

Comprehensive Study of Coronavirus Disease 2019 (COVID-19) Classification based on Deep Convolution Neural Networks

Miyuka Nakamura*, Jiangkun Wang**, Sinchhean Phea^{1,***}, and Abderazek Ben Abdallah

University of Aizu, Graduate School of Computer Science and Engineering, Adaptive Systems Laboratory, Aizu-Wakamatsu, Japan

Abstract. Artificial Intelligence (AI) has recently become a topic of study in different applications, including healthcare, in which timely detection of anomalies can play a vital role in patients health monitoring. The coronavirus disease 2019 (COVID-19), caused by the SARS-CoV-2 virus, colloquially known as the Coronavirus, disrupts large parts of the world. The standard way to test for COVID-19 is Reverse Transcription Polymerase Chain Reaction (RT-PCR), which uses collected samples from the patient. This paper presents an efficient convolution neural network software implementation for COVID-19 and other pneumonia disease detection targeted for an AI-enabled smart biomedical diagnosis system (AIRBiS). From the evaluation results, we found that the classification accuracy of the abnormal (COVID-19 and pneumonia) test dataset is over 97.18%. On the other hand, the accuracy of the normal is no more than 71.37%. We discussed the possible problems and proposals for further optimization.

1 Introduction

Artificial intelligence (AI) is transforming modern medicine practice as it assists medical doctors in diagnosing patients more accurately. Recently, the number of people suffering from coronavirus disease 2019 (COVID-19) continues to increase worldwide. Therefore, the realization of a quick diagnosis system would allow medical doctors to treat patients accurately.

Reverse Transcription Polymerase Chain Reaction (RT-PCR) is the mainstream for COVID-19 testing. However, the RT-PCR test's accuracy is 60% to 90%, and the accuracy of diagnosis using lung X-ray images is 80%, from T. Ai, et. 2020 [1]. We consider that medical imaging contains a lot of information for diagnosing infection based on work in [1]. The Work in [1] reported that the accuracy of the results diagnosed by doctors from CT images was higher than that of PCR tests. The work in [2] is a basic study that challenged the learning of medical imaging. While the research in [3] raises doubts about the correctness of the evaluation method when learning rare and data-poor images such as COVID-19. Research work in [13, 14] proposed an architecture and preliminary prototyping results of a dependable real-time system for health monitoring. We proposed an AI-enabled Real-time Biomedical System (AIRBiS) [4] based on a deep neural network to detect pneumonia and COVID-19 from X-ray images of lungs in real-time. The ultimate goal is to get a high-precision neural network model to detect COVID-19 infection from X-ray images.

This paper presents an efficient convolution neural network software implementation for COVID-19 and other pneumonia disease detection. As a preliminary step, we classify the X-ray images into two labels: normal and abnormal/inflamed. The neural network model detects inflammation from X-ray images of the patient's lungs. We conducted experiments with simple LeNet-based three convolution layers models and VGG-16 [6], MobileNet V1 [7] and InceptionResNet V2 [8], and compared the accuracy of their classification result.

2 Convolution Neural Network Overview

Convolution Neural Network (CNN) is a class of deep neural networks. CNN includes a feature extraction part and a classification part shown in Fig. 1. The feature extraction part is composed of convolutional layers, pooling layers, and activation layers. The classifier part classifies the extracted features into output labels using fully connected layers. Our laboratory optimized CNN by accelerating using FPGA [5]. CNN is widely used in image classification and speech recognition.

