Detection of Lung Diseases through image classification

Ching-Wen Wang
cwang553@wisc.edu

Erika Yeonseo Park
ypark258@wisc.edu

Jina Yang
jyang633@wisc.edu

Project Link
https://github.com/JasmineWang553/Stat-453-Chest-X-ray-Data

Abstract

Due to the high volume of people infected with COVID-19 and self-reports of respiratory symptoms, medical institutions were unable to provide quick screenings, diagnosis, and appropriate treatments to all lung disease patients while executing preventive measures for COVID-19 at the same time. We are motivated to provide effective deep learning models to alleviate burdens on hospitals in providing quick and accurate diagnosis of lung diseases, in particular for COVID-19 and Viral Pneumonia patients which have known potential risks of spreading. We utilized transfer learning to train on Convolutional Neural Networks (CNN), AlexNet, ResNet50, and VGG19 with batch normalization to train on a Radiography Chest X-ray image database. Our primary aim is to select models with lowest misclassifications in predicting "COVID" and "Viral Pneumonia" class as "Normal" and the highest test set accuracy. In our study, ResNet50 was determined as the most suitable model to our project's goal with its comprehensive performance in predicting the correct class labels and the best generalization performance, in which it achieved a test set accuracy of 95.46%.

1. Introduction

Failure to detect lung disorders or have delayed diagnosis brings devastating results for the patient. COVID-19 pandemic is an ongoing global public health crisis that has started infecting the world population in January 2019 and resulted in 148 million cases with 3.1 million deaths up until the end of April 2021 [39]. COVID-19, caused by a fatal virus SARS-CoV-2 that originated from bats, can lead to severe complications such as respiratory failures and long-term organ injuries [15]. Compared to two previously identified coronavirus discovered in this century, which are Sars-CoV in 2003 and MERS-CoV in 2012, symptoms of Sars-CoV-2 are relatively more moderate. However, it's growth rates and infection rates completely dominate the other two. Sars-CoV-2 is considered to be lethal to certain groups of the population, including elderly people and people with weakened immune systems. Not only that it can cause serious and irreversible respiratory complications, but it can also lead to kidney and heart failures. Consequently, early detection of the virus is critical for increasing survival rates, speeding up treatments, and preventing the spread of the diseases.

According to professionals in the front line of the respiratory field, “From a diagnostic point of view, interstitial lung diseases are confusing diseases because signs and symptoms are similar to a wide range of respiratory conditions.” [20] This brought out concerns that people who have been infected by fatal and contagious viruses, such as COVID-19, may falsely be diagnosed to other lung diseases. If people who has COVID-19 is being mistakenly identified as other lung diseases, it is highly probable that the diagnosis results can lead to lack of opportunity in seeking the right medical help and taking correct preventive measures to stop the spread of the disease.

Hence, it is imperative to utilize sophisticated methods, such as digital assistants, to achieve the highest degree of protection and diagnostic speed in identifying the cause of their symptoms, and guide them for further treatments.

The two most prominent type of COVID testing that can be performed in a quick setting are antigen test and polymerase chain reaction (PCR) tests, which detects protein fragments and Ribonucleic Acid(RNA) that is specific to the Sars-CoV-2 [14]. Compared to the two efficient testing methods, Chest X-radiation(X-ray) scans can be considered as an expensive and inefficient way to detect signs of coronavirus. However, X-ray holds its strong merits in detecting not only coronavirus, but also lung tumors, pneumonia, cystic fibrosis, and many other lung diseases that can help along with the disease's spread and its severity. Therefore with its availability, affordability, and accessibility in general hospitals, X-ray image is regarded as one of the most important modes of information in aiding the doctor's diagnosis. Not to mention that the X-ray images is an important data source in the image recognition field of machine learning and deep learning communities. Besides X-ray images, it is argued that computed tomography(CT) scans would give more sophisticated results by providing three-dimensional images. However, taking the affordable prices and availability into account, X-ray would be a more preferable choice for the general public and data collection purposes.

Accordingly, the goal of our project is to experiment with several deep learning models to quickly and accurately identify patients' lung diseases given their chest X-ray images. Conceivably, the experimented deep learning models would aid to establish a useful building block for future works to generate digital assistants for hospitals. The motivation for the project came from the current state of the COVID-19 pandemic, where people face imbalanced situations between lacking hospital resources and the overflowing amount of patients visiting the hospitals. The imbalanced situation has made it difficult for patients who suffer from respiratory problems other than COVID-19 to access their required treatments while staying safe without getting infected.

