.59% | 19 |
| ImageNet | Fine-tune | 0.0728 | 97.44% | **98.59%** | 97.90% | 98.25% | **11** |
| ImageNet | Feature Extractor | 0.1401 | 94.88% | 94.90% | 94.88% | 94.89% | 28 |
| ImageNet & COVID | Fine-tune | **0.0649** | **98.29%** | 98.29% | **98.29%** | **98.29%** | 16 |
| ImageNet & COVID | Feature Extractor | 0.0823 | 97.10% | 97.17% | 97.10% | 97.12% | 29 |

TABLE VI. PERFORMANCE ON PNEUMONIA CLASS IN PNEUMONIA DATASET

| Pre-trained Dataset | Schemes | Test Pneumonia Precision | Test Pneumonia Sensitivity (Recall) | Test Pneumonia F1-Score |
|---|---|---|---|---|
| None | Fine-tune | 97.67% | 97.67% | 97.67% |
| ImageNet | Fine-tune | 98.59% | 97.90% | 98.25% |
| ImageNet | Feature Extractor | 96.72% | 96.27% | 96.50% |
| ImageNet & COVID | Fine-tune | 98.38% | **99.30%** | **98.84%** |
| ImageNet & COVID | Feature Extractor | **98.82%** | 97.20% | 98.00% |

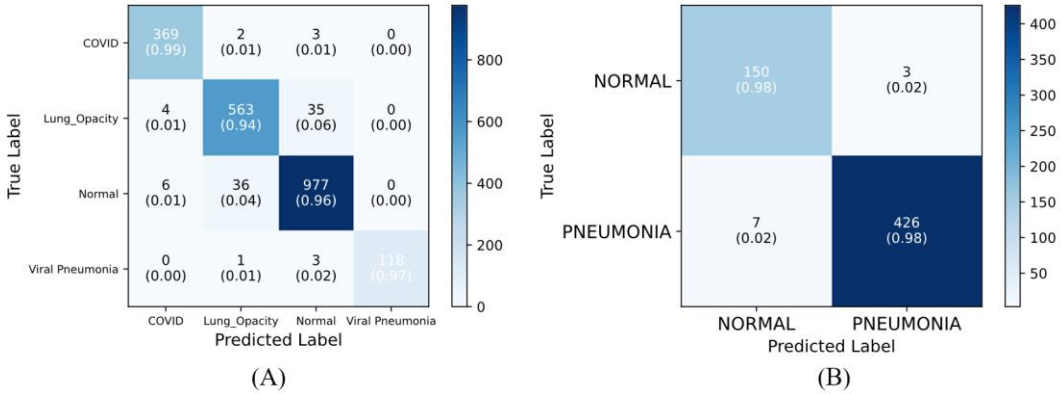


Fig. 8. Confusion matrices of classification on COVID dataset (A), and Pneumonia dataset (B) using our best model

## V. CONCLUSIONS

In this study, we proposed a deep-learning method using chest radiographs to help with COVID-19 and Pneumonia diagnosis.

Specifically, a generic practical paradigm was summarized to devise a suitable model for specific medical imaging classification tasks. Our proposed model consists of a DenseNet-121-based feature extractor and an MLP classifier. Our experiments concluded that using radiographs without non-lung area was possible to train a practicable model but made it harder to train a classification model with high accuracy. Furthermore, a particular weighted cross entropy loss function was selected after experiments to overcome the class-imbalance problem, which not only improved our model's testing accuracy but also reduced training time costs. Finally, after being fine-tuned on two datasets, our pre-trained model achieved satisfying performance with 95.75% testing accuracy, 96.71% sensitivity, and 96.20% precision for the COVID-19 dataset. Regarding the Pneumonia dataset, our model achieved 98.29 % testing accuracy and 99.30% sensitivity for pneumonia samples.

In future studies, we will go further to pre-train some popular models on grayscale ImageNet to thoroughly investigate the remaining doubts on Experiment 1 and make an effort to resolve medical imaging tasks more efficiently.

## ACKNOWLEDGMENT

Yuxuan Sun and Linxu Guo thank Dr. Teoh Teik Toe (Correspondence: ttteoh@ntu.edu.sg; Tel.: 65-97905202), the director of the AI lab at Nanyang Technological University, who provided necessary guidance on this study's topic selection. Without his patience and careful guidance during these months, we would not have completed this study successfully.

## REFERENCES

[1] Rudan, I., Boschi-Pinto, C., Biloglav, Z., Mulholland, K., and Campbell, H. (2008). Epidemiology and etiology of childhood pneumonia. Bull. World Health Organ. 86, 408–416.

[2] Mcluckie, A. (2009). Respiratory disease and its management, Volume 57 (Springer).