This section describes four CNN models (LeNet, VGG-16, MobileNet, and InceptionResNet V2), which we investigated in this study. A simple model based on LeNet [9] has been created. The simple CNN model has three convolution and pooling layers, so we call it LeNet3 net. VGG-16 [6] is a CNN composed of 16 layers, including convolutional layers and max-pooling layers. MobileNet [7] using Depthwise Separable Convolution is a lighter network than ordinary CNNs. InceptionResNet V2 [8] is a network model that combines multi-layered use with the Inception module

*e-mail: m5241120@u-aizu.ac.jp

**e-mail: d8222108@u-aizu.ac.jp

***e-mail: s1250250@u-aizu.ac.jp

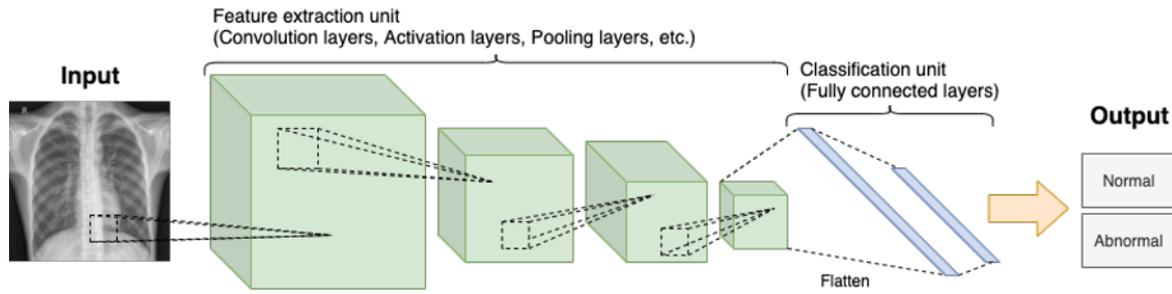


Figure 1: AIRBiS Software System with Convolution Neural Network.

and prevention of gradient disappearance and divergence by the residual learning (shortcut connection).

3 Dataset Preparation

We extract our dataset from two public datasets. The COVID-19 data collection [10, 11] is a dataset of lungs diseases X-ray images published for educational purposes. There are 478 images of COVID-19 extracted from the published GitHub code. Kaggle dataset [12] is used in the competition of classification pneumonia x-ray images, launched by Dr. Paul in 2017. It has 1583 normal lung images and 4273 pneumonia lung images. The specific number of data used in experiment is shown in Table 1 and 2. There are two labels, abnormal and normal, consist of three types of image data, COVID-19, Pneumonia, and normal.

Table 1: Dataset for training

Label	Training	Validation
Abnormal (COVID-19) (Pneumonia)	4243 (C: 400) (P: 3843)	50 (C: 10) (P: 40)
Normal	3897	50

Table 2: Dataset for testing

Label	Testing
Abnormal	458 (C: 68, P: 390)
Normal	234

Preprocessing and Augmentation

The COVID-19 images were a mixture of PNG and JPEG. So firstly, the PNG images were converted to JPEG. The preprocessing code is shown in Listing 1. The Kaggle dataset contained 1341 images of normal lungs. But this is less than an image with an abnormal label. Therefore, we tripled the normal image to almost the same number as the image with the abnormal label. The data has augmented with zooming and rotation, shown in Listing 2.

Listing 1: The code of data pre-processing (Python)

```
def preprocess (input_path , flag_delete_original_files = True):
    filepath_list = glob.glob(input_path + '/*.png')
    for filepath in filepath_list :
        basename = os.path.basename( filepath )
        save_filepath = out_path + '/' + basename[:-4] + '.jpeg'
        img = Image.open( filepath )
        img = img.convert( 'RGB' ) # RGBA(png)->RGB(jpg)
        img.save( save_filepath , "JPEG" , quality=95)
        if flag_delete_original_files :
            os.remove( filepath )
```

Listing 2: The code of data augmentation (Python)

```
for i, file in enumerate(files):
    img = load_img( file )
    x = img_to_array( img )
    x = np.expand_dims( x , axis=0 )
    # ImageDataGenerator
    datagen = ImageDataGenerator(
        zoom_range=0.2, # lower, upper = [1-0.2, 1+0.2]
        rotation_range=5 # [-5,5]
    )
    # Creating new two images
    g = datagen.flow( x , batch_size=1, save_to_dir = output_dir ,
        save_prefix = 'aug_' + str(i) , save_format = 'jpeg' )
    for __ in range(2):
        batch = g.next()
```

