A Robust End-to-End CNN Architecture for Efficient COVID-19 Prediction form X-ray Images with Imbalanced Data

Zakariya A. Oraibi[1,*], Safaa Albasri[2]

[1]Department of Computer Science, College of Education for Pure Sciences, Unviersity of Basrah, Basrah, Iraq
[2]Electrical Engineering Department, College of Engineering, Mustansiriyah University, Baghdad, Iraq
E-mail: zakaria_au@uobasrah.edu.iq, asafaa@uomustansiriyah.edu.iq
[*]Corresponding author

Keywords: COVID-19, Deep Learning, Coronavirus Infection, Data Augmentation, Pandemic

Received: April 10, 2023

The world witnessed big changes in 2019 when a new virus called coronavirus affected the lives of hundreds of millions of individuals and led to huge disruptions in healthcare systems. Early prediction of this virus was a top priority to limit its damage and save countless lives. Many advanced artificial intelligence technologies like deep learning have used chest X-ray images for this task. In this paper, a new CNN architecture is introduced to classify chest X-ray images. The new model is applied on a $256 \times 256 \times 3$ *input image and consists of six convolutional blocks. In addition, we improve the performance of our model by adding regularization techniques, including batch normalization and dropout. We tested our model using an imbalanced COVID-19 dataset of 5000 COVID and Non-COVID images. Four metrics were used to test the new model: sensitivity, specificity, precision, and F1 score. In experiments, we achieved a sensitivity rate of 97%, a specificity rate of 99.32%, a precision rate of 99.90%, and F1 score of 97.73% despite being provided with fewer training images. In conclusion, we proposed a light deep learning model capable of achieving high prediction accuracy that outperformed the best deep learning methods in terms of specificity and achieved high sensitivity result.*

Povzetek: Uporabljena je nova metoda z arhitekturo CNN za zaznavanje koronavirusa iz rentgentskih slik pljuč.

1 Introduction

In late 2019, a deadly variant of coronavirus (CoV) family was discovered in Wuhan, China named SARS-CoV-2. This new coronavirus member (COVID-19) caused a global pandemic and was responsible for millions of infections and deaths among people [2]. World Health Organization (WHO) announced this virus as a global pandemic in March, 2020 [3]. As of March 16, 2023, there were 760,360,956 confirmed COVID-19 cases and 6,873,477 deaths around the world [1] and numbers keep rising every day. As a result, new measures for people interaction have been imposed globally and lockdown has also been enforced to limit the effect of this virus [4]. It is also worthy to mention that the new COVID-19 vaccines rolled out in 2021 helped significantly to curb the numbers of the new infections and deaths.

The reason for the widespread of the virus among people is because it is highly transmissible respiratory infection that causes breathing difficulty which could lead to lung failure and causes death. Due to this, people are forced to keep a distance between each other as well as avoid direct contact [5]. So, a good solution to prevent the spread of the virus is to isolate people. The main symptoms of COVID-19 include fever, headache, and cough. In addition, other symptoms could also arise to some individuals. Early detection of this virus can help separate patients from normal individuals and prevent the spread of this contagious disease [6].

A successful technique to diagnose COVID-19 is by employing medical image processing methods on chest X-ray images. The advantage of using chest X-ray images include the availability of these images in many diagnostic centers, portability, and they are easily accessible [7]. So, methods that rely on chest X-ray images are feasible to detect abnormalities in lung and diagnose COVID-19. Deep learning techniques for image classification and detection showed huge potential in biomedical imaging analysis [8]. In addition, the use of deep learning methods made it possible to analyze large COVID-19 datasets with thousands of images without relying heavily on radiologists to achieve this tedious task and without consuming much time [9].

Researchers have thoroughly studied and analyzed COVID-19 infection from chest X-ray images using different CNN techniques. Many approaches have been proposed to tackle this problem. A new CNN introduced by Abbast et al. [10] named DeTraC to predict COVID-19 cases. A

[1]https://covid19.who.int/

Table 1: Summary of related work.

Technique	Year	Dataset	Performance
Pre-trained CNN models [1]	2020	COVID-Xray-5k	Specificity = 97.0% , Sensitivity = 96.7%
DeTraC [10]	2021	Collection of images	Accuracy = 93.1% , Sensitivity = 100%
PDCOVIDNet [11]	2020	2905 chest X-ray images	Accuracy = 96.58%
38-layers CNN [12]	2021	Independent dataset	Accuracy = 99.5%
Optimized CNN [13]	2021	6432 chest X-ray images	Accuracy = 96.0%
Pre-trained CNN models [14]	2021	Collection of CT scan imagges	Accuracy = 99.4%
Transfer learning framework [14]	2020	622 image samples	Accuracy = 97.95% , Specificity = 98.85%
Transfer learning with Xception [16]	2020	1102 chest X-ray images	Sensitivity = 99.27% , Specificity = 99.34%
Ensemble of pre-trained models [18]	2021	2326 X-ray images	Sensitivity = 90.5% , Specificity = 90.0%

high accuracy of 93.1% was achieved using their approach. However, this method requires two training stages. In [11], Chowdhury et al. employed chest X-ray images and proposed a PDCOVIDNet architecture which relies on parallel dilated CNN. Authors were able to capture the important features to produce an accuracy of 96.58%. Reshi et al. [12] proposed a CNN model with 38 layers which include 6 convolutional layers to classify COVID-19 images. Their model is applied to an input image size of $150\times150\times3$. The drawback is this model works on a relatively small batch size. They achieved a high accuracy of 99.5%. In [13], Pathan et al. proposed an optimized CNN framework to classify COVID-19 images by employing Grey Wolf Optimizer (GWO) to optimize the parameters of the model. A classification accuracy of 98.8% was obtained.