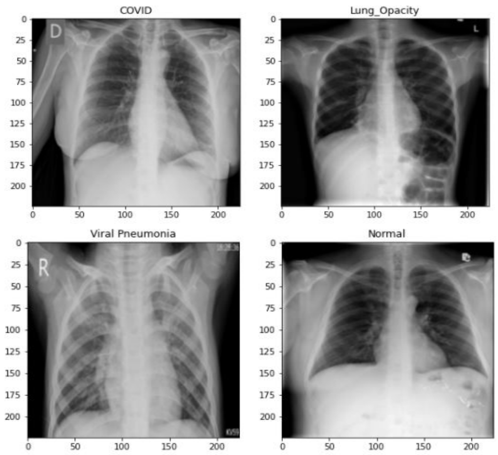


Figure 1: Original Image Data

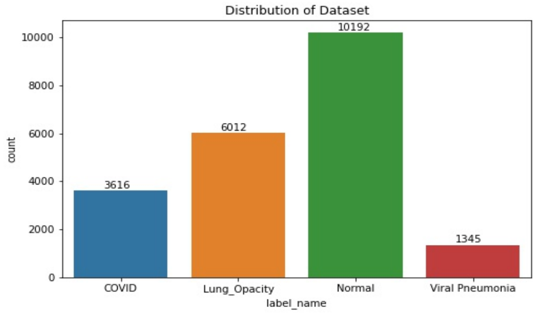


Figure 2: Distribution of Radiography Dataset

2. Dataset

An obtained radiography database of chest X-ray images is examined [35]. The database is acquired by a team of researchers from Qatar University, Doha, Qatar, University of Dhaka, Bangladesh and their collaborators, medical doctors, from Pakistan, and Malaysia. The database is gathered from different sources including Italian Society of Medical and Interventional Radiology (SIRM) COVID-19 DATABASE [24], Novel Corona Virus 2019 Dataset developed by Joseph Paul Cohen and Paul Morrison, and Lan Dao in GitHub [38], and images extracted from 43 different publications [8][26]. The database contains 21,165 chest X-ray images with 4 classes: COVID-19, viral pneumonia which is an infection of lungs caused by viruses that are not COVID, opaque lungs that are non-COVID and non viral pneumonia infections, and healthy lungs [figure1]. There are 3616 images labeled “COVID”, 6,012 images for “Lung Opacity”, 10,192 images for “Normal”, and 1,345 images for “Viral Pneumonia” [figure2].

3. Related Works

Studies have been conducted since the outbreak of pandemic to acquire more information on the contagious virus. Considering the advancement in the field of computer vision on X-ray images, many methods and techniques are implemented to target the correct prediction of Sars-CoV-2 and improve the accuracy respectively. For the given task of chest X-ray image classification, three previous studies are reviewed, which propose effective pre-processing, training, and evaluating methods for image classification tasks performed primarily on classification of lung diseases.

Dataset & Pre-processing Previous studies have investigated and evaluated pre-processing techniques on X-ray images by employing deep learning models. Gordienko Y. et al. [12] concentrated on detecting marks of lung cancer from Chest X-ray images while considering pre-processing techniques of lung segmentation through UNet-based CNN and bone elimination that excludes clavicle and rib shadows. The data set utilized for training and validation is from the JSRT data set, which consists of 247 images containing 154 cases with lung nodules and 93 cases without lung nodules. To assess the adequacy of the pre-processing techniques, Gordienko Y. et al. constructed another data set, which was formed by eliminating bone shadows(BSE-JSRT) from the same 247 images from the JSRT data set with pre-processing methods. The final results showed improved accuracy and decreased loss in BSE-JSRT. Thus, the pre-processing method substantiated the effectiveness of image pre-processing before training neural networks. Despite the study demonstrated the practicability and efficiency of the proposed bone shadow elimination and segmentation techniques, the study has the limitation of having a high degree of over-fitting.

Transfer learning & Training Along with the advancement of the deep learning field and the outbreak of the pandemic, numerous architectures are being developed to target COVID-19. Khan A. et al.[18] proposed CoroNet, which is a deep neural network utilizing Convolutional architecture to detect numerous lung diseases: Normal, Pneumonia Bacterial, Pneumonia Viral, and COVID-19. The dataset utilized for the study is created by collecting Chest X-ray images from two different databases to reach an appropriate data size, which in total consisted of 1300 images. The proposed algorithm "CoroNet" is a CNN that has its basis on Xception architecture and is pre-trained on the ImageNet dataset. Transfer-learning is utilized through employing Xception models. The Xception models are fine-tuned and revised to adapt X-ray images. Although the overall accuracy is 89.6 % for 4-class prediction and 93% for 3-class classification of COVID, Pneumonia, and normal, the study provided useful information to guide clinical practitioners and radiologists in facing future diagnosis and guidance of COVID-19 patients.

Contrasting to the limited dataset for CoroNet, Covid-Net, a neural network proposed by Wang et el.[36], is structured with a combination of convolution layers with a range of kernel sizes from 7x7 to 1x1. The model is pre-trained on the ImageNet dataset[16] before training on 13,975 Chest X-Ray(CXR) images that have 8,066 Normal (healthy people), 5,538 Non-COVID (non-COVID-19 pneumonia), and 266 COVID patient cases. The model is distinctive by having its long-distance connectivity in various parts of the network which preserves essential information to aid the prediction of labels. Data augmentation and image normalization are applied before training the Covid-Net. When the model performance was estimated, Covid-Net achieved a 93.3% accuracy which successfully exceeds ResNet50 and VGG19 that has 83.0% and 80.6 % test accuracy. Also, the model achieved a 98.9% PPV, which means that the false-positive rate for COVID-19 predictions is low.