4 Evaluation

4.1 Comparison of accuracy with four CNNs

We trained models and tested them with the test dataset. The input images have been resized to (224,224,3) because it was used as a fixed value in the VGG-16 paper [6]. The Adam optimizer was used to train all models. In the table, the abnormal column shows the number of images predicted to be abnormal, and the normal column indicates the number of images predicted to be normal. The precision column shows the accuracy of the correct prediction.

The learning models evaluation result in tables 3, 5 and 6 show the low normal label prediction accuracy, VGG-16 in Table 4 predicted that all test images were Abnormal,

so the binary classification performing unsatisfactory currently.

Table 3: LeNet3 net using an adam optimizer

Label	Abnormal	Normal	Precision (%)
Abnormal	450 (C: 66) (P: 384)	8 (C :2) (P :6)	98.25 (C: 97.06) (P: 98.46)
Normal	83	151	64.53

Table 4: VGG-16 using an adam optimizer

Label	Abnormal	Normal	Precision (%)
Abnormal	458 (C: 68) (P: 390)	0 (C: 0) (P: 0)	100.00 (C: 100.00) (P: 100.00)
Normal	234	0	0.00

Table 5: MobileNet V1 using an adam optimizer

Label	Abnormal	Normal	Precision (%)
Abnormal	447 (C: 68) (P: 379)	11 (C: 0) (P: 11)	97.60 (C: 100.00) (P: 97.18)
Normal	67	168	71.37

Table 6: Inception-ResNet V2 using an adam optimizer

Label	Abnormal	Normal	Precision (%)
Abnormal	455 (C: 68) (P: 387)	3 (C: 0) (P: 3)	99.34 (C: 100.00) (P: 99.23)
Normal	77	157	67.09

4.1.1 Experiments with SGD optimizer

For VGG16, when the optimizer was Adam, all the results were Abnormal, and it could not be classified correctly. Therefore, we changed the optimized r to stochastic gradient descent (SGD) and experimented. SGD's Parameters: lr = 0.001, momentum = 0.9. As shown in table 7, the results of accuracy are close to that of other learning models.

4.2 Real-time augmentation

We augmented the dataset in real-time to improve the base of accuracy. Since all the training data can be randomly generated for each epoch, it can be said that if 50 epochs train 8140 training images, we can get 407,000 images.

Table 7: VGG-16 using a SGD optimizer

label	Abnormal	Normal	Precision (%)
Abnormal	451 (C: 68) (P: 383)	7 (C: 0) (P: 7)	98.47 (C: 100.00) (P: 98.21)
Normal	67	167	71.37

However, since the random value may be the same as the original image, it is less than 407,000. We implemented this method as shown in the listing 3. In the code, x_train contains an image for learning. Line 3 sets the augmentation method. Line 6 shows that a random augmentation is applied to the training image. Line 8 is a function to learn, and a learning image to be augmented at random is specified in its argument.

The graph of the experimental results is shown in the figure 2. The result was 94.36% with real-time augmentation. The normal label improved significantly from 64% to 86%, whereas the accuracy of abnormal labels was the same 98.25%. This experiment was trained by 50 epochs using the LeNet3 model.