Deep transfer learning models have also been proposed to classify COVID-19 images. Ahuja et al. [14] used four pre-trained models, ResNet18, ResNet50, ResNet101, and SqueezeNet, and achieved 99.4% testing accuracy. El-Rashidy et al. [15] introduced an end-to-end CNN framework that includes monitoring the patient in real time using cloud and transfer learning. Their model achieved a sensitivity rate of 98.85%. A transfer learning model proposed by Wang et al. [16] has shown 96.75% classification accuracy using Xception model. Furthermore, authors improved their accuracy by fusing deep feature with SVM classifier. However, this method requires a two stage training which consumes much more time. In [17], Khan et al. proposed to use multiple CNN architectures: ResNet121, ResNet50, VGG16, and VGG19, with transfer learning for the prediction of COVID images. The overall accuracy generated by their work was 99.3%. A deep learning classifier that uses an ensemble of pre-trained CNNs was used by Keidar et al. [18]. These models include ReNet34, ReNet50¸ ReNet152, and Vgg16. This ensemble approach was applied on a dataset of 2326 COVID images and achieved a high accuracy of 90.3%. Table 1 summarizes the aforementioned techniques.

From the literature, many of the methods used to predict COVID-19 cases used chest X-ray images. This shows the importance of employing chest X-ray images as an analysis tool for radiologists. There is still a problem to overcome which is the imbalanced number of images provided for COVID and Non-COVID cases. In our work, we overcome this issue by using data augmentation and a new proposed efficient deep learning architecture to enhance the classification accuracy of COVID-19 images.

In this paper, a new CNN architecture trained from scratch is proposed to classify COVID-19 images. This new architecture employs six convolutional blocks with each block includes one convolutional layer, one ReLU layer, and one max-pooling layer. To further improve the performance of our architecture and reduce overfitting, we also added dropout and batch normalization. All input images must be resized to $256 \times 256 \times 3$. The proposed model was evaluated on a challenging COVID-19 dataset with imbalanced number of samples for both COVID and Non-COVID classes. It is good to mention that in early 2020 there were few datasets available for COVID-19 classification from chest X-ray images. Since then, many datasets were introduced but most of them are still a collection of other datasets. One dataset which was ensembled by [19] and made available for COVID-19 prediction purposes is considered a challenging one due to having imbalanced number of images for both COVID and Non-COVID classes. The total number of images available in this dataset is small in relative to other datasets. Authors in [1] who collected the dataset used it to relabel all images and combine the newly labeled images with another dataset to create a novel set of images with around 5000 images called COVID-Xray-5k which we are using in the current work.

The remainder sections of this paper are as follows: Sec-

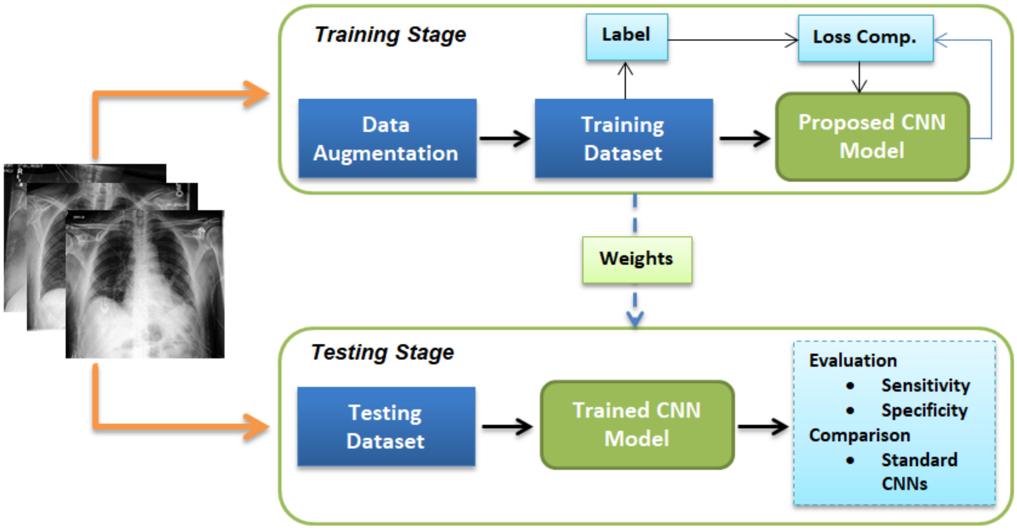

Figure 1: Pipeline of COVID-19 classification. The proposed architecture is fed with the training images to generate the weights. then, weights are employed to predict the testing images.