Besides studies that developed their own framework and training on imbalanced datasets, a study was conducted on utilizing existing deep learning frameworks such as ResNet50, VGG16, VGG19, and DenseNet121. Data augmentation of rotating, flipping, resizing images to 224x224x3 were applied to images and transfer learning is employed. The models were trained on 630 COVID-19 X-rays and 642 Normal X-rays and tested on 100 test X-ray images each in both classes. VGG19 was found to be the best and achieved a test accuracy of 99.33% and false negative rate (FNR) of 0.0 %. The proposed study is similar to our proposed direction in the project, however, we will be utilizing more images, in particular 2,986 more images with COVID labels and 2 more class labels associated.

4. Method Description

The given task of image classification on X-ray images are performed and evaluated through transfer learning, modeling frameworks, and visualizing techniques.

4.1. CNN

Convolutional neural networks(CNN) are explicitly used for the given task of classifying the image data because CNNs are known to be "a specialized kind of neural network for processing data that has a known grid-like topology[11]." CNNs are one of the most basic and reliable algorithms for image classification that is commonly in use. The convolutional layers of CNNs are often described as feature extraction layers by automatically extracting and learning relevant features from raw data that preserves important information on image traits.

As figure (3) shows, the raw data will be input through a convolutional layer first, and followed by pooling layers that will extract low-level features from the raw data. Pooling operations are sub-sampling layers that are utilized in particular to increase computational efficiency and reduce over-fitting. To be specific, one of the sub-sampling methods, max-pooling, takes the maximum value from a neighbor of pixels, while another sub-sampling method, mean-pooling, takes the mean values from the neighboring pixels. With a full pass of going through combinations of convolutional and pooling layers, the extracted features will be pass through the later fully-connected layers, also known as multilayer perceptron, to predict a target value or class label for the input data.

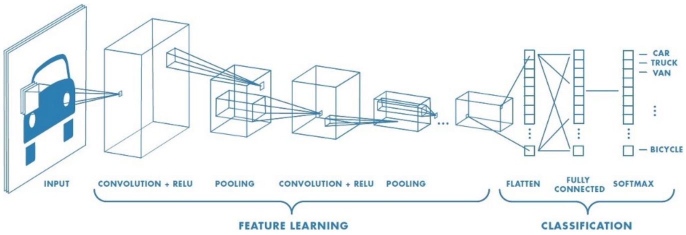


Figure 3: Visualization of CNN Structure [32]

In addition to the layer itself, activation layers such as layers implementing **ReLu activation function** follow the convolutional or linear layers to provide a weighted sum of inputs and introduce non-linearity to the models. Net inputs z from the previous layer will be used to activate the hidden unit $a = \sigma(z)$, where σ is the activation function chosen as the ReLu function (Equation 1) [29].

$$\sigma(z) = \begin{cases} 0, & \text{if } z < 0 \\ \text{z}, & \text{if } z \geq 0 \end{cases} \tag{1}$$

A commonly seen activation in multi-class classification is the **softmax** activation function. It compresses net outputs of the last layer into the range of $(0, 1)$ and enables the resulting elements to be interpreted as class probabilities.

Different techniques, such as batch normalization[17], are also utilized to speed up the training process and adapt the model for better generalization performance to unseen data. Batch normalization successfully standardizes the distribution of the input data by centering the data around 0. Consequently, the internal co-variance is greatly reduced, which leads to accelerated training processes, allowance for higher learning rates, and more lenient rules in parameter initialization.

The calculated class probabilities are passed into loss functions. Then, parameters are updated according to the calculated gradients in each layer [3] to evaluate the model performance and allow back propagation, which updates the parameters to improve the models' training accuracy.

One common loss function for multi-class classification is the **cross-entropy loss** (equation 2)[30]. In this function, classes are assumed to be mutually exclusive.

$$H_a(y) = \sum_{i=1}^{n} \sum_{k=1}^{K} -y_k^{[i]} log(a_k^{[i]}) \tag{2}$$

4.2. CNN Variants

Benefiting from the powerful architecture of CNN in capturing relevant features from raw data, variants of CNN are continued to be developed. Limited by computational expenses and hardware specs, we will only be using 3 CNN models: AlexNet, Visual Geometry Group(VGG), and the Residual Network (ResNet).