[3] WHO COVID-19 Dashboard. Geneva: World Health Organization, 2020. Available online: https://covid19.who.int/ (last cited: 9/29/2022).

[4] Kermany, Daniel S., Michael Goldbaum, Wenjia Cai, Carolina CS Valentim, Huiying Liang, et al. "Identifying medical diagnoses and treatable diseases by image-based deep learning." Cell 172, no. 5 (2018): 1122-1131.

[5] Donahue, J., Jia, Y., Vinyals, O., Hoffman, J., Zhang, N., Tzeng, E., and Darrell, T. (2013). DeCAF: A Deep Convolutional Activation Feature for Generic Visual Recognition. Proceedings of the 31st International Conference on Machine Learning 32, 647–655

[6] Razavian, A.S., Azizpour, H., Sullivan, J., and Carlsson, S. (2014). CNN Features Off-the-Shelf: An Astounding Baseline for Recognition. In 2014 IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 512–519.

[7] Yosinski, J., Clune, J., Bengio, Y., and Lipson, H. (2014). How transferable are features in deep neural networks? NIPS'14 Proceedings of the 27th International Conference on Neural Information Processing Systems 2, 3320–3328.

[8] Japkowicz, Nathalie, and Shaju Stephen. "The class imbalance problem: A systematic study." Intelligent data analysis 6.5 (2002): 429-449.

[9] Shorten, Connor, and Taghi M. Khoshgoftaar. "A survey on image data augmentation for deep learning." Journal of big data 6.1 (2019): 1-48.

[10] Phan, Trong Huy, and Kazuma Yamamoto. "Resolving class imbalance in object detection with weighted cross entropy losses." arXiv preprint arXiv:2006.01413 (2020).

[11] Chowdhury, Muhammad EH, Tawsifur Rahman, Amith Khandakar, Rashid Mazhar, Muhammad Abdul Kadir, Zaid Bin Mahbub, Khandakar Reajul Islam et al., "Can AI Help in Screening Viral and COVID-19 Pneumonia?," in IEEE Access, vol. 8, pp. 132665-132676, 2020, doi: 10.1109/ACCESS.2020.3010287.

[12] Rahman, Tawsifur, et al. "Exploring the effect of image enhancement techniques on COVID-19 detection using chest X-ray images." Computers in biology and medicine 132 (2021): 104319.

[13] Kermany, Daniel; Zhang, Kang; Goldbaum, Michael (2018), "Labeled Optical Coherence Tomography (OCT) and Chest X-Ray Images for Classification", Mendeley Data, V2, doi: 10.17632/rscbjbr9sj.2

[14] Ross Wightman. "PyTorch Image Models." https://github.com/rwightman/pytorch-image-models. 2019. doi: 10.5281/zenodo.4414861. 1

[15] Xie, Yiting, and David Richmond. "Pre-training on grayscale imagenet improves medical image classification." Proceedings of the European Conference on Computer Vision (ECCV) Workshops. 2018.

[16] Huang, Gao, et al. "Densely connected convolutional networks." Proceedings of the IEEE conference on computer vision and pattern recognition. 2017.

[17] Chauhan, Tavishee, Hemant Palivela, and Sarveshmani Tiwari. "Optimization and Fine-Tuning of DenseNet model for classification of Covid-19 cases in Medical Imaging." International Journal of Information Management Data Insights 1.2 (2021): 100020.

[18] Lin, Tsung-Yi, et al. "Focal loss for dense object detection." Proceedings of the IEEE international conference on computer vision. 2017.

[19] Aurelio, Yuri Sousa, et al. "Learning from imbalanced data sets with weighted cross-entropy function." Neural processing letters 50.2 (2019): 1937-1949.

[20] Misra, Diganta. "Mish: A self regularized non-monotonic neural activation function." arXiv preprint arXiv:1908.08681 4.2 (2019): 10-48550.

[21] Javadi Moghaddam, Seyyed Mohammad, and Hossain Gholamalinejad. "A novel deep learning based method for COVID-19 detection from CT image." Biomedical Signal Processing and Control 70 (2021): 102987.

[22] Adeel Hassan. (2021). AdeelH/pytorch-multi-class-focal-loss: 1.1 (1.1). Zenodo. https://doi.org/10.5281/zenodo.5547584

[23] Dozat, Timothy. "Incorporating nesterov momentum into adam." (2016).FENG C, HUANG Z, WANG L L, et al. A novel triage tool of artificial intelligence assisted diagnosis aid system for suspected COVID-19 pneumonia in fever clinics[J/OL]. med Rxiv, 2020: 20039099.

[24] Yuxuan Sun, "TLM-for-COVID-and-Pneumonia-Diagnosis" Github repository. [Online]. Available: https://github.com/Erostrate9/TLM-for-COVID-and-Pneumonia-Diagnosis