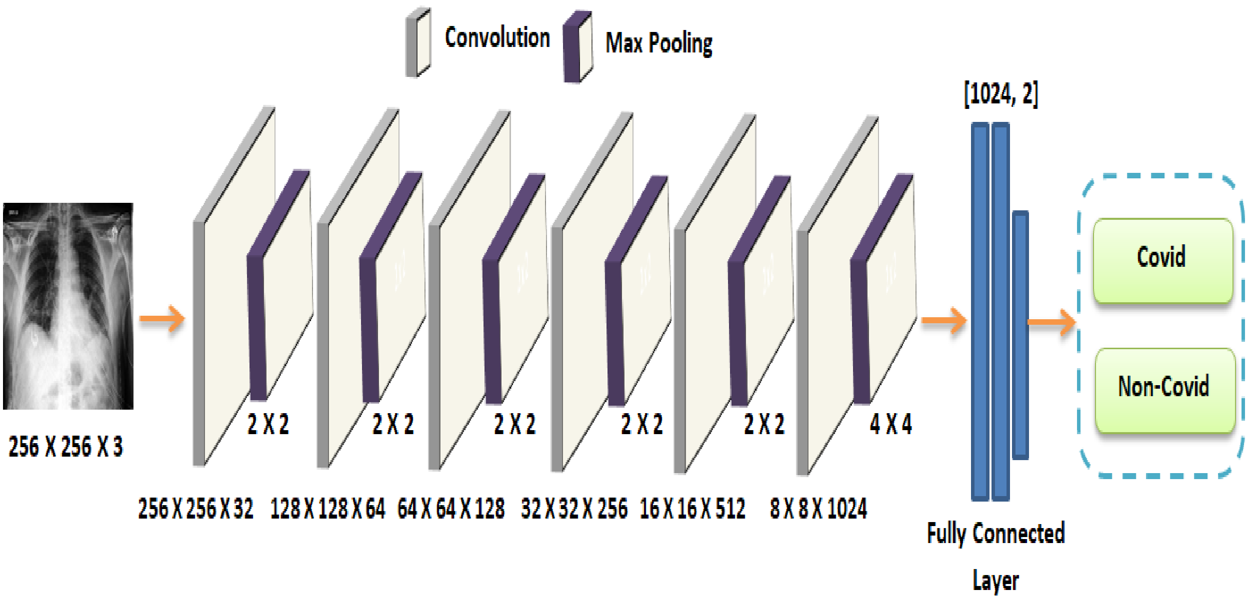

Figure 2: Blocks of the proposed deep learning architecture.

tion 2 describes the new architecture. Section 3 describes in details the dataset used in our work. Section 4 introduces the results of the new approach. Section 5 discusses the results and finally, we present conclusions and future challenges in section 6.

2 Methodology

Our deep learning architecture that will be used to analyze COVID-19 images is presented in this section. Figure 1 illustrates the steps of the classification task. First, the dataset of COVID-19 images is divided into two subsets, training and testing. In the training stage, COVID images are further divided into training and validation subsets in order to evaluate the training stage. Data augmentation is also applied on the original dataset images to balance the samples of both COVID and Non-COVID classes. After the proposed model is trained, the weights are used in the testing stage where each image from the testing subset is classified to either COVID or Non-COVID. The layers and specifications of the proposed architecture are described in detail in the following subsections.

2.1 The proposed architecture

The pattern recognition technique used in our paper to exploit the patterns of the COVID-19 images is based on deep convolutional neural network. The reason to choose deep learning method for this image analysis task is because it proved to be so powerful in classifying images and learning features [20, 21, 22]. As a result, we built a deep learning network and trained it to predict COVID-19 patients from chest X-ray images. The proposed CNN architecture is presented in Figure 2.

Our end-to-end CNN model is based on creating convolutional blocks to exploit the detailed COVID-19 image patterns. Architectures proposed before including the work of Khan et al. [23] employed four convolutional blocks with ten convolutional layers. Authors claim that their model improves accuracy by exploiting edge-based operations to better examine the textural and structural variations of an image along with the boundary-related information. In addition, authors also used varying number of convolutioanl layers in each block.

The new CNN architecture employs six convolutional blocks as shown in Figure 2. Each block comprises of one convolutional layer followed by a ReLU as an activation function and max-pooling. By using six convolutional blocks, COVID image features are extracted efficiently. Fully connected layer is added at the end to perform the classification task. In addition, our model prevents overfitting while training because batch normalization and dropout layers are added to it. The design of our architecture is more powerful in terms of using six convolutional blocks instead of four as in [23]. The benefit of designing a deep CNN is to allow the model to efficiently learn complex features from the input COVID image. Hence, the model generates more convolutioanl maps. On the other hand, Reshi et al. [12] proposed a deep learning model with six convolutional layers, however, their CNN works on a small input image size of $150 \times 150 \times 3$ compared to our model that works on an input image size of $256 \times 256 \times 3$.

$$f_{m,n} = \sum_{a=1}^{L}\sum_{b=1}^{D} f_{m+a,n+b} K_{a,b} \quad (1)$$

Equation 1 computes the convolutional operation with M and N are the dimensions of the input feature map f. K is a kernel of size $(L \times D)$.

$$f_{m,n}^{max} = max_{a=1,..,wz,b=1,..,wz} f_{m+a,n+b} \quad (2)$$

Equation 2 computes the max-pooling operations with a window size of wz.

2.2 Details of the proposed model

The first convolutional block consists of a single convolutional layer that takes the image that has 32 kernels of size 3×3 with padding one and stride one. After the convolution operation is done on the resized image of $256 \times 256 \times 3$ dimension, a ReLU activation function is applied. Then, a max-pooling is applied with a stride of 2 and a 2×2 filter is also used. After that, the second convolutional block also consists of a single convolutional layer that takes the input image shape from the previous layer with $128 \times 128 \times 32$ dimensions and apply the convolution process with 64 filters this time. ReLU and max-pooling with a stride of 2 and a filter size of 2×2 is also utilized bringing the output shape of the image after the second convolutional layer to be $64 \times 64 \times 64$. The process continues for blocks 3, 4, 5, and 6 with each one consists of one convolutional layer, one ReLU, and one max-pooling as previously discussed. The sixth block has a max-pooling with a filter of 4×4 and final image size is $8 \times 8 \times 1024$ as shown in Figure 2. Finally, a flatten layer and a linear layer was used to perform the classification task.

