

# ModCOVNN: a convolutional neural network approach in COVID-19 prognosis



Ahmed Wasif Reza <sup>a,1,\*</sup>, Jannatul Ferdous Sorna <sup>a,2</sup>, Md. Momtaz Uddin Rashel <sup>a,3</sup>,  
Mir Moynuddin Ahmed Shibly <sup>a,4</sup>

<sup>a</sup> Department of Computer Science and Engineering, East West University, Dhaka, Bangladesh  
<sup>1</sup> wasif@ewubd.edu; <sup>2</sup> sornaewu@gmail.com; <sup>3</sup> momtazrasel8@gmail.com; <sup>4</sup> shiblygnr@gmail.com  
\* corresponding author

## ARTICLE INFO

### Article history

Received December 25, 2020  
Revised March 22, 2021  
Accepted April 4, 2021  
Available online April 20, 2021

### Keywords

COVID-19  
Convolutional neural network  
Biomedical image classification  
Computer vision  
X-ray

## ABSTRACT

COVID-19 is a devastating pandemic in the history of humankind. It is a highly contagious flu that can spread from human to human. For being so contagious, detecting patients with it and isolating them has become the primary concern for healthcare professionals. However, identifying COVID-19 patients with a Polymerase chain reaction (PCR) test can sometimes be problematic and time-consuming. Therefore, detecting patients with this virus from X-ray chest images can be a perfect alternative to the de-facto standard PCR test. This article aims at providing such a decision support system that can detect COVID-19 patients with the help of X-ray images. To do that, a novel convolutional neural network (CNN) based architecture, namely ModCOVNN, has been introduced. To determine whether the proposed model works with good efficiency, two CNN-based architectures – VGG16 and VGG19 have been developed for the detection task. The experimental results of this study have proved that the proposed architecture has outperformed the other two models with 98.08% accuracy, 98.14% precision, and 98.4% recall. This result indicates that proper detection of COVID-19 patients with the help of X-ray images of the chest is possible using machine learning methods with high accuracy. This type of data-driven system can help us to overcome the current appalling situation throughout the world.



This is an open access article under the [CC-BY-SA](#) license.



## 1. Introduction

The consequence of the COVID-19 epidemic, previously observed by Chinese people only, has now become a matter of serious concern for the whole world. In late January – 2020, the epidemic was proclaimed public healthcare emergency of international significance, and ultimately on March 11, 2020, the World Health Organization has declared COVID-19 as a pandemic [1][2]. The mortality rate is growing alarmingly worldwide, requiring an emergency alert solution, and avoiding the accelerated dissemination of this pandemic. COVID-19 pandemic [3] could be severe because of its seriousness and high transmissibility. The healthcare system in many nations has reached the failure point because of the high and growing increasing rate of COVID-19 cases [1]. The impact on the healthcare sector is still high because of the large number of patients requiring long-term ICU treatment and automatic ventilators [4]. Already there are shortages of test kits and ventilators. Most countries had announced [1] full lockdown and have told their citizens to remain indoors and to be rigid to prevent gatherings.

Since the appropriate and rapid mass distribution of perfectly working vaccines is still difficult for various reasons, the global situation has been appalling to billions of people [5][6]. Among COVID-19

cases, symptoms occurring from nasal sore, septic shock, dry cough and fever, failure of multi-organ, extreme pneumonia, death [7][8], all are observed. Pneumonia is an illness that induces inflammation predominantly in the air cells of the lung, which is responsible for transferring oxygen [8]. The disorder of the respiratory system can be caused by COVID-19 [9]. The pulmonary system is the portion of the human organism that is the most infected by the virus, and therefore the use of chest X-rays could be a more effective approach to detect whether a patient is affected by COVID-19 or not than thermal scanning [10]. Another challenging aspect of the health care system is that the individuals infected with COVID-19 [11] could have transmitted this disease without any symptoms.

In this situation, an effective prognosis is essential for timely treatment to minimize huge potential pressure in the health sector. Therefore, it is also of particular significance that the prognosis of the suspicious patient is tested to promote the next phase for the affected individuals. Early detection of such individuals can also prevent more people from being affected. Moreover, there appears to be a significant lack of costly test kits due to infected people's exponential development [12]. Hence, X-rays can be alternative to detect patients with COVID-19. It can also play a key factor in detecting the COVID-19 virus with its low cost and rapid speed with its wide variety of applications [13]. Based on these newly emerged circumstances, this study has aimed at designing an efficient machine learning system that can screen out COVID-19 patients. Because of the increasingly growing COVID-19 positive cases worldwide, most countries face a lack of kits and equipment. The minimal amount of test kits and a growing number of everyday cases motivated us to create something which can easily detect whether a patient is COVID-19 positive or not. In this article, we introduce a model using machine learning that can help detect COVID-19 without using any test kit. To meet our desired goal, we have also compared our results with other state-of-the-art works. Our designed model has demonstrated very promising results.

