

An Optimized CNN Model Architecture for Detecting Coronavirus (COVID-19) with X-Ray Images

Anas Basalamah¹ and Shadikur Rahman^{2,*}

¹Umm Al-Qura University, Makkah, Saudi Arabia

²Daffodil International University, Dhaka, Bangladesh

*Corresponding Author: Shadikur Rahman. Email: shadikur35-988@diu.edu.bd

Received: 16 January 2021; Accepted: 28 April 2021

Abstract: This paper demonstrates empirical research on using convolutional neural networks (CNN) of deep learning techniques to classify X-rays of COVID-19 patients versus normal patients by feature extraction. Feature extraction is one of the most significant phases for classifying medical X-rays radiography that requires inclusive domain knowledge. In this study, CNN architectures such as VGG-16, VGG-19, ResNet50, ResNet18 are compared, and an optimized model for feature extraction in X-ray images from various domains involving several classes is proposed. An X-ray radiography classifier with TensorFlow GPU is created executing CNN architectures and our proposed optimized model for classifying COVID-19 (Negative or Positive). Then, 2,134 X-rays of normal patients and COVID-19 patients generated by an existing open-source online dataset were labeled to train the optimized models. Among those, the optimized model architecture classifier technique achieves higher accuracy (0.97) than four other models, specifically VGG-16, VGG-19, ResNet18, and ResNet50 (0.96, 0.72, 0.91, and 0.93, respectively). Therefore, this study will enable radiologists to more efficiently and effectively classify a patient's coronavirus disease.

Keywords: X-ray image classification; X-ray feature extraction; COVID-19; coronavirus disease; convolutional neural networks; optimized model

1 Introduction

Since the beginning of 2020, the novel Coronavirus disease (COVID-19) influenced by the critical acute respiratory syndrome coronavirus 2 (SARS-CoV-2) has grown into a serious health problem worldwide [1,2]. So far, a total of 20,820,389 confirmed cases and 747,476 death are documented globally [2], which led to the World Health Organization (WHO) announcing a worldwide epidemic on March 11, 2020. While 95% of infected patients survive the disease, 5% of patients deceased, and 1% of patients experience severe or critical medical conditions [3]. The main symptoms of COVID-19 are cold, cough, tiredness, and complexity in breathing, with critical conditions leading to organ failure and death. Yet, a comparative analysis indicates that patients can be infected without showing symptoms, while the virus can cause death in patients with collapsed immune systems [4]. COVID-19 spreads from one person to



This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

another through corporal interaction and proximity. In general, healthy individuals can be affected simply through breath contact or indirectly through hand contact with a COVID-19 carrier [5]. The current clinical approach to test this virus is based on reverse transcription-polymerase chain reactions (RT-PCR). As these tests take 4-6 hours to recognize COVID-19 sufferers, this comprises a great threat for spreading to a large populace considering the above-average contagious nature of the virus [6]. To reduce this risk of spread, this paper aims to quickly identify COVID-19 with X-ray images and computer vision technology.

Recently, a variety of studies have proven the capability to observe the presence of COVID-19 without a RT-PCR test kit. Some studies have shown that using a Computed Tomography (CT) and X-ray with CNN of Deep Learning (DL) method can be a better option to detect this disease more accurately. The work of image categorization has been driven in a single particular domain of physiology, such as chest CT scans of COVID-19 patients [7], chest X-ray images of different types of patients (i.e., COVID-19, pneumonia) [8,9]. This study proposes an optimized model based on Convolutional Neural Network (CNN) architecture to automatically identify COVID-19 in medical chest X-ray radiography.

The paper contributes to a comprehensive assessment of the five milestone CNN architectures, i.e. VGG16, VGG19, ResNet18, ResNet50, and this paper has optimized model for classifying COVID-19 X-ray. Optimized model architecture provides both faster results and greater accuracy than other models on the used small datasets. Normal chest X-ray and COVID-19 chest X-ray were applied to detect the symptoms of the classified disease. In this application, CNN architecture is optimized and combines multiple convolutions and pooling layers to quickly train the model with better accuracy. For this purpose, the well-known Deep transfer learning (DTL) models of CNN architectures are used to optimize the CNN model architecture using a Keras Sequential API and TensorFlow for image classification. The contributions of this paper are as follows:

- A novel application of convolutional neural networks of deep learning (DL) on COVID-19 detection based on chest X-ray.
- Various deep learning (DL) methods on COVID-19 detection and an identified most-convenient deep approach.
- An optimized CNN model architecture compared to DTL models.