3 Materials

3.1 Dataset

The new CNN model described in the previous section was evaluated on a challenging dataset called COVID-Xray-5k [1]. This dataset as mentioned earlier was collected by Minaee et al. for COVID-19 researches. Images of the dataset are a collection of Computed Tomography (CT) and X-ray images. In order to get the full dataset, authors derived COVID-19 samples from another set of images called Covid-Chest-Dataset [19]. Covid-Chest-Dataset is updated frequently and consists of additional information including patients' age and sex. Hense, radiology specialists in [1] preserved anterior-posterior samples for the detection of COVID-19 infections. Finally, Radiologists have approved

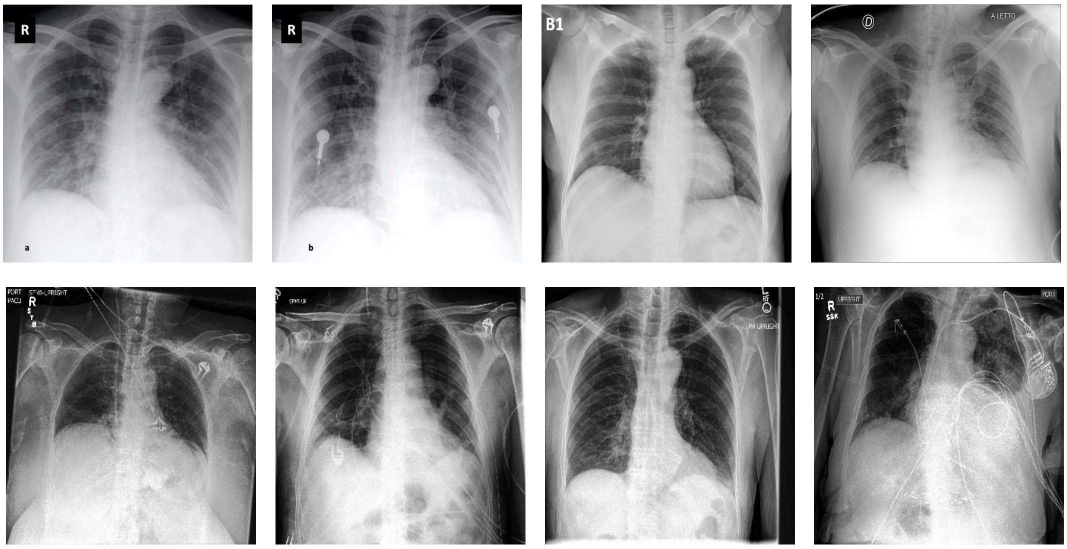

Figure 3: Sample images from COVID-Xray-5k dataset. The four samples in row 1 are COVID-19 class images. The other four samples in row 2 are Non-COVID class images. Images from both classes have been collected and validated by radiologists from two datasets.

only 184 images of COVID-19 after being thoroughly examined.

In regard to Non-COVID image samples, Minaee et al. used a set of images form another dataset called Chex-Pert dataset [24]. This dataset has huge number of X-ray samples collected from tens of thousands of patients. Only 2,000 Non-COVID images were chosen for the training purpose of machine learning algorithms and another 3,000 images for the purpose of testing as shown in Table 2. Finally, 5000 images are provided in total for the Non-COVID class. Sample images of both classes are shown in Figure 3.

Table 2: Training and testing samples of COVID-Xray-5k dataset. Fewer number of training images are provided.

Image Subsets	COVID	Non-COVID
Training Subset	84	2000
Testing Subset	100	3000

Images of this dataset vary in resolution. Some of them have low resolution of 400×400 pixels. While others have high resolution of 1900×1400 pixels.

3.2 Data augmentation

Due to the insufficient data available in training for the COVID class, deep learning models can easily overfit causing an increase in the loss function during the training stage. To prevent overfitting while training the deep learning model and to provide better prediction accuracy during testing, data augmentation is used by increasing the number of original data [25]. In general, we are trying to prevent our model from biasing towards Non-COVID class during training due to the huge gap between the number of images of the two classes. Augmentation has become an important tool in deep learning techniques to solve the issues related with insufficient amount of data and to provide different views of the same image to enhance model training.

In our work, both classes of the dataset were augmented by performing random rotation (0 - 45 degrees), shifting image width and height by 20% each, shearing the image by 20%, and zooming by 20%. In addition, image flipping horizontally was also applied. In total, we increased the images in the COVID and Non-COVID classes to be 2134 and 3480, respectively as shown in Table 3 In the implementation, black boundaries resulted from applying these operations were filled with a reflection of the original image pixels. The strategies used for image augmentation were helpful to add hundreds of samples to the original training set, making our model effective to be applied on the dataset.

Table 3: Total number of COVID and Non-COVID training images after augmentation.

Class	Original	Augmented
COVID-19	84	2134
Non-COVID	2000	3480

4 Results

This section demonstrates the results of applying our proposed deep learning architecture to classify COVID-19 images. In addition, a comparison with the state-of-the-art

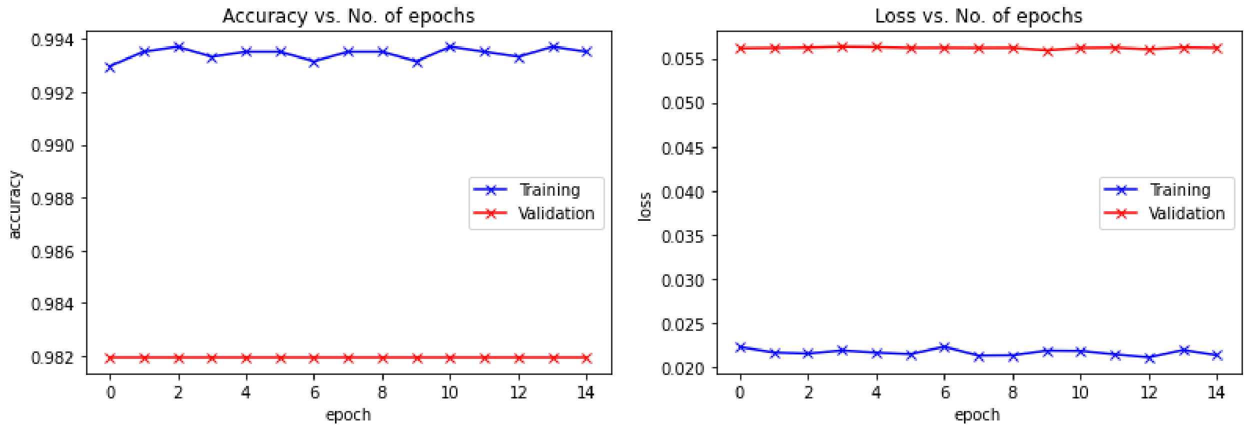

Figure 4: Training results after applying the proposed model without using regularization techniques. The graph on the left represents the training accuracy after 15 epochs. Training loss curve vs validation loss curve are represented by the graph on the right.