AlexNet AlexNet is designed by Alex Krizhevsky with Ilya Sutskever and Geoffrey Hinton. AlexNet contains 8 layers with learnable parameters in which 5 layers are convolutional layers with decreasing filter size and 3 are fully connected linear layers [37]. In 2012, AlexNet won the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) with top-5 error rate of 15.3%, which is considerably low compared to 26.2% from the model that placed second class [22]. The secret of the low error rates stems from the implementation of rectified linear units (ReLU) as activation function instead of tanh and an overlapping pooling strategy, which preserves the local connection between results[37]. The change of activation function speeds up the training process 6 times compared to a regular CNN model trained with tanh activation function. Moreover, the application of overlap pooling strategy awarded them with a reduction of the top-1 and top-5 error rates by 0.4% and 0.3% [22]. However, the large number of neurons 650,000 and 60 million parameters can be problematic due to overfitting and long computational time. Therefore, data augmentations, which is a pre-processing method of flipping or rotating images, and dropout, which is randomly dropping neurons to break model's dependency on certain neurons, has to be performed. Although AlexNet is an iconic model in the computer vision community, there are more complex models with different layering strategies that outperforms AlexNet in generalization accuracy.

VGG VGG is a convolutional neural network that is proposed by the Visual Geometry Group(VGG) at the University of Oxford in the paper "Very Deep Convolutional Networks for Large-Scale Image Recognition." [34] The strength of VGG lies in its use of small-sized convolution filters (3x3) that leads to better-generalized performance in large-scale image recognition. With the small convolution filters, it enables the extension of the model's depth to 19 layers with trainable weights, which is deeper compared to AlexNet's 8 learnable layers. The depth of VGG networks further enables the opportunity of implementing multiple ReLu functions, which adds non-linearity and amplifies the learning effect of the model.

However, the downside of VGG is its slow training time and computational efficiency. Moreover, VGG is vulnerable to vanishing gradients problems, which drastically degrades the model performance.

ResNet Residual Networks (ResNet) are deep neural networks, which are similar to VGG nets. However, ResNets take a step-by-step approach with skipped layers, which contain the ReLu activation function(equation 1) and batch normalization(BatchNorm2D) [17]. Adding more layers is expected to help when building a neural network. However, problems often arise, such as vanishing or exploding gradient problems. The skip connections accommodate the problem by allowing the network to jump over layers. Skipping effectively simplifies the network that speeds learning by reducing the impact of vanishing gradients as there are fewer layers to propagate through. ResNet models are also discussed to be "feed forward neural networks with shortcut connections"[13].Using the skip connections, the forward propagation is described as follows:

$$\begin{aligned} a^{(l+2)} &= \sigma(z^{l+2} + a^{(l)}), \\ &= \sigma(a^{(l+1)}W^{(l+2)} + b^{(l+2)} + a^{(l)}) \end{aligned} \tag{3}$$

where a is an activation function, W is a weight matrix, b is a bias vector, and (l), ($l + 1$), and ($l + 2$) denote layer location[28]

4.3. Transfer Learning on pretrained models

Packages such as Keras[7], Tensorflow[2], and Pytorch[25] provide options to fine tune the pre-trained models for different training purposes, instead of manually implementing complex models. In the project, we will be using transfer learning on the pre-trained CNN architectures from Pytorch-TorchVision [23] that was trained on a subset of ImageNet data set[9], which contains 1,281,167 training images, 50,000 validation images, and 100,000 test images that has 1000 unique labels with 224x224 colored images as input [16]. With transfer learning, the model uses and applies patterns that have already been learned on the given task, instead of training parameters from scratch. Transfer Learning is particularly useful in achieving high accuracy for relatively small data sets.

The obtained X-ray image data set for the project is concluded to be relatively small and unbalanced. Thus, transfer-learning method is employed; pre-trained models are fine-tuned for our classification purposes.

4.4. GradCAM

GradCAM is a method to visualize the process of what CNN model layers focus on in object recognition [33]. The most important feature of GradCAM is its highlighting feature that focuses on the regions in the image that CNN model layers focused on for predicting an image as its true class label. The method works in capturing the essential gradients flowing into the target layer and mapping them onto the image with heat map localization coloring. If region appears to be close to red rainbow color spectrum, it means that the part of the image is focused most by the model. Contrastingly, if region shows light blue or no coloring, the region is not in the model layer's interest during the classification process.

5. Proposed Method

The main task of our project is to correctly predict lung diseases on given patients' chest X-ray images, especially COVID-19 and viral pneumonia. Although lung opacity may be contagious too, we do not have enough information on the actual disease name of each image with lung opacity label to know it is infectious, thus we are emphasizing more on lowering misclassifications of viral pneumonia and COVID-19, which hold high risks in droplet infection. We will be utilizing variants of Convolutional Neural Networks (CNN), AlexNet, ResNet, and VGG, to train on a Radiography Chest X-ray image database [8][26] to determine the best model for the classification task. Due to the imbalanced data set and the small number of images in our target classes, we will be implementing image pre-processing and utilizing transfer learning along with hyperparameter tuning to enhance the model's performance. The CNN models will be benchmarked by our baseline model with logistic regression. The best model will be chosen based on the lowest number of misclassifying ”COVID” and ”Viral Pneumonia” patients as ”Normal” and the highest test set accuracy.

5.1. Pre-Processing

Augmentation Data Augmentation such as rotation and horizontal flipping was implemented on images before model training. The augmentation is to prevent models from only recognizing images that are placed in the same position. We also resized the images to 224x224x3 to match the image size that pre-trained models were trained on.