2 Related Work

Machine learning (ML), artificial intelligence (AI), and deep learning (DL) have all seen accelerated and broadened applications in biomedical research. Study [8] proposed a new CNN model (MobileNetV2, SqueezeNet) and Support Vector Machines (SVM) for the classification of COVID-19 chest X-ray. It produced a 99.27% accuracy rate of image classification, but showed an insufficient number of X-ray images as the major drawback of their work. Study [9] also used DTL with the CNN frameworks VGG19, MobileNet v2, Inception, Xception, and Inception ResNet v2 for classification of COVID-19 chest X-ray images with three other types (normal: 504 images; bacterial pneumonia: 700 images; confirmed COVID-19: 224 images) with a 96.78% accuracy rate for image classification. Study [10] also used a DL model named ResNet50 for the classification of COVID-19 X-ray images with three different forms (normal, bacterial, viral) from chest radiographs at an accuracy rate of 96.23% for the dataset. A deep learning (DL)-based technique applying chest X-ray is described by [11], which represented a conventional CNN based on ResNet50, InceptionV3, and Inception-ResNetV2 models to predict COVID-19 patients. This study used an image dataset on a two-class chest X-ray (COVID-19 positive or negative) at a reported 98% accuracy. In [12], an initial screening model was established to separate COVID-19 and Influenza-A viral pneumonia from healthy cases applying pulmonary CT scans and deep learning (DL) approaches. As CNN is an end-to-end network, it is fit for X-ray analysis, and in study

[13], a CNN is applied to execute a model for diagnosing defects with 16,569 chest X-ray images to execute a multi-class classify model for an accurate diagnosis. Study [14] performed a classification of pneumonia X-ray radiography using three deep learning (DL) methods with 1) a training model from scratch, 2) a model without fine-tuning, and 3) a fine-tuned model. Applying the ResNet model, they identify pneumonia X-ray images with multiple labels. They also applied the Multi-Layer Perceptron (MLP) as a classification approach and obtained an 82.2% average accuracy. In [15], pneumonia was detected in chest X-ray using DL and DTL methods in five new models. The authors reported a score of 96.4% accuracy with their established ensemble deep learning model. Study [16] compared three CNN architectures (i.e., LeNet, AlexNet, and GoogLeNet) to classify medical radiography of anatomy objects with their modified CNN architecture which provides better performance. Their modified CNN architecture is based on AlexNet containing multiple convolutions and pooling layers.

3 Research Methodology

First, a dataset of COVID-19 chest X-ray radiography is selected for this research. After that, the image classifier is built with TensorFlow [17] by implementing a CNN architecture, and a CNN model is proposed to identify COVID-19 as positive or negative. As preparation, several steps need to be taken to complete the process, including image pre-processing, training model, an analyzing model, an optimizing model, an evaluation model process, and the classification of an image with the help of DL methods. Then, the image classification is received based on the CNN model of this research. Three main points are presented in this work: I. Classification image, II. Detection disease, and III. COVID-19 extraction. Fig. 1 represents an overview of this research.

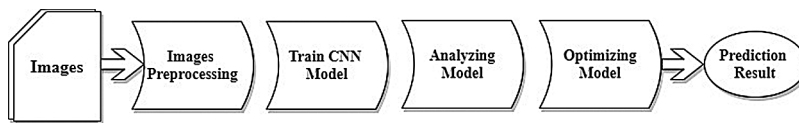


Figure 1: Research methodology

3.1 Training Corpus Generation

This research built a *corpus* by collecting X-ray radiography of normal patients and COVID-19 patients from online sources, which comprises of a total of 2,134 chest X-ray radiography from both COVID-19 patients and normal patients with two identified classes: I. COVID-19 X-ray image (infections), and II. Normal X-ray (no infections). Fig. 2 summarizes the numerical representation of the COVID-19 patient radiographs.

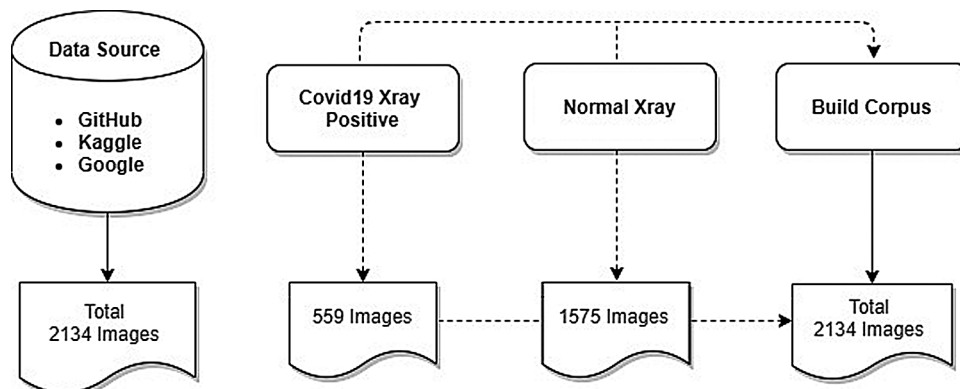


Figure 2: Numerical representation of the *corpus*

The current version of the dataset contains 559 COVID-19 positive radiographs of 342 COVID-19 patients, including 219 females and 128 males. There are a total of 1,575 negative cases of pneumonia used as normal X-ray images.

Corpus Labels Categorizing

After collecting the COVID-19 images, the X-rays were categorized for each patient. As mentioned, for this paper only two *corpus* labels categories were created. As such, the researchers used healthy patient X-rays alongside COVID-19 patient images. The dissimilarities between normal X-rays and COVID-19 X-rays were defined as clear and fuzzy chest X-rays, respectively. Fig. 3 provides an example of both X-ray image categories.