Listing 3: The code of real-time augmentation

```

1 from tensorflow.keras.preprocessing.image import
  ImageDataGenerator
2
3 datagen = ImageDataGenerator(
4     zoom_range=0.2, shear_range = 0.2
5 )
6 datagen.fit(x_train)
7
8 history = try_model.fit(
9     datagen.flow(x_train, y_train, batch_size=batchsize),
10    steps_per_epoch=len(y_train)/batchsize,
11    validation_data=(x_val, y_val),
12    epochs=50)
    
```

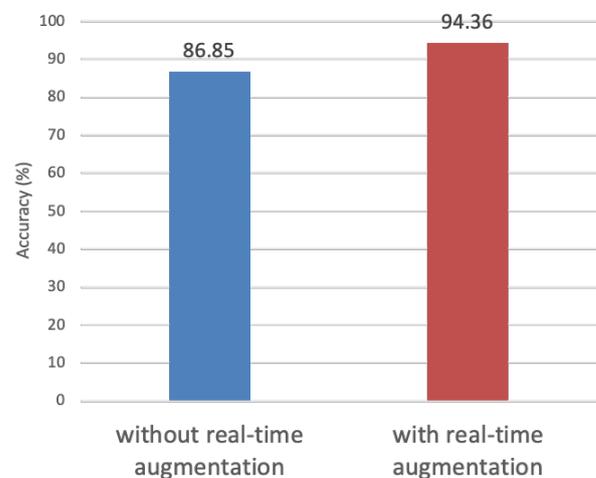


Figure 2: Comparison of accuracy of learning with/without real-time augmentation

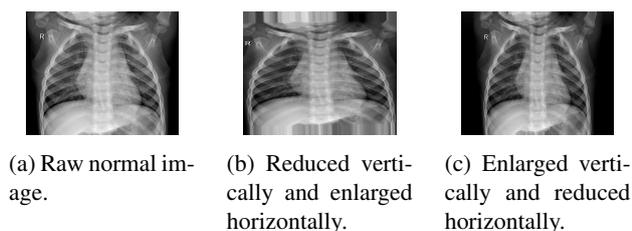


Figure 3: The results of the zoom augmentation operation

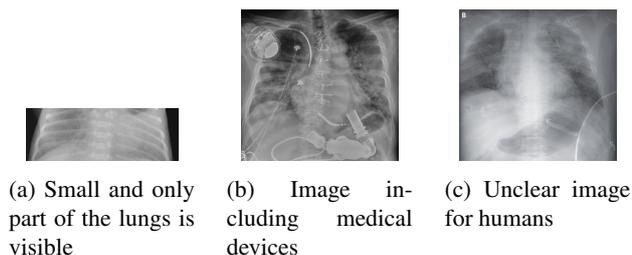


Figure 4: X-ray images containing noise from wires and medical equipment, from COVID-19 label.

5 Discussion

The experiment results show approximately 20% reduces the accuracy of normal X-ray images compared to that of abnormal. We consider some possible reasons for the significant accuracy drop in normal X-ray images.

The first is the variety in image quality. The image size in the dataset before resizing varies from 6KB to 8304KB. Images with extensive data can drop some image features when resized to (224,224,3). However, Many small pictures (e.g., 6KB) do not have enough pixels for training and test, as shown in Figure 4a.

Secondly, the aspect ratio of the X-ray images is not preserved. This problem occurs during the resizing operation in the augmentation process, which randomly zooms in and out the vertical and horizontal fields. The X-ray image can be quite different if the original aspect ratio is not kept, as shown in figure 3. Therefore, we will implement other augmentation methods in future experiments. Similarly, the aspect ratio is altered when the X-ray images are resized to a square because many pictures are not square initially. To preserve the aspect ratio when resizing, we suggest a pre-processing of cropping the image into squares before resizing it.

The third problem is incomplete images and interference. There are some incomplete X-ray images in which only a part of the lung is visible, as shown in figure 4a, plenty of images come with interference, such as medical equipment as shown in figure 4b, and a number of images that are unclear in human view, as shown in figure 4c. In the future, we will investigate the effects of these factors.