Standardization Before comparing models and algorithms, z-score standardization (Equation 4) is proceeded to standardize image pixels. Instead of calculating the mean and the standard deviation of the pixel values from the obtained data set, two different sets of mean and standard deviation values were utilized. The first set contains normalization values that are specific for radiography images, and the other matches the ImageNet's normalization mean and standard deviation which was used previously in pre-trained models.

$$x_{std}^{[i]} = \frac{x^{[i]} - \mu_x}{\sigma_x} \tag{4}$$

where $x^{[i]}$ = a pixel value in an X-ray image, μ_x = the population mean of ImageNet dataset, σ_x = the standard deviation of ImageNet dataset

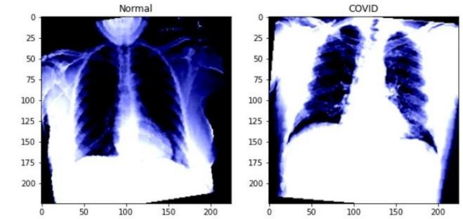


Figure 4: standardization and data augmentation applied images

5.2. Train

Dataset is split into 80% Train, 10% Validation, and 10% test with each set holding 16,932, 2117, and 2116 images from 4 classes. Three models of CNN: AlexNet, VGG, and ResNet will be used, and different types and layers of each model will be experimented on. We will be loading pre-trained models from the Pytorch package and fine-tune each model respectively to achieve its best performance. Hyperparameter tuning will also take place with trials for different models using different standardization values, optimizers, batch sizes, and epoch sizes. Also, to obtain comparable model performances, a random seed is set to 1 in train-test set splitting and model training.

Model Name	Train Acc	Test Acc	(fn) COVID	(fn) Viral Pneumonia	(fn) Lung Opacity
AlexNet	97.87%	93.86%	5	3	63
VGG19_bn	98.12%	94.33%	6	5	51
ResNet50	**98.59%**	**95.46%**	**3**	**6**	**54**
Logistic Regression	60.84%	62.24 %	214	25	197

Table 1: Comparing Models: (fn)TRUE LABEL is number of images that are misclassified as ”Normal” when the real label is TRUE LABEL in test set prediction

5.3. Evaluation

The best model obtained by comparing different hyperparameter settings and fine-tuning of layers are evaluated in different ways.

$$Accuracy = \frac{1}{n}\sum_{i=1}^{n} L(\hat{y}^{[i]}, y^{[i]}) \times 100, \tag{5}$$

$$L(\hat{y}, y) = \begin{cases} 1 & \text{if } \hat{y} = y, \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

where n = the number of examples in the test set, $\hat{y}$ = predicted class label, y = the ground truth

The models with lowest missclassification rate of ”COVID” and ”Viral Pneumonia” patients as ”Normal” and the highest test accuracy (5) will be selected within each variants of the same type of algorithm, for example Resnet34 vs. Resnet50. Then, applying with the same criteria, each classes' precision (7), recall (8) and F1 score (9) will be calculated for each model in order to determine the best model suitable for our project's purpose. We are primarily aiming for models with high recall rates in ”COVID” class, since it represents the proportion of correctly classified ”COVID” labels when ”COVID” should be predicted.

$$\text{precision} = \frac{\text{true positive}}{\text{true positive} + \text{false positive}} \tag{7}$$

$$\text{recall} = \frac{\text{true positive}}{\text{true positive} + \text{false negative}} \tag{8}$$

$$F1 = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \tag{9}$$

Class Label	Precision	Recall	F1 score
COVID	25.86	28.75	27.23
Lung Opacity	53.85	70.59	61.09
Normal	88.04	67.14	76.19
Viral Pneumonia	0.0	nan	nan

Table 2: Logistic Regression

Class Label	Precision	Recall	F1 score
COVID	96.55	94.65	95.59
Lung Opacity	88.78	94.22	91.42
Normal	95.65	93.17	94.39
Viral Pneumonia	96.97	95.52	96.24

Table 3: AlexNet

Class Label	Precision	Recall	F1 score
COVID	97.7	94.44	96.05
Lung Opacity	90.38	94.31	92.31
Normal	95.36	93.96	94.65
Viral Pneumonia	**96.21**	**96.95**	**96.58**

Table 4: VGG19 with batch norm

Class Label	Precision	Recall	F1 score
COVID	**98.28**	**97.71**	**97.99**
Lung Opacity	90.71	96.42	93.48
Normal	97.43	93.99	95.68
Viral Pneumonia	95.45	96.92	96.18

Table 5: ResNet50

6. Results

In the end, ResNet50 is proven to be the model with the best performance after all. With fine-tuning the layer 3 and 4 block, which each contains 3 blocks with 3 convolutional layers that are 1x1, 3x3, and 1x1, and 1 linear layer, in the end, ResNet50 yielded the best test accuracy 95.46% and the highest f1 score for labels COVID and F1 score. It was found that fine-tuning the last convolutional layer of Alexnet and its 6 fully connected layers yielded a decent result by obtaining a test accuracy of 93.86%. For VGG models, setting the last 3 convolutional layers and its 6 fully connected layers to require gradients in training is the best in generalizing performance and having the lowest misclassification rates in predicting class labels. In particular, VGG19 with batch norm achieved a fairly well perfor-