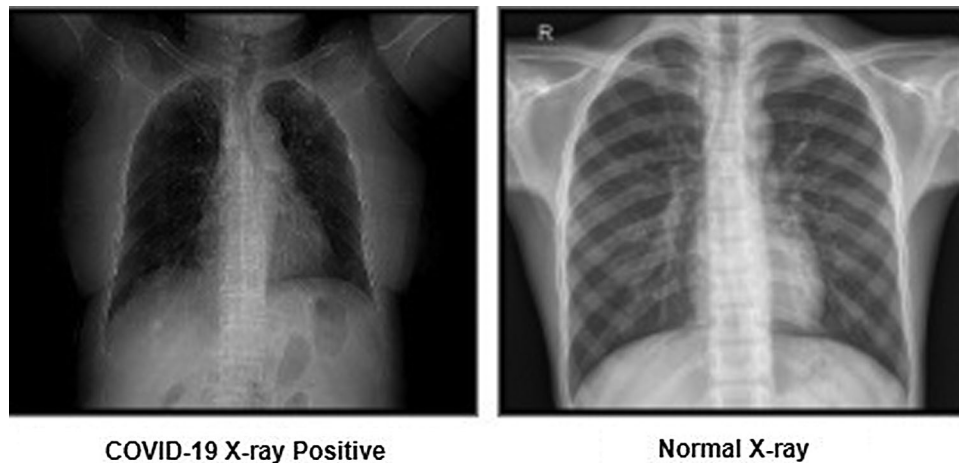


Figure 3: Representation of *corpus* labels

3.2 Image Preprocessing

To correctly analyze the COVID-19 *corpus* datasets, all X-ray images were preprocessed. In most cases, it was identified that image data was not accurately cleaned. As cleaning requires image preprocessing, we have taken the necessary image preprocessing steps for the training, validation, and testing of our model. The following steps have been used: read image, resize image, and remove noise. Fig. 4 shows an overview of our *corpus* image preprocessing process.

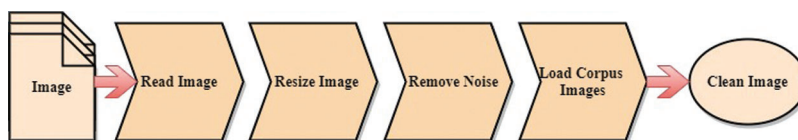


Figure 4: Image preprocessing process

3.2.1 Read Image

In this step, we load the path to our *corpus* dataset into a variable and also store the categories (normal X-ray vs. COVID-19 X-ray positive) of our *corpus* X-rays into a variable. Then we generated two-loop to store folders carrying x_ray images inside arrays. We have given the index number respectively (0, 1) for each level of the category variable. To read the image, python libraries have been used, such as OpenCV, Numpy, os, and matplotlib.

3.2.2 *Resize Image*

In this step, we have resized our *corpus* image to better train the model. In the early stage, the *corpus* image size is not the same quality, and each image is different in shape. Hence, we have established a standard size for all X-rays maintain into the training model algorithm.

3.2.3 *Remove Noise*

After resizing the image, we have smoothed each X-ray to erase unnecessary noise. We used a Gaussian blur to remove the image noise. Gaussian blur is the goal of blurring a X-ray with a Gaussian function.

3.2.4 *Load Training Data*

After the preceding steps, we have created a training data load function for the model training. From that function, we get the classification index numbers to classify the images of the model. After that, we have used the Pickle Library to save the *corpus* labels images for a training model. [Tab. 1](#) presents the result of the load training *corpus* data.

Table 1: Result of load training *corpus* image

<i>Corpus</i> Labels	Index Numbers	Load Labels Images
COVID-19 X-ray positive	[0]	559/559
Normal X-ray	[1]	1575/1575

3.3 *Research Training Model*

This research used the most popular CNN architecture of DTL (i.e., VGG-16, VGG-19), residual networks (i.e., RestNet50, RestNet18) and this paper proposed optimized DL model architecture. The CNN architecture represents a significant scientific improvement in image classification [18]. To train this study's model, X-ray *corpus* data was collected in the first step. In the second step, *corpus* data of the X-ray images was trained with the chosen training model. Third, the image classification process receives an input (i.e., the X-ray images) and predicts the image classification (COVID-19 positive or negative). [Fig. 5](#) represents an outline of the research training model.

3.3.1 *Convolutional Neural Networks*

CNNs are a particular kind of deep neural network that accomplishes magnificently in computer vision issues such as image classification or object detection. This research created an image classifier with TensorFlow by executing a CNN to classify COVID-19 X-rays. [Fig. 6](#) shows an overview of our CNNs model architecture.

3.4 *Analyzing Model Architecture*

In this section, to perform analyzing model architecture, we analyze the convolutional layers for better training our model architectures. We have introduced TensorBoard, which is an application that has been used to optimize the model's training statistics over time. To analyze the training model, we have analyzed various layers, such as the Conv2DLayer, DenseLayer, FlattenLayer, and ActivationLayer. We also analyzed the LayerSizes = [16,32,64,128,256] which is enough for the *corpus* images. And we have chosen the number of epochs = (5 to 30), the number of batch size = 32, validation split = 0.3 and callbacks. [Fig. 7](#) shows an overview of analyzing the CNN model architectures.