approaches on the same dataset is also provided. Performance evaluation was done using four metrics: sensitivity, specificity, precision, and F1 score. Each of these metrics is calculated using the corresponding equation below:

$$Sensitivity = \frac{\sum TN}{\sum FP + \sum TN} \tag{3}$$

$$Specificity = \frac{\sum TP}{\sum TP + \sum FN} \tag{4}$$

$$F1\ Score = \frac{2\sum TP}{2\sum TP + FP + FN} \tag{5}$$

$$Precision = \frac{\sum TP}{\sum TP + \sum FP} \tag{6}$$

The newly introduced CNN model is implemented using PyTorch software. In order to train our model, we used Adam optimizer and SGD optimizer to determine the best one among them. A batch size of 128 was also employed to properly train our model and to speedup the training process. The number of epochs used during training was 15. Furthermore, two learning rates were used: 0.001 and 0.0001. Cross-entropy loss function was used in the implementation as in Equation. 7 to evaluate the performance of the classification task. The implementation of the proposed CNN model and the experiments are all done on the Google Colaboratory platform with NVIDIA T4 Tensor Core GPU.

$$Loss_{CE} = -\sum_{i=1}^{n} t_i \log(p_i) \tag{7}$$

where t_i is the truth label and p_i is the softmax probability for the i_{th} class.

In order to check the robustness of our new model, experiments were divided into two sets. The first set of experiments deal with evaluating our model without any regularization techniques. Regularization techniques involve using batch normalization in each convolutioanl block and dropout. The second set of experiments deal with evaluating the model after adding regularization techniques.

4.1 Performance evaluation without regularization

We begin evaluating our model by using the pure proposed architecture blocks without any regularization techniques (image normalization and dropout) to check the behaviour of convolutioanl layers and feature maps to see if the performance is superior or if overfit prevails. In the experiments, we used 15 epochs to train the network since the model may overfit if the number of epochs used while training is increased. The graph on left in Figure 4 demonstrates the training accuracy during the training stage after 15 epochs. The accuracy as we can see keeps going up and down because the validation loss is greater than training loss. Since training and validation losses diverge hugely as in the graph on right in Figure 4, overfit is happening and regularization techniques are needed to fix this issue.

Table 4 shows the sensitivity, specificity, F1 score, and precision rates on the test data. As we can see, the proposed model without regularization is performing good in terms of sensitivity scoring 95% when Adam optimizer is used and the learning rate is 0.001. However, the model performs poorly in terms of specificity achieving 88.16% when SGD optimizer is used with a learning rate of 0.001. Hence, regularization is needed to reduce the overfit and to further improve the classification accuracy. F1 score and precision rates are very high scoring 93.62% and 99.81% respectively using SGD optimizer with learning rate of 0.001.

Figure 5 presents the confusion matrix using SGD optimizer with 0.001 learning rate. As we can see, only five samples are mis-classified as Non-COVID while 355 samples are mis-classified as COVID. There is a potential for improvement especially with Non-COVID class since hun-

Table 4: Model performance with Adam and SGD optimizers. Results are reported without using regularization techniques.

Optimizer	Learning Rate	Sensitivity	Specificity	F1 score	Precision
Adam	0.001	95.0%	86.31%	93.42%	99.71%
Adam	0.0001	96.0%	84.1%	94.13%	98.95%
SGD	0.001	95.0%	88.16%	93.62%	99.81%
SGD	0.0001	96.0%	84.53%	94.21%	98.88%

dreds of samples are mistakenly classified.

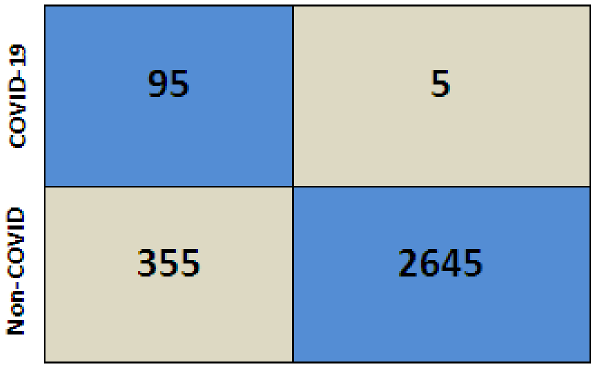

Figure 5: Confusion matrix for results without regularization. The optimizer used is SGD with learning rate of 0.001.

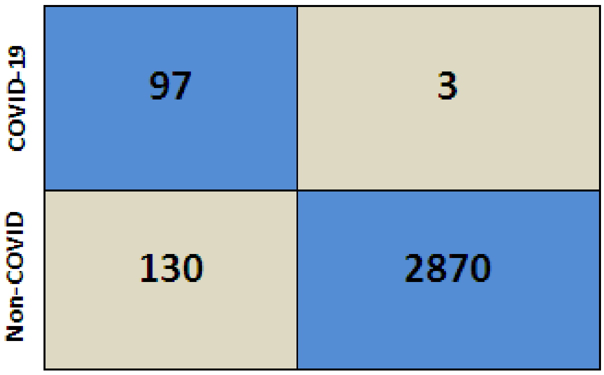

Figure 6: Confusion matrix for results with regularization. The optimizer applied during training is SGD with learning rate of 0.001.