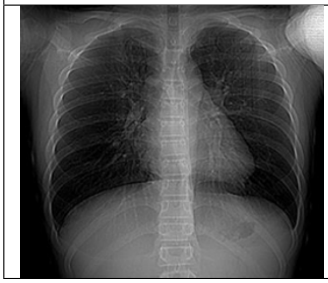
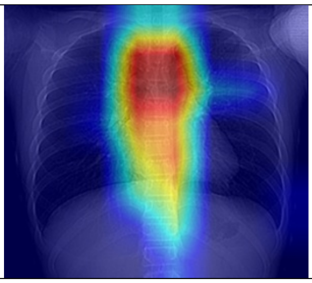
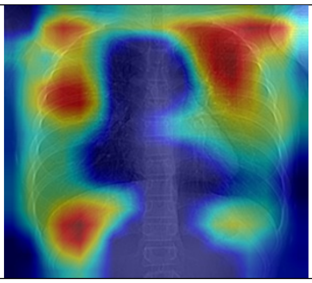
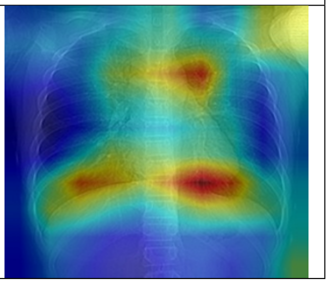
Original COVID image	AlexNet	VGG19 batch norm	ResNet50
None	Layer 5	Layer 16	Layer 49
			

Table 6: Visualizing focus regions of the corresponding convolutional layer in CNN models

mance in obtaining 94.33% test accuracy.

In behalf of the goal of our project of detecting COVID, recall score is also regarded as an important criteria for evaluation metric. High recall scores means to have lower number of falsely classified cases. Thus, achieving high recall scores is one other important criteria with the goal of achieving high test accuracy.

Judging by the evaluation criteria, ResNet50 is still considered to be the best model among all models. ResNet50 not only yielded the lowest amount of misclassification rate of the highly contagious ”COVID” among all models which resulted in a recall rate of 97.71%, it also achieved the highest precision rate of 98.28% in predicting the labels as ”COVID” correctly. Another class label that is also focused on is ”Viral Pneumonia” due to its easy spread. Although ResNet50 has a high recall rate 96.92 % in predicting viral pneumonia class, VGG19 with batch norm exceeded the recall rate with the best recall rate of 96.95% and f1 score of 96.58 % associated with the ”Viral Pneumonia” class. Therefore, in predicting the ”Viral Pneumonia” class correctly, VGG19 with batch norm would be a more suitable choice.

7. Discussions

Regarding to pre-processing methods, we initially planned to utilize Contrast limited adaptive histogram equalization (CLAHE) to enhance the images for better model performance. However, it is recognized that image enhancement methods such as CLAHE, Successive Means Quantization transform(SMQT), Wavelet transform, and Laplace operator are likely to degrade the results of transfer learning through Chen's paper [6]. The result was further confirmed by conducting test trials with CLAHE utilizing the AlexNet, in which test accuracy of only 73.37% was yielded. Since the model performed with the experimented pre-processing methods resulted in worse generalization performance compared to the general test accuracy of 90+%, CLAHE was discarded from our options in the pre-processing stage.

Moreover, regarding normalization methods, it is further discovered that the mean and standard deviation for radiography images outperformed the set of mean and standard deviation values used for ImageNet dataset in all 3 models in terms of producing lower than 8 misclassification counts of ”COVID” or ”Viral Pneumonia” patients as ”Normal”. Thus, only resizing, using the best set of normalization values, and data augmentations were applied to images [figure 4].

During the training of models, we have each performed different sets of hyperparameter tuning on all 3 models. Better results were reached by using the larger batch sizes and smaller epochs for complicated models, such as VGG19 and ResNet50, and smaller batch sizes and larger epochs for simpler models, for example, AlexNet. The fine-tuning of layers was also a major element in our training process. We discovered that fine tuning around 20-25% of the convolutional layers and linear layers can yield test accuracy around 1-1.5 % higher than accuracy of solely tuning on the linear and fully connected layers. By freezing gradients of the earlier layers and unfreezing gradients of the later layers, the model can be adjusted to interpret images from our dataset and extract more relevant features. Another experiment we tried on was comparing Adaptive Moment Estimation (Adam) optimizer [19] against stochastic gradient descent (SGD) optimizer. As a result, Adam optimizer outperformed the SGD optimizer of around 6% in test accuracy with the same setting of hyperparameters on the same model. Although the actual reasons may not be clear, we suspect the Adam's outstanding performance could possibly be due to its ability of adjusting learning weights by the calculated squared gradients and momentum accordingly. [4].