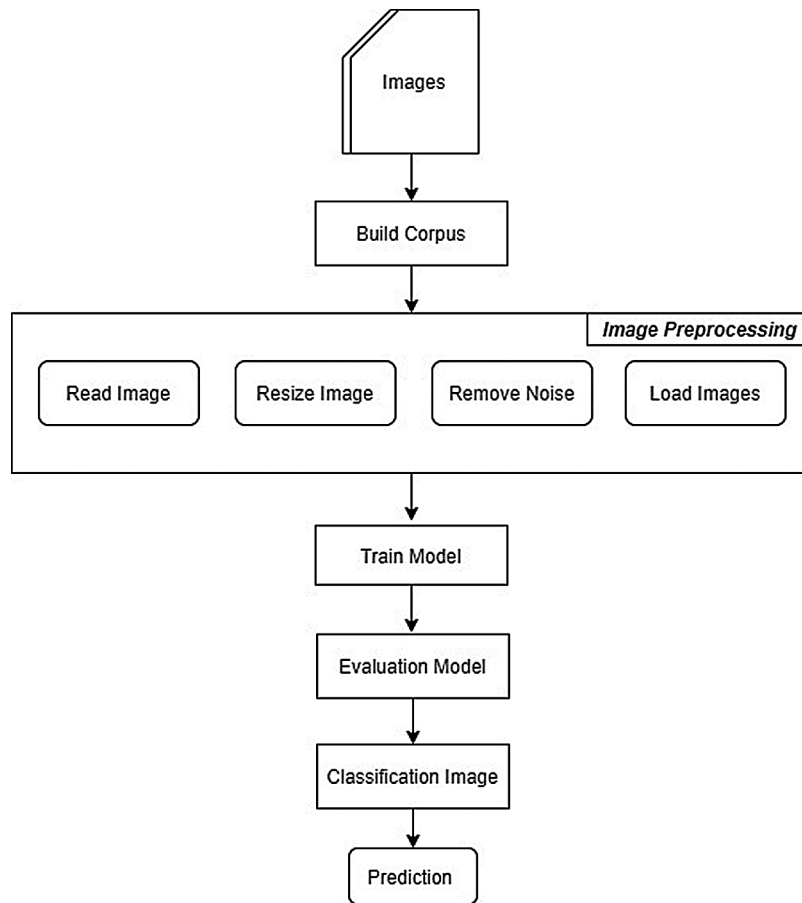


Figure 5: Representation of the training model

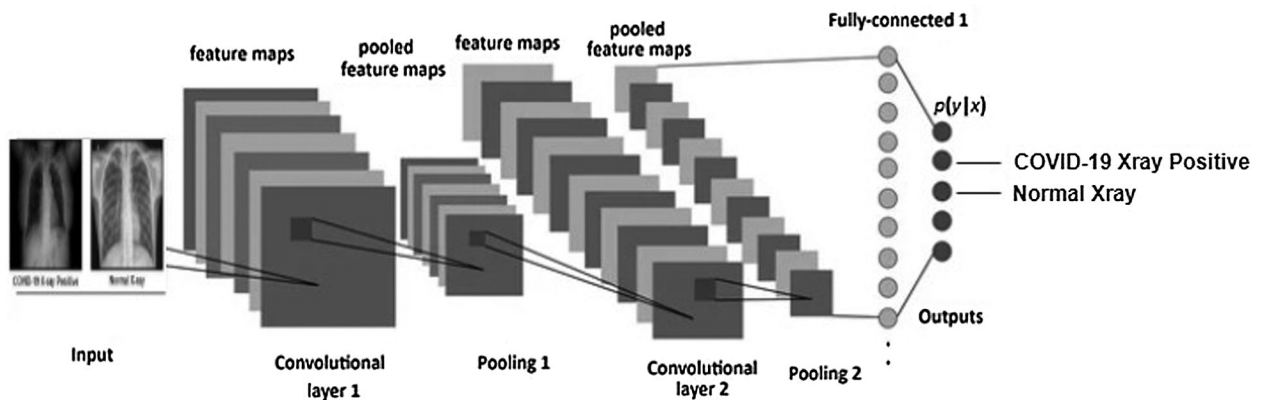


Figure 6: CNN model architecture

3.5 Optimizing Model Architecture

After analyzing the model, we have optimized the convolutional layers for better training our analyzing model architectures. We have introduced TensorFlow GPU, which is an application that we have used to optimize the models. For optimizing the analyzed model, we have optimized several layers such as DenseLayers = [0,1,2] and ConvLayers = [1,2,3]. We have also optimized the LayerSizes = [32,64,128]

which is enough for the *corpus* images. In addition, we have chosen the number of epochs = 5, the Number of batch size = 32, validation split = 0.3 and callbacks.