The latest literature has inspired the applied methods of this study on recognizing COVID-19 from x-ray images. There are a handful of works that have been done in this field of biomedical image classification. In this study [1] to detect the cases of COVID-19, researchers proposed an Xception-based model naming CoroNet. For that, they used two datasets which worked extremely well on both. The model has achieved a detection rate of COVID-19 respectively 99%, 94.59%, and 89.5% for binary class, 3-classes, and 4-classes for a chest X-ray dataset. To detect COVID-19 from X-ray images, Wang and Wong [14] proposed a model named COVID-Net, which belongs to a deep residual architecture. This model is one of the earliest experiments on COVID-19 that utilized deep neural networks to categorize four classes of X-ray images. Considering the chest X-ray images, a VGG19 model with transfer learning was used in another work [15] where the researchers were able to obtain an accuracy of 98.75% and 93.48% for three and two-class classification tasks, respectively.

One such experiment was done in another study [16] where three different CNN models InceptionV3, InceptionResNetV2, and ResNet50 were used, respectively. Among them, 98%, which is the best accuracy has been achieved by ResNet50 for two-class classification. As any cases of pneumonia had not been included in their experiments, it is unclear how well their system could differentiate cases among other pneumonia-affected instances and COVID-19. To detect COVID-19 cases from X-ray images, a DarkNet architecture based on CNN was proposed in another study [17]. They had achieved an accuracy of 87.02% for the three-class classification challenge and 98.08% for the binary classification challenge. Xu *et al.* [18] and Stephen *et al.* [19] have used various deep learning methods for the automatic COVID-19 and pneumonia diagnosis task, while Sethy and Behera [20] implemented SVM and ResNet50, an architecture of CNN, to build their model for the recognition purpose of COVID-19. Moreover, their proposed technique reached a 95.38% accuracy considering a two-class problem. COVIDX-Net was proposed [21] for recognizing COVID-19 by examining X-rays. In their study, the number of images for COVID-19 positive was 25, where the number of healthy images was also 25. Using these images, they attain an accuracy of 90%. A detection rate of 96% is recorded by Li *et al.* [22] for a model COVNet designed, which could detect COVID-19 to extract visual properties from chest CT-images.

Butt *et al.* [23] introduced two independent models of classification to differentiate between COVID-19 images, with a gross accuracy of 86%. A ResNet23 based network was used for their first model, while the 2nd one was designed based on the structure of the first network where the fully connected layer was concatenating with the mechanism of location-attention for improving the overall accuracy rate. Brunese *et al.* [24] used 6,523 chest X-rays images for experimental analysis from different institutions. Detecting COVID-19 by their recommended model, it reached an accuracy equal to 97%. Their proposed model consisted of three phases: firstly, to detect the presence or absence of pneumonia in a chest X-ray, secondly to differentiate between COVID-19 and pneumonia, and the final move was directed at localizing the areas of COVID-19 presence symptoms involvement in the X-ray. This scientific base has led us to use a convolutional neural network approach for COVID-19 prognosis from chest x-ray images.

In the following sections of this article, the methods of this study have been presented. After that, the obtained results with appropriate discussion have been described. Finally, future works with concluding remarks have been discussed.

## 2. Method

This study classified chest X-ray images into three classes – COVID-19, normal, and pneumonia. For the classification task, a novel convolutional neural network (CNN) – based architecture namely ModCOVNN, has been introduced. Additionally, two convolutional neural network-based architectures – VGG16, and VGG19 have been used for biomedical image classification challenges for comparative analysis purposes. In this section, the used methodologies, including our proposed methodology, have been described.

### 2.1. CNN layers

Any artificial neural network that contains at least one convolutional layer is known as a convolutional neural network. Unlike other feature-wise tabular datasets, images are subject to have contextual information. The features corresponding to the images are not structured. Just using the pixel values of images as the input of a machine learning model does not produce expected performances. Therefore, the features from the images are needed to be extracted properly. Handcrafted features can also be generated from the images, but that is a very tedious job. Convolutional neural networks come to aid in this matter. These neural networks extract features from the images and classify those features into the desired classes as well.

The convolutional layer is the centerpiece of a CNN. It is the main part of the feature extraction process within the network. The other layers of a CNN revolve around the convolutional layers. The convolution process starts with several randomly initialized filters with a fixed window. Each filter slides over a matrix representation of an image and calculates scalar multiplication with the portion of the image the window overlaps. The values of each scalar multiplication are stored in another matrix which is a transformed matrix representation of the original input of a convolutional layer. In this way, the matrix representation of an image gets convolved into more understandable features. Each filter produces a transformed matrix. If a single matrix passes through a convolutional layer with  $n$  filters, there will be  $n$  transformed matrices. Let  $A$  be a sliced  $m \times n$  matrix from the input image. It is the  $x$ th row-wise slice and the  $y$ th column-wise slice from the original pixelated image. Assume that a filter is an  $m \times n$  matrix denoted by  $F$  and the transformed matrix is denoted by  $T$ . Note that, usually, a filter is a square matrix i.e.,  $m = n$ , and both  $A$  and  $F$  matrix must be of the same dimension. Now, each entry of the transformed matrix  $T$  will be filled with the dot product between  $A$  and  $F$  using the following formula,

$$T[x][y] = \sum_{i=0}^m \sum_{j=0}^n A[i][j] * F[i][j] \quad (1)$$

Now, if the convolving continues and the dimension of the matrix is not reduced, the feature space becomes very large and unmanageable. Therefore, the dimension of a matrix should be reduced regularly in a meaningful way. Pooling layers help to reduce the dimension of a matrix in a sophisticated way.