After obtaining test accuracy rates, precision, recall, and f1 scores, we applied GradCAM [33] to one of the COVID mages as a method of visualization. GradCAM visualizes the region that each of the models focused on the most while the model interpreted and predicted class labels. We were able to display an X-ray image of a COVID patient with the focusing regions of each model's last convolutional layer [table 6]. From the GradCAM images, we clearly observed that ReNet50 is well deserved as the best model we found, because it did not over-emphasize the region of the heart but instead focuses more on the expanding tree similar to network in the lungs that is an evident symptom of COVID-19. For AlexNet, we can see that the model is emphasizing more on the esophagus region and where the lungs connect of the patient, but it still performs well in capturing the network like structure in the connecting region of the lungs. Interestingly, even though the test of VGG accuracy is higher than AlexNet in classifying COVID class labels, VGG19 with batch norm does not do well enough in capturing the essential feature of COVID patients. Instead, it focuses more on the body boundaries of the person.

This inefficient capturing of VGG19 is also reflected in the number of misclassifications of ”COVID” as ”Normal” in table 1. Compared to AlexNet, VGG19_bn's number of misclassifications in ”COVID” on the test set exceeded AlexNet's misclassifications by 1, and its recall 94.44 % is slightly lower than AlexNet's recall rate 94.65 %, which means VGG19 with the batch norm is more likely to wrongly predict COVID patients as other class labels than AlexNet.

Nonetheless, ResNet50 can be prone to over-fitting even if it seems powerful, since the training accuracy reaches around 99 to 100 % after 25 epochs of training and the validation accuracy was observed to plateau around 93 to 94% after 15 epochs. Despite the limitation of overfitting, our hardware and software devices are also constraints that restricted performances of all 3 models, especially in the fine tuning level of pre-trained models. The best combination of hyperparameters for each model were applied as close as possible to the restriction of our device.

8. Conclusion

Overall, transfer learning is a preferable method to train on imbalanced and small data set like ours. All the models have exceeded the bench mark model Logistic Regression's train and test accuracy, thus demonstrated the effectiveness of these models in classification tasks. In this study, by utilizing pre-trained models, ResNet50 was determined as the most suitable model for our project's purpose in which the test accuracy is 95.46% and precision, recall and f1 score for 3 classes, ”COVID”, ”Lung Opacity”, and ”Normal” are fairly high. Although in predicting ”Viral Pneumonia” VGG19 with batch norm may yield better results, ResNet50's comprehensive performance outshined other models in its highest generalizing accuracy in predicting the correct labels and lowest misclassifiction rates in classifying, especially for ”COVID” patients.

Our study result's may be limited to technical difficulties, but we believed that it shed some light on the importance of transfer learning in classifying chest X-ray images with correct labels, especially on the latest corona virus Sars-CoV-2. Considering the constraints of our devices, future directions include implementing more image pre-processing methods, for example lung segmentation and rib cage shadow removal, applying transfer learning on other more complicated pre-trained models such as wide ResNet, DenseNet, or models that are pre-trained on relevant X-ray images. In the future, we hope that not only in the times of pandemic but also in a day-to-day setting that deep learning models will be able to alleviate burdens from both patients and hospital workers by providing help in speedy diagnosis and appropriate treatments in the best possible way.

9. Resources

The chest X-ray image dataset is obtained from Kaggle[1], which is a data platform that offers public data. The primary python libraries we used include NumPy and pandas to clean and tidy data, Matplotlib and opencv for plotting pictures and graphs, and PyTorch[25] for building a wide range of Neural Networks. Data loaders, evaluations, plotting, and model training are performed through utilizing helper functions that are provided from a lecture by Professor Raschka [27]. Code was also borrowed from Mr. Debarshi Chanda's kaggle notebook [5] to generate GradCAM images. The computer hardware for running $Python$ 3.8.5 [31] through $Jupyter\ Notebook$ [21] was each members' lab-top. Google Colab platform is used for model training and to work collaboratively. Python scripts were also utilized to create helper functions. All members worked collaboratively by utilizing Google Colab and GitHub [10] to share codes to work together more efficiently.

10. Contribution

Each group member contributed to the modeling and writing process. Ching-Wen implemented helper functions for pre-processing step. Ching-Wen also prepared templates for each algorithm and guided the direction for the project. For the experimental step, Ching-Wen built the codes for the general framework and experimented with AlexNet. Erika built the codes and experimented with ResNet. Jina was responsible for VGG and visualization of the chosen models for each algorithm. All group members participated in writing the report. Erika was responsible for the introduction, related work, and method description. Ching-Wen provided resources by researching and organizing the information, writing proposed methods, and results. Jina was responsible for organizing the results and writing the discussion and conclusion sections. Each member helped to write every section of the report.

References

[1] Kaggle. `https://www.kaggle.com`.

[2] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org.

[3] S.-i. Amari. Backpropagation and stochastic gradient descent method. *Neurocomputing*, 5(4-5):185–196, 1993.

[4] V. Bushaev. Adam — latest trends in deep learning optimization.