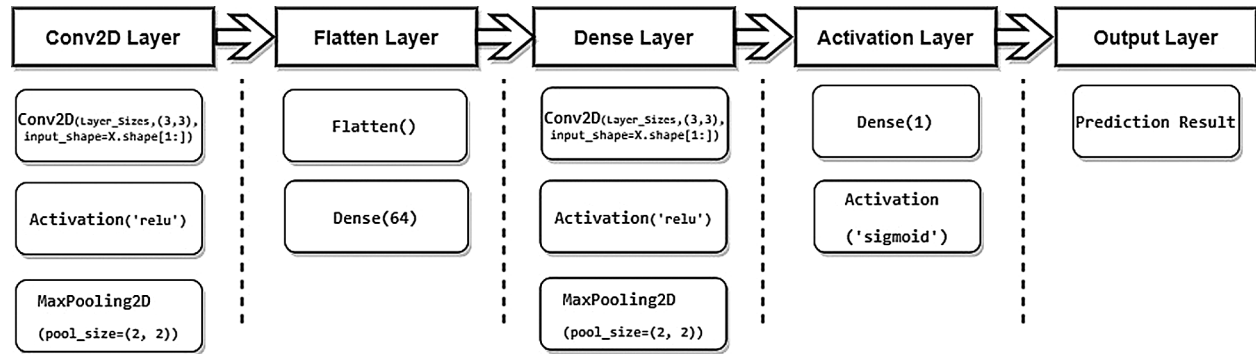


Figure 7: Analyzing CNN model architectures

For the final experimental propose model architecture, we consider our layer to detect a COVID-19 classification. CNN architecture details covering the interpretation of the layers, such as the Conv2DLayer, DenseLayer, FlattenLayer, and ActivationLayer. [Tab. 2](#) show an overview of the proposed model architecture layers.

Table 2: Summary of propose model architecture layers

Layers Types	Output-shape	Param (#)
Conv2D_L_1 (Conv2D)	(None, 48, 48, 128)	1280
Activation_L_1 (Activation)	(None, 48, 48, 128)	0
Max polling2D_L_1 (MaxPooling)	(None, 24, 24, 128)	0
Conv2D_L_2 (Conv2D)	(None, 22, 22, 128)	147584
Activation_L_2 (Activation)	(None, 22, 22, 128)	0
Max polling_2D_L_2 (MaxPooling)	(None, 11, 11, 128)	0
Conv2D_L_3 (Conv2D)	(None, 9, 9, 128)	147584
Activation_L_3 (Activation)	(None, 9, 9, 128)	0
Max polling2D_L_3 (MaxPooling)	(None, 4, 4, 128)	0
Flatten_L_1 (Flatten)	(None, 2048)	0
Dense_L_1 (Dense)	(None, 128)	262272
Activation_L_4 (Activation)	(None, 128)	0
Dense_L_2 (Dense)	(None, 128)	16512
Activation_L_5 (Activation)	(None, 128)	0
Dense_L_3 (Dense)	(None, 1)	129
Activation_L_6 (Activation)	(None, 1)	129

We optimized the model layers architecture ([Algorithm 1](#)), and consider the layer to measure the performance of the models. We modified all dense layers, layer sizes, and convolution layers in a feature extraction of the identified image classification with the optimized model architecture. [Algorithm 1](#) describes the optimized model architecture layers.

Algorithm 1: Optimized proposed model

```

Data: Corpus X-ray images
Result: Image classification from X-ray images
DenseLayers = [0,1,2];
LayerSizes = [32,64,128];
ConvLayers = [1,2,3];
for DenseLayer in Dense_Layers do
  for LayerSize in LayerSizes do
    for ConvLayer in ConvLayers do
      Model = Sequential();
      Conv2D(layer_size, (3, 3);
      Activation('relu');
      MaxPooling2D(pool_size=(2, 2);
      for l in range(ConvLayers-1) do
        Conv2D(LayerSize, (3, 3);
        Activation('relu');
        MaxPooling2D(pool_size=(2, 2);
      end
      Flatten();
      for l in range(DenseLayer) do
        Dense(LayerSize);
        Activation('relu')
      end
      Dense(1);
      Activation('sigmoid');
    end
  end
end

```

Optimizing led to multiple combinations of the model. [Tab. 3](#) provides an overview of the combination architecture of the optimizing architecture.

4 Result & Discussion

Here, we evaluate the image classification performance of our model by gathering COVID-19 X-ray images to create image classification results. After that, we built the X-ray image classifier with TensorFlow by executing a CNN architecture and optimized proposed models architecture to identify X-ray images into two types: COVID-19 positive or negative (i.e., normal). Several steps were taken to complete the preparation process, such as image pre-processing, training model, analyzing model, optimizing model, evaluation model process and classification image by deploying deep learning methods.

Table 3: Combinations of optimized model architecture

No.	ConvLayers	LayerSizes	DenseLayers	No.	ConvLayers	LayerSizes	DenseLayers
1	1_Conv_L	32_Nodes_S	0_Dense_L	15	3_Conv_L	64_Nodes_S	1_Dense_L
2	2_Conv_L	32_Nodes_S	0_Dense_L	16	1_Conv_L	128_Nodes_S	1_Dense_L
3	3_Conv_L	32_Nodes_S	0_Dense_L	17	2_Conv_L	128_Nodes_S	1_Dense_L
4	1_Conv_L	64_Nodes_S	0_Dense_L	18	3_Conv_L	128_Nodes_S	1_Dense_L
5	2_Conv_L	64_Nodes_S	0_Dense_L	19	1_Conv_L	32_Nodes_S	2_Dense_L
6	3_Conv_L	64_Nodes_S	0_Dense_L	20	2_Conv_L	32_Nodes_S	2_Dense_L
7	1_Conv_L	128_Nodes_S	0_Dense_L	21	3_Conv_L	32_Nodes_S	2_Dense_L
8	2_Conv_L	128_Nodes_S	0_Dense_L	22	1_Conv_L	64_Nodes_S	2_Dense_L
9	3_Conv_L	128_Nodes_S	0_Dense_L	23	2_Conv_L	64_Nodes_S	2_Dense_L
10	1_Conv_L	32_Nodes_S	1_Dense_L	24	3_Conv_L	64_Nodes_S	2_Dense_L
11	2_Conv_L	32_Nodes_S	1_Dense_L	25	1_Conv_L	128_Nodes_S	2_Dense_L
12	3_Conv_L	32_Nodes_S	1_Dense_L	26	2_Conv_L	128_Nodes_S	2_Dense_L
13	1_Conv_L	64_Nodes_S	1_Dense_L	27	3_Conv_L	128_Nodes_S	2_Dense_L
14	2_Conv_L	64_Nodes_S	1_Dense_L				