4.2 Performance evaluation with regularization

Regularization techniques, including batch normalization and dropout, are very important to prevent model fluctuation and overfitting. The experiments conducted previously showed the potential of our six blocks architecture in producing good prediction results. However, since the model was not trained effectively due to the lack of batch normalization and dropout, it is possible to enhance model training and improve classification results. In the implementation, we added batch normalization in each convolutional block. In addition, dropout was used to the fully connected layer with a rate of 0.1 which means 10% of the inputs will be excluded during each update cycle.

In order to reduce overfitting and allow the training accuracy to converge faster with no fluctuation, we used batch normalization and dropout to the new architecture. The graph on left of Figure 7 shows the training accuracy vs validation accuracy while the graph on right shows that the training and validation losses converge close to 0 after applying the regularization techniques. Table 5 demonstrates the results of the experiments using different optimizers and different learning rates. The number of epochs used in these experiments has also been set to 15. We can observe that results have improved for both classes especially for the Non-COVID class. Adam optimizer with 0.001 learning rate achieved a 95.0% sensitivity and 99.32% specificity outperforming all results set previously without using regularization techniques. However, SGD optimizer set the best result for sensitivity, 97.0% and the specificity rate of 95.67% is slightly less than ADAM optimizer with a learning rate of 0.001. In addition, F1 score and precision were very high achieving 97.73% and 99.90% respectively. Figure 6 demonstrates the confusion matrix resulted from applying our model using SGD optimizer with 0.001 learning rate. Only three COVID samples were mis-classified as COVID and only 130 Non-COVID samples were misclassified as COVID.

It is worthy to mention that SGD can be locally unstable in some cases but, it can outperform Adam optimizer in terms of generalization performance. Adam optimizer produced superior specificity results compared to SGD which produced the best sensitivity result. In general, we can observe that adding regularization techniques to our proposed deep learning model helped in reducing overfitting and improving the classification performance. Both sensitivity and specificity metrics improved while using the same hyper parameters in training.

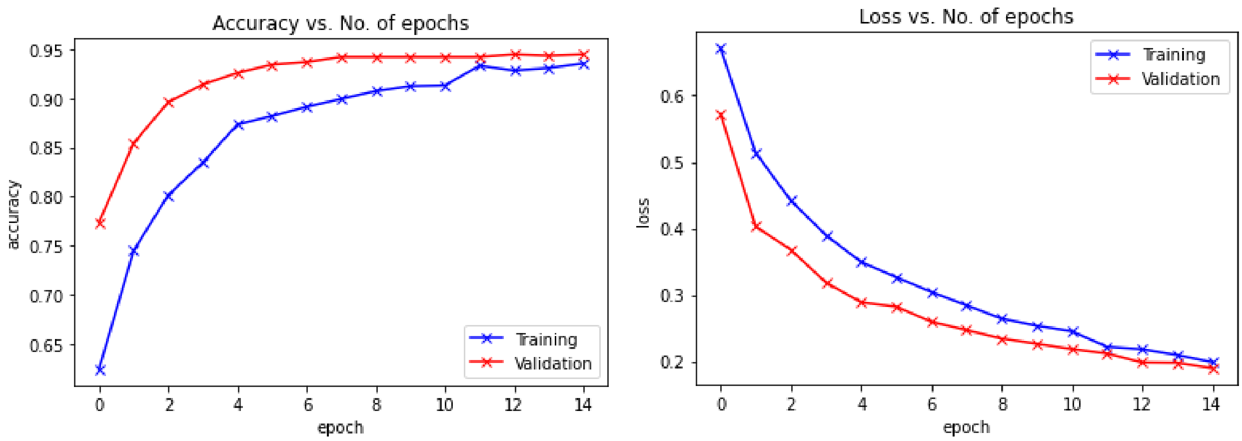

Figure 7: The graph on the left is the training accuracy vs validation accuracy after applying the regularization technique to our model, while the graph on the the right is the training loss vs the validation loss.

Table 5: Model performance after applying our proposed deep learning architecture with batch normalization and dropout.

Optimizer	Learning Rate	Sensitivity	Specificity	F1 score	Precision
Adam	0.001	95.0%	99.32%	98.05%	99.29%
Adam	0.0001	95.0%	99.13%	98.35%	99.31%
SGD	0.001	97.0%	95.67%	97.73%	99.90%
SGD	0.0001	97.0%	87.34%	95.41%	97.27%

4.3 Comparison with state-of-the-art CNN approaches

In the experiments, we applied our proposed model on a standard dataset which was published by Minaee et al. [1]. In their work, authors applied several pre-trained deep learning models with fine tuning and different thresholding values for each model. Results of four standard deep learning models with our proposed model are reported in Table 6.We can observe that ResNet18 and ResNet50 produced good sensitivity and specificity rates. On the other hand, SqueezNet produced the best sensitivity rate of 98.0% with a specificity rate of only 92.9%. DenseNet-121 produced similar sensitivity rate to SqueezeNet (98.0%) but, the specificity rate was only 75.1% which is the lowest rate among the four pre-trained models.

Results of our proposed model using Adam optimizer and learning rate of 0.001 are the best in terms of specificity (99.3%) and in terms of sensitivity, we scored 95.0%. Using SGD optimizer, we managed to improve the sensitivity to become 97.0% with specificity reported to be 95.7%. These results reveal that our model outperformed all pre-trained models in terms of specificity and was only 1% lower than the best sensitivity result that is 98.0%. All our best results were generated after applying batch normalization and dropout to the model. This shows the robustness of regularization techniques in reducing overfitting and improving overall classification results.