[5] D. Chanda. Gradcam: Visualize your cnn.

[6] X. Chen. Image enhancement effect on the performance of convolutional neural networks, 2019.

[7] F. Chollet. Keras. `https://github.com/fchollet/keras`, 2015.

[8] M. E. H. Chowdhury, T. Rahman, A. Khandakar, R. Mazhar, M. A. Kadir, Z. B. Mahbub, K. R. Islam, M. S. Khan, A. Iqbal, N. A. Emadi, M. B. I. Reaz, and M. T. Islam. Can ai help in screening viral and covid-19 pneumonia? *IEEE Access*, 8:132665–132676, 2020.

[9] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009.

[10] github. Github, 2020.

[11] I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. MIT Press, 2016. `http://www.deeplearningbook.org`.

[12] J. H. W. Z. K. Y. A. O. R. O. Gordienko Yu, Gang Peng. Deep learning with lung segmentation and bone shadow exclusion techniques for chest x-ray analysis of lung cancer. *Advances in Computer Science for Engineering and Education*, 754, 2019.

[13] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition, 2015.

[14] M. Healthcare. What's the difference between covid-19 rapid and pcr tests?

[15] Healthline. What to know about covid-19 and pneumonia.

[16] ImageNet.

[17] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift, 2015.

[18] A. I. Khan, J. L. Shah, and M. M. Bhat. Coronet: A deep neural network for detection and diagnosis of covid-19 from chest x-ray images. *Computer Methods and Programs in Biomedicine*, 196:105581, 2020.

[19] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[20] J. S. K.K. Mujeeb Rahman. Diagnostic delay and misdiagnosis in interstitial lung disease(ild) at primary health care level. *European Respiratory Journal*, 48, 2016.

[21] T. Kluyver, B. Ragan-Kelley, F. Pérez, B. Granger, M. Bussonnier, J. Frederic, K. Kelley, J. Hamrick, J. Grout, S. Corlay, P. Ivanov, D. Avila, S. Abdalla, and C. Willing. Jupyter notebooks – a publishing format for reproducible computational workflows. In F. Loizides and B. Schmidt, editors, *Positioning and Power in Academic Publishing: Players, Agents and Agendas*, pages 87 – 90. IOS Press, 2016.

[22] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25:1097–1105, 2012.

[23] S. Marcel and Y. Rodriguez. Torchvision the machine-vision package of torch. In *Proceedings of the 18th ACM International Conference on Multimedia*, MM '10, page 1485–1488, New York, NY, USA, 2010. Association for Computing Machinery.

[24] S. Mittal, V. K. Venugopal, V. K. Agarwal, M. Malhotra, J. S. Chatha, S. Kapur, A. Gupta, V. Batra, P. Majumdar, A. Malhotra, et al. A novel abnormality annotation database for covid-19 affected frontal lung x-rays. *medRxiv*, 2021.

[25] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer. Automatic differentiation in pytorch. 2017.

[26] T. Rahman, A. Khandakar, Y. Qiblawey, A. Tahir, S. Kiranyaz, S. B. Abul Kashem, M. T. Islam, S. Al Maadeed, S. M. Zughaier, M. S. Khan, and M. E. Chowdhury. Exploring the effect of image enhancement techniques on covid-19 detection using chest x-ray images. *Computers in Biology and Medicine*, 132:104319, 2021.

[27] S. Raschka. Lecture 12: Improving gradient descent-based optimization. University Lecture.

[28] S. Raschka. Lecture 14: Introduction to cnns part 2: with applications in python – cnn architectures. *STAT453 Lecture slides*, 2021.

[29] S. Raschka. Lecture 6: Automatic differentiation with pytorch. *STAT453 Lecture slides*, 2021.

[30] S. Raschka. Lecture 8: Logistic regression and multi-class classification. *STAT453 Lecture slides*, 2021.

[31] V. G. Rossum and F. L. Drake. *Python 3 Reference Manual*. CreateSpace, Scotts Valley, CA, 2009.

[32] S. Saha. A comprehensive guide to convolutional neural networks — the eli5 way.

[33] 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 *Proceedings of the IEEE international conference on computer vision*, pages 618–626, 2017.

[34] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition, 2015.

[35] A. K. Tawsifur Rahman, Muhammad Chowdhury. Covid-19 radiography database covid-19 chest x-ray database.

[36] L. Wang, Z. Q. Lin, and A. Wong. Covid-net: A tailored deep convolutional neural network design for detection of covid-19 cases from chest x-ray images. *Scientific Reports*, 10(1):1–12, 2020.

[37] J. Wei. Alexnet: The architecture that challenged cnns.

[38] H. B. Winther, H. Laser, S. Gerbel, S. K. Maschke, J. B. Hinrichs, J. Vogel-Claussen, F. K. Wacker, M. M. Höper, and B. C. Meyer. Covid-19 image repository, may 2020. *URL https://figshare. com/articles/dataset/COVID-19_Image_Repository/12275009/1*.

[39] Worldometer. Covid-19 coronavirus pandemic live updates.