4.1 Image Classification Results

In order to identify the *corpus* images as COVID-19 positive or negative, CNN architecture image classification algorithms are used. Tab. 4 shows the accuracy of the CNN training model.

Table 4: Results of the initial trained CNN model

Models	Loss	Accuracy	Validation loss	Validation accuracy
VGG-16	0.11	0.96	0.07	0.95
VGG-19	0.61	0.72	0.56	0.72
RestNet18	0.23	0.91	0.43	0.93
RestNet50	0.20	0.93	0.25	0.93
Proposed Model	0.06	0.97	0.05	0.98

After the initial training model, the proposed model architecture results were analyzed for better training of the optimized model architecture. We have used the TensorBoard visualization graph for analyzing model results. Our analyzing model LayerSizes = [16,32,64,128,256] respectively train accuracy [0.94,0.96,0.95,0.96,0.94]. Fig. 8 shows the accuracy of analyzing the CNN training model.

After running the analyzing model, the results of training accuracy were chosen as the better LayerSizes = [32,64,128] respectively training accuracy [0.96,0.95,0.96]. Several layers such as DenseLayers = [0,1,2] and ConvLayers = [1,2,3] were further optimized in the model. We also optimized the LayerSizes = [32,64,128] which is enough for the *corpus* images. Fig. 9 shows the accuracy of optimizing the CNN training models.