5 Discussion

The prediction technique applied in this paper relies on a new deep learning architecture that uses few convolutional layers and produce high accuracy results. The new model was designed to accept images with $256 \times 256 \times 3$ in size in order to preserve the original shape of the human lung during the convolution process. The intuition behind using few layers per block is because the task involves predicting only two classes. As a result, the proposed model was designed to overcome overfitting during training by using one layer per convolutional block which proved to be sufficient to produce high training and validation accuracy. In addition, CNN parameters were selected to train the model properly and to reduce overfitting. We showed that 15 epochs with a learning rate of 0.001 produced the lowest training and validation loss and produced high specificity and sensitivity predictions. Furthermore, we showed that Adam optimizer outperformed SGD optimizer because it generalizes better than the latter. Results are reported for the two optimizers in section 4. The state-of-the-art approaches summarized in Table 1 relied mostly on transfer learning of existing CNN architectures like ResNet and VGG. Nevertheless, the work of Rishi et al. [12] trained a CNN architecture from scratch

Table 6: Evaluating the proposed model with different deep learning architectures.

Model	Sensitivity	Specificity
ResNet18 [26]	98.0%	90.7%
ResNet50 [26]	98.0%	89.6%
SqueezeNet [27]	**98.0**%	92.9%
Densenet-121 [28]	98.0%	75.1%
Proposed [Adam, 0.001]	95.0%	**99.32**%
Proposed [SGD, 0.001]	97.0%	95.67%

but the batch size was only $150 \times 150 \times 3$. In our approach, the model handles a bigger batch size of $256 \times 256 \times 3$. On the other hand, Ozturk et al. [29] presented a deep learning model with 19 layers which combines both DarkNet and You Only Look Once (YOLO) and produced a robust 97.0% sensitivity rate and 96.7% specificity rate applied on a dataset which was collected from two different sources for the purpose of COVID-19 detection.

One limitation we faced during the experiments is that the dataset used in this paper consists of imbalanced number of images for both COVID and Non-COVID classes which makes the problem of accurate classification even harder. To solve this issue, in section 3.2 we used an augmentation method to enlarge the dataset and preserve the shape of the lung in order to use the new images for the training stage. Furthermore, results were obtained before and after adding regularization to our architecture. The classification performance of our model was evaluated using both specificity and sensitivity. Results reported in section 4 proved that using a light deep learning architecture for COVID prediction task is sufficient to get a high classification accuracy. Therefore, we recommend to train new CNN architectures from scratch with a big image size and to use regularization techniques as a method to improve the training process of CNN and to allow more convergence of training and validation loss. In addition, we showed through experiments that light deep learning models trained from scratched could produce high classification accuracy in comparison to deep transfer learning models. Furthermore, Although SGD optimizer outperformed ADAM optimizer slightly in terms of sensitivity, Adam optimizer performance was in general better than SGD. This is because unlike SGD, ADAM requires fewer parameters for tuning and can handle sparse gradients very well.

6 Conclusion

This paper introduces a new CNN architecture to predict the class of the given X-ray images as either COVID or Non-COVID. The proposed model uses six convolutional blocks. Every block comprises of a single convolutional layer, ReLU layer, and max-pooling layer. Additionally, regularization techniques were included in the model to reduce overfitting and improve the classification rates. These techniques involve batch normalization which was added in each covolutional block and dropout. Experimental results performed on a challenging COVID-19 imbalanced dataset of 5000 image samples proved that our model achieved the best specificity rate compared to state-of-the-art pre-trained models performed on the same dataset. In addition, our model produced comparable results in terms of sensitivity falling 1% off the best result scored on the same dataset. Furthermore, F1 score and precision results are very robust using the new CNN architecture.

The Future work will focus on improving the model by adding more convolutional layers per block and perform a thorough evaluation using more epochs. Moreover, we plan to use the model on other publicly available COVID-19 datasets to validate its robustness. Optimal hyper parameters will be applied to generate better training results. Since the healthcare sector physicians in dire need for an automatic detection of COVID-19, the more accurate our model can achieve this task the highly likely it is employed in the sector.

References

[1] S. Minaee, R. Kafieh, M. Sonka, S. Yazdani, and G. J. Soufi, "Deep-covid: Predicting covid-19 from chest x-ray images using deep transfer learning," *arXiv preprint arXiv:2004.09363*, 2020.

[2] L. Pang, S. Liu, X. Zhang, T. Tian, and Z. Zhao, "Transmission dynamics and control strategies of covid-19 in wuhan, china," *Journal of Biological Systems*, vol. 28, no. 03, pp. 543–560, 2020.

[3] D. Cucinotta and M. Vanelli, "Who declares covid-19 a pandemic," *Acta Bio Medica: Atenei Parmensis*, vol. 91, no. 1, p. 157, 2020.

[4] Z. A. A. Alyasseri, M. A. Al-Betar, I. A. Doush, M. A. Awadallah, A. K. Abasi, S. N. Makhadmeh, O. A. Alomari, K. H. Abdulkareem, A. Adam, R. Damasevicius, *et al.*, "Review on covid-19 diagnosis models based on machine learning and deep learning approaches," *Expert systems*, vol. 39, no. 3, p. e12759, 2022.

[5] L. Bao, H. Gao, W. Deng, Q. Lv, H. Yu, M. Liu, P. Yu, J. Liu, Y. Qu, S. Gong, *et al.*, "Transmission of severe acute respiratory syndrome coronavirus 2 via close contact and respiratory droplets among human angiotensin-converting enzyme 2 mice," *The Journal of infectious diseases*, vol. 222, no. 4, pp. 551–555, 2020.

[6] S. Schiaffino, S. Tritella, A. Cozzi, S. Carriero, L. Blandi, L. Ferraris, and F. Sardanelli, "Diagnostic performance of chest x-ray for covid-19 pneumonia during the sars-cov-2 pandemic in lombardy, italy," *Journal of thoracic imaging*, vol. 35, no. 4, pp. W105–W106, 2020.