6 Conclusion and Future Work

This paper presents a comprehensive study of coronavirus disease 2019 (COVID-19) classification based on deep convolution neural networks. From the study, we found that the binary classification was possible for all models except VGG-16 under the condition that the optimizer was set to Adam. The diagnosis accuracy of abnormal (COVID-19 and pneumonia) is 97.18 to 99.34% for the current dataset, which is outstanding in all experimented learning models. On the other hand, the correct classification accuracy of normal/healthy lung X-ray images is 67.09 to 71.37%, which is currently unsatisfactory. It shows that there is a false positive problem. After changing the optimizer of the VGG-16 model from Adam to SGD in an additional experiment, the training model was able to binary-classify with accuracy close to other training models. From this experiment, we can conclude that the best optimizer differs with different learning models.

Real-time augmentation increased the accuracy from 86.85% to 94.36%. This result showed that the accuracy was greatly improved by improving the dataset.

In the future, we will investigate the effects of inappropriate images in the dataset. Besides, we will examine training other CNN models using real-time augmentation and using other optimizers.

References

- [1] T. Ai, Z. Yang, H. Hou, C. Zhan, C. Chen, W. Lv, Q. Tao, Z. Sun, and L. Xia, Correlation of chest ct and rt-pcr testing in coronavirus disease 2019 (covid-19) in china: a report of 1014 cases, *Radiology*, p. 200642, 2020.
- [2] Li, Q., Cai, W., Wang, X., Zhou, Y., Feng, D. D., Chen, M. (2014, December). Medical image classification with convolutional neural network. In 2014 13th International Conference on Control Automation Robotics Vision (ICARCV) (pp. 844-848). IEEE.
- [3] Gianluca Maguolo and Loris Nanni. A critic evaluation of methods for covid-19 automatic detection from x-ray images, 2020.
- [4] Abderazek Ben Abdallah, Huankun Huang, Nam Khanh Dang, and Jiangning Song. Ai processor, Nov. 2020. Japanese Patent Application Laid-Open No 2020-194733.
- [5] The H Vu, Ryunosuke Murakami, Yuichi Okuyama, and Abderazek Ben Abdallah. Efficient optimization and hardware acceleration of cnns towards the design of a scalable neuro inspired architecture in hardware. In *2018 IEEE International Conference on Big Data and Smart Computing (BigComp)*, pages 326–332. IEEE, 2018.
- [6] Simonyan, Karen and Zisserman, Andrew. Very deep convolutional networks for large-scale image recognition, *arXiv 1409.1556*, 2014.
- [7] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Inception-v4,

- inception-resnet and the impact of residual connections on learning. *arXiv preprint arXiv:1704.04861*, 2017.
- [8] Szegedy, Christian and Ioffe, Sergey and Vanhoucke, Vincent and Alemi, Alex. Inception-v4, inception-resnet and the impact of residual connections on learning. *arXiv preprint arXiv:1602.07261*, 2016.
- [9] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.
- [10] Cohen, Joseph Paul and Morrison, Paul and Dao, Lan Covid-19 image data collection. *arXiv 2003.11597*, 2020.
- [11] Cohen, Joseph Paul and Morrison, Paul and Dao, Lan and Roth, Karsten and Duong, Tim Q and Ghassemi, Marzyeh. Covid-19 image data collection: Prospective predictions are the future. *arXiv 2006.11988*, 2020.
- [12] Paul Mooney. Chest X-Ray Images (Pneumonia). <https://www.kaggle.com/paultimothymooney/chest-xray-pneumonia>, 2020.
- [13] A. Ben Ahmed, Y. Kimezawa and A. Ben Abdallah, "Hardware/software prototyping of dependable real-time system for elderly health monitoring," 2013 World Congress on Computer and Information Technology (WCCIT), Sousse, 2013, pp. 1-6, doi: 10.1109/WCCIT.2013.6618696.
- [14] A. Ben Ahmed, Y. Kimezawa and A. Ben Abdallah, "Towards smart health monitoring system for elderly people," 4th International Conference on Awareness Science and Technology, Seoul, 2012, pp. 248-253, doi: 10.1109/iCAwST.2012.6469622.