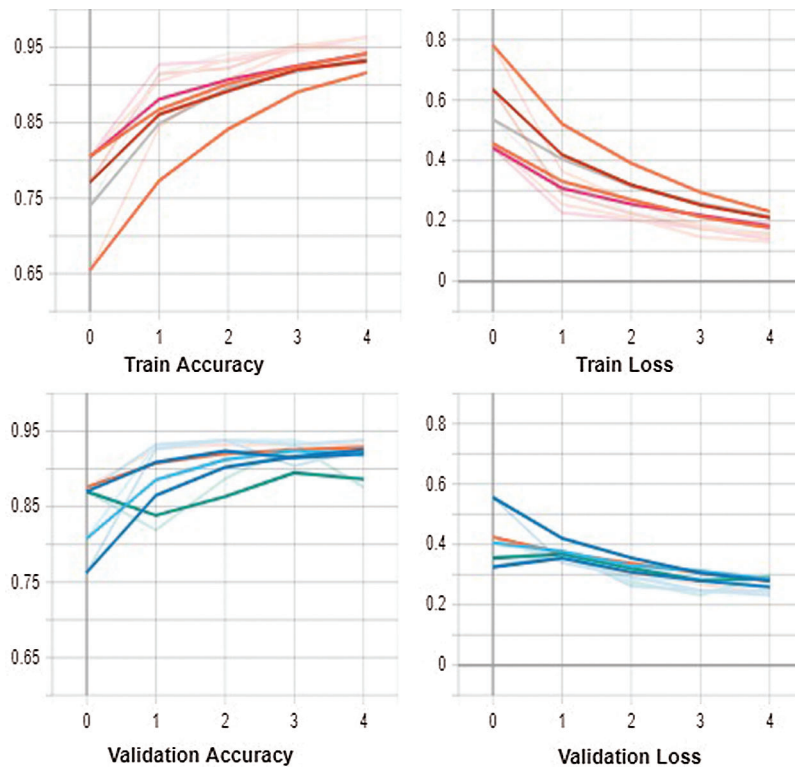


Figure 8: Result analysis of trained CNN models

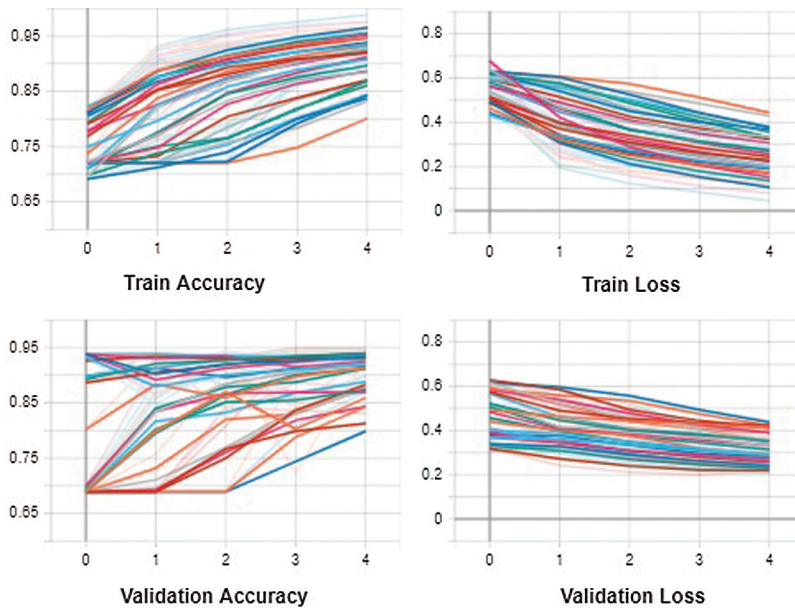


Figure 9: Optimization results of trained CNN models

Overall, optimizing the CNN model led to improved training accuracy and validation accuracy models. It also returned the lowest validation loss models possible. Models with one dense layer were discovered to provide better results overall. There are some successful models with DenseLayers. [Tab. 5](#) shows the top ten optimizing models.

Table 5: Representation of the top ten optimized model results

Model	Loss	Accuracy	Validation Loss	Validation Accuracy
1-Conv-128-Nodes-2-Dense	0.04	0.98	0.21	0.94
1-Conv-128-Nodes-1-Dense	0.07	0.97	0.20	0.94
1-Conv-64-Nodes-1-Dense	0.08	0.97	0.25	0.94
1-Conv-128-Nodes-0-Dense	0.12	0.96	0.21	0.94
1-Conv-64-Nodes 2-Dense	0.12	0.96	0.21	0.93
1-Conv-32-Nodes 0-Dense	0.06	0.97	0.06	0.98
2-Conv-32-Nodes 0-Dense	0.06	0.97	0.10	0.96
3-Conv-32-Nodes 0-Dense	0.07	0.97	0.06	0.97
1-Conv-64-Nodes 0-Dense	0.05	0.97	0.05	0.98
3-Conv-64-Nodes 0-Dense	0.06	0.97	0.05	0.98

4.2 Validation Corpus Images

The COVID-19 *corpus* image data that were used to train the CNN architectures and to optimize the model architecture, contains 2,134 X-ray radiography (COVID-19 positive X-ray: 559 vs. normal X-ray: 1575). We train our models considering 70% (1,494 images) as a training dataset of the COVID-19 images and used the remaining 30% of data (640 X-ray images) to test the model. The test datasets are used to make a neutral assessment of the final model to adopt to the training dataset.

4.3 Confusion Matrix in Corpus Dataset

The confusion matrix provides a visualization of the performance of the trained model. It has also summarized the results of testing the model for further inspection. This research created an optimizing CNN to classify the *corpus* image for visualization confusion matrix in TensorBoard. We constructed an optimizing neural network to identify images in the *corpus*. This dataset consists of 224×224 grayscale images of two labels (normal X-ray vs. COVID_19 positive X-ray)) of the *corpus*. Fig. 10 shows the visualization confusion matrix in TensorBoard.

4.4 COVID-19 Image Classification Estimation

The trained model provides COVID-19 X-ray classification results against the *corpus* images. The instance model classifies X-ray images based on the optimized training model which classifies the COVID-19 image through the *corpus* labels and labels index. Fig. 11 represents the classification results of X-ray images.

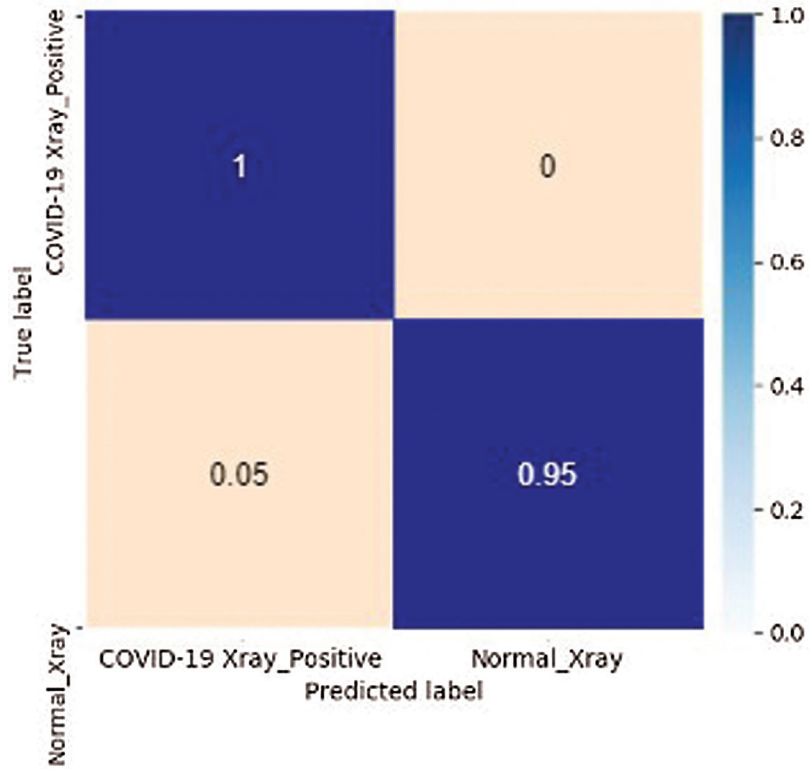


Figure 10: Confusion matrix in the *corpus*





Input Images	Corpus Labels	Labels Index	COVID-19 Results
	COVID-19 Xray_Positive	[0]	Positive
	Normal_Xray	[1]	Negative
	Normal_Xray	[1]	Negative
	COVID-19 Xray_Positive	[0]	Positive