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

[8] M. Cellina, M. Orsi, T. Toluian, C. V. Pittino, and G. Oliva, "False negative chest x-rays in patients affected by covid-19 pneumonia and corresponding chest ct findings," *Radiography*, vol. 26, no. 3, pp. e189–e194, 2020.

[9] A. Kumar, A. R. Tripathi, S. C. Satapathy, and Y.-D. Zhang, "Sars-net: Covid-19 detection from chest x-rays by combining graph convolutional network and convolutional neural network," *Pattern Recognition*, vol. 122, p. 108255, 2022.

[10] A. Abbas, M. M. Abdelsamea, and M. M. Gaber, "Classification of covid-19 in chest x-ray images using detrac deep convolutional neural network," *Applied Intelligence*, vol. 51, no. 2, pp. 854–864, 2021.

[11] N. K. Chowdhury, M. Rahman, M. A. Kabir, *et al.*, "Pdcovidnet: a parallel-dilated convolutional neural network architecture for detecting covid-19 from chest x-ray images," *Health information science and systems*, vol. 8, no. 1, pp. 1–14, 2020.

[12] A. A. Reshi, F. Rustam, A. Mehmood, A. Alhossan, Z. Alrabiah, A. Ahmad, H. Alsuwailem, and G. S. Choi, "An efficient cnn model for covid-19 disease detection based on x-ray image classification," *Complexity*, 2021.

[13] S. Pathan, P. Siddalingaswamy, and T. Ali, "Automated detection of covid-19 from chest x-ray scans using an optimized cnn architecture," *Applied Soft Computing*, vol. 104, p. 107238, 2021.

[14] S. Ahuja, B. K. Panigrahi, N. Dey, V. Rajinikanth, and T. K. Gandhi, "Deep transfer learning-based automated detection of covid-19 from lung ct scan slices," *Applied Intelligence*, vol. 51, no. 1, pp. 571–585, 2021.

[15] N. El-Rashidy, S. El-Sappagh, S. Islam, H. M. El-Bakry, and S. Abdelrazek, "End-to-end deep learning framework for coronavirus (covid-19) detection and monitoring," *Electronics*, vol. 9, no. 9, p. 1439, 2020.

[16] D. Wang, J. Mo, G. Zhou, L. Xu, and Y. Liu, "An efficient mixture of deep and machine learning models for covid-19 diagnosis in chest x-ray images," *PloS one*, vol. 15, no. 11, p. e0242535, 2020.

[17] I. U. Khan and N. Aslam, "A deep-learning-based framework for automated diagnosis of covid-19 using x-ray images," *Information*, vol. 11, no. 9, p. 419, 2020.

[18] D. Keidar, D. Yaron, E. Goldstein, Y. Shachar, A. Blass, L. Charbinsky, I. Aharony, L. Lifshitz, D. Lumelsky, Z. Neeman, *et al.*, "Covid-19 classification of x-ray images using deep neural networks," *European radiology*, vol. 31, no. 12, pp. 9654–9663, 2021.

[19] J. P. Cohen, P. Morrison, L. Dao, K. Roth, T. Q. Duong, and M. Ghassemi, "Covid-19 image data collection: Prospective predictions are the future," *arXiv preprint arXiv:2006.11988*, 2020.

[20] A. Khan, A. Sohail, U. Zahoora, and A. S. Qureshi, "A survey of the recent architectures of deep convolutional neural networks," *Artificial intelligence review*, vol. 53, no. 8, pp. 5455–5516, 2020.

[21] A. Signoroni, M. Savardi, S. Benini, N. Adami, R. Leonardi, P. Gibellini, F. Vaccher, M. Ravanelli, A. Borghesi, R. Maroldi, *et al.*, "Bs-net: Learning covid-19 pneumonia severity on a large chest x-ray dataset," *Medical Image Analysis*, vol. 71, p. 102046, 2021.

[22] J. Zhu, B. Shen, A. Abbasi, M. Hoshmand-Kochi, H. Li, and T. Q. Duong, "Deep transfer learning artificial intelligence accurately stages covid-19 lung disease severity on portable chest radiographs," *PloS one*, vol. 15, no. 7, p. e0236621, 2020.

[23] S. H. Khan, A. Sohail, M. M. Zafar, and A. Khan, "Coronavirus disease analysis using chest x-ray images and a novel deep convolutional neural network," *Photodiagnosis and Photodynamic Therapy*, vol. 35, p. 102473, 2021.

[24] J. Irvin, P. Rajpurkar, M. Ko, Y. Yu, S. Ciurea-Ilcus, C. Chute, H. Marklund, B. Haghgoo, R. Ball, K. Shpanskaya, *et al.*, "Chexpert: A large chest radiograph dataset with uncertainty labels and expert comparison," in *AAAI Conference on Artificial Intelligence*, vol. 33, pp. 590–597, 2019.

[25] L. Perez and J. Wang, "The effectiveness of data augmentation in image classification using deep learning," *arXiv preprint arXiv:1712.04621*, 2017.

[26] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

[27] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer, “Squeezenet: Alexnet-level accuracy with 50x fewer parameters and < 0.5 mb model size,” *arXiv preprint arXiv:1602.07360*, 2016.

[28] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, “Densely connected convolutional networks,” in *the IEEE conference on computer vision and pattern recognition*, pp. 4700–4708, 2017.

[29] T. Ozturk, M. Talo, E. A. Yildirim, U. B. Baloglu, O. Yildirim, and U. R. Acharya, “Automated detection of covid-19 cases using deep neural networks with x-ray images,” *Computers in biology and medicine*, vol. 121, p. 103792, 2020.