Figure 11: Classification results of X-ray images

5 Conclusion

This study performed a unique technique of image classification to determine any medical radiography problems for the automated identification of COVID-19. The proposed method is an application to detect COVID-19 (positive or negative) by extracting features from the X-ray image dataset. For this application, the CNN's structure that integrates multiple convolutions and pooling layers for high-level feature extraction learning was optimized. The work presented here will help radiologists to effectively and efficiently classify a patient's COVID-19 status based on X-ray images. In the future, these findings can be used to create as an image classification system (i.e., an application tool for medical use) to identify COVID-19 in patients using enhanced X-ray imagery.

Funding Statement: The authors did not receive any specific funding for this study.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

- [1] K. Roosa, Y. Lee, R. Luo, A. Kirpich, R. Rothenberg *et al.*, "Real-time forecasts of the COVID-19 epidemic in China from February 5th to February 24th," *2020 Infectious Disease Modelling*, vol. 5, pp. 256–263, 2020.
- [2] World Health Organization, *World Health Organization coronavirus disease (COVID-19) dashboard*. Vol. 5. Geneva, Switzerland: World Health Organization, 2020. [Online]. Available at: <https://covid19.who.int>.
- [3] Worldometers, "Coronavirus Pandemic," 2019. [Online]. Available at: <https://www.worldometers.info/coronavirus>.
- [4] M. S. Razai, S. Mohammad, K. Doerholt, S. Ladhani and P. Oakeshott, "Coronavirus disease 2019 (COVID-19): A guide for UK GPs," *BMJ*, pp. 368, 2020.
- [5] X. Peng, X. Xu, Y. Li, L. Cheng, B. Ren *et al.*, "Transmission routes of 2019-nCoV and controls in dental practice," *International Journal of Oral Science*, vol. 12, no. 1, pp. 1–6, 2020.
- [6] J. Zhao, Y. Zhang, X. He and P. Xie, "COVID CT-Dataset: A CT scan dataset about COVID-19," arXiv preprint arXiv:2003.13865, 2020.
- [7] X. Xu, X. Jiang, C. Ma, P. Du, X. Li *et al.*, "A deep learning system to screen novel coronavirus disease 2019 pneumonia," *Engineering*, vol. 6, no. 10, pp. 1122–1129, 2020.
- [8] M. Toğaçar, B. Ergen and Z. Cömert, "Application of breast cancer diagnosis based on a combination of convolutional neural networks, ridge regression and linear discriminant analysis using invasive breast cancer images processed with autoencoders," *Medical Hypotheses*, vol. 135, pp. 109503, 2020.
- [9] I. D. Apostolopoulos and T. A. Mpesiana, "COVID-19: Automatic detection from x-ray images utilizing transfer learning with convolutional neural networks," *Physical and Engineering Sciences in Medicine*, vol. 43, pp. 635–640, 2020.
- [10] M. Farooq and A. Hafeez, "COVID-resnet: A deep learning framework for screening of COVID-19 from radiographs," arXiv preprint arXiv:2003.14395, 2020.
- [11] A. Narin, C. Kaya and Z. Pamuk, "Automatic detection of coronavirus disease (COVID-19) using x-ray images and deep convolutional neural networks," arXiv preprint arXiv:2003.10849, 2020.
- [12] S. Wang, B. Kang, J. Ma, X. Zeng, M. Xiao *et al.*, *A deep learning algorithm using CT images to screen for Corona Virus Disease (COVID-19)*. MedRxiv, 2020.
- [13] Y. Dong, Y. Pan, J. Zhang and W. Xu, "Learning to read chest X-ray images from 16000+ examples using CNN," in *2017 IEEE/ACM Int. Conf. on Connected Health: Applications, Systems and Engineering Technologies (CHASE)*, IEEE, pp. 51–57, 2017.
- [14] I. M. Baltruschat, H. Nickisch, M. Grass, T. Knopp and A. Saalbach, "Comparison of deep learning approaches for multi-label chest X-ray classification," *Scientific Reports*, vol. 9, no. 1, pp. 1–10, 2019.
- [15] V. Chouhan, S. K. Singh, A. Khamparia, D. Gupta, P. Tiwari *et al.*, "A novel transfer learning based approach for pneumonia detection in chest X-ray images," *Applied Sciences*, vol. 10, no. 2, pp. 559, 2020.

- [16] S. Khan and S. P. Yong, "A deep learning architecture for classifying medical images of anatomy object," in *2017 Asia-Pacific Signal and Information Processing Association Annual Summit and Conf. (APSIPA ASC)*, IEEE, pp. 1661–1668, 2017.
- [17] A. Géron, "Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems," O'Reilly Media, 2019.
- [18] S. Lawrence, C. L. Giles, A. C. Tsoi and A. D. Back, "Face recognition: A convolutional neural-network approach," *IEEE Transactions on Neural Networks*, vol. 8, no. 1, pp. 98–113, 1997.