After every single or couple or three or more convolutional layers, a pooling layer is added. Pool size is defined, and that slider slides over the convolved matrix. While sliding, each pool's maximum or average is calculated and stored in a new matrix. In this way, the dimension of a matrix is reduced. Suppose  $A$  with dimension  $p \times p$  is the  $x$ th row-wise pool and the  $y$ th column-wise pool from the input matrix. Let  $R$  be the reduced matrix after pooling; then the reduced matrix will be calculated with either equation (2) for max-pooling or equation (3) for average pooling.

$$R[x][y] = \max(A) \quad (2)$$

$$R[x][y] = \frac{\sum_{i=0}^p \sum_{j=0}^p A[i][j]}{p * p} \quad (3)$$

After combining several convolutional layers and the max-pooling layer, the transformed matrices are squashed into a 1D vector utilizing a flatten layer when the feature extraction process seems to come to a meaningful end. After that, few dense layers are added. The last dense layer, the network's output layer, contains the number of neurons equal to the number of classes in a classification task. In this study, the number of neurons at the output layer is three – as we are interested in classifying X-rays into COVID-19, normal, and pneumonia. We have also used dropout layers in our ModCOVNN. Dropout layers have been used in between two fully connected dense layers. A dropout layer allows only a portion of the information to be propagated forward from the previous layer. This random selection of information to be forwarded helps the trained CNN from overfitting.

## 2.2. VGG16

VGG16 is a model of the deep neural network and transfer learning. In our study, we used the VGG16 model for the recognition of COVID-19. In 2015, the VGG16 network was introduced and developed by Simonyan and Zisserman [25]. Its architecture has been presented in Fig. 1. From Fig. 1, we can see that the architecture starts with back-to-back convolutional layers. After that, a max-pooling layer is added. Then there are again two back-to-back convolutional layers and a max-pooling layer. After that, three convolutional layers are followed by a pooling layer. Again, there are three convolutional layers followed by a max-pooling layer that repeats two times. The architecture ends with three dense layers. The last of these dense layers is the output layer with three neurons, indicating the three classes of our classification task. In total, there are thirteen convolutional layers and three dense layers. If we sum them up, the count is sixteen. That is the reason why this network is named VGG16.

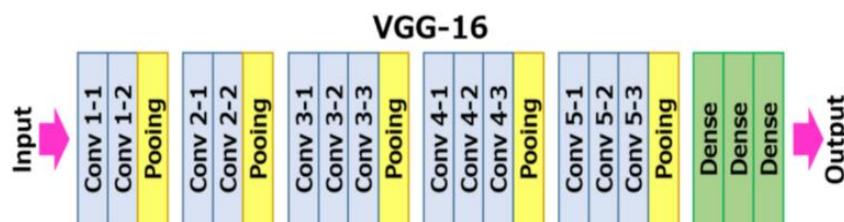


Fig. 1. VGG16 network.

## 2.3. VGG19

Another convolutional neural network that has been used in this study is VGG19. Like VGG16, it is also introduced by Simonyan and Zisserman [25] in 2015. It is a slightly deeper network than the VGG16. Instead of sixteen layers in VGG16, VGG19 has nineteen layers. Sixteen of them are convolutional layers with three additional dense layers. The difference between the two networks lies in the repetition of three consecutive convolutional layers followed by a pooling layer. For VGG16, those three blocks of three convolutional layers with a pooling layer have been replaced by four convolutional layers with a pooling layer. The rest of the network is identical to the VGG16 network.

## 2.4. ModCOVNN

This study also introduced a CNN-based architecture – ModCOVNN to classify the X-rays into COVID-19, normal, and pneumonia. It is a less deep architecture than VGG16 and VGG19. As opposed to sixteen and nineteen layers in VGG16 and VGG19, this network has nine layers. The block diagram of this architecture has been presented in Fig. 2. In this model, at first, there are two convolution layers with 32 filters. Each filter has a dimension of  $3 \times 3$ . Then comes the max-pooling layer with a stride of  $2 \times 2$ . Again, comes one convolution layer with 64 filters. The filters of this layer have a dimension of  $3 \times 3$ . Then, a max-pooling layer with stride  $2 \times 2$  is added. After that, comes one convolution layer with the number of filters 128. Similar to the previous convolutional layers, the filters of this layer also have a dimension of  $3 \times 3$ . After that, a max-pooling layer with stride  $2 \times 2$  is added. This combination of the max-pooling layer after a convolutional layer has been repeated four times. The process of feature extraction ends with a flatten layer. After that, there comes dense  $\rightarrow$  dropout  $\rightarrow$  dense  $\rightarrow$  dropout  $\rightarrow$  dense layers. The last dense layer has acted as the output layer that has three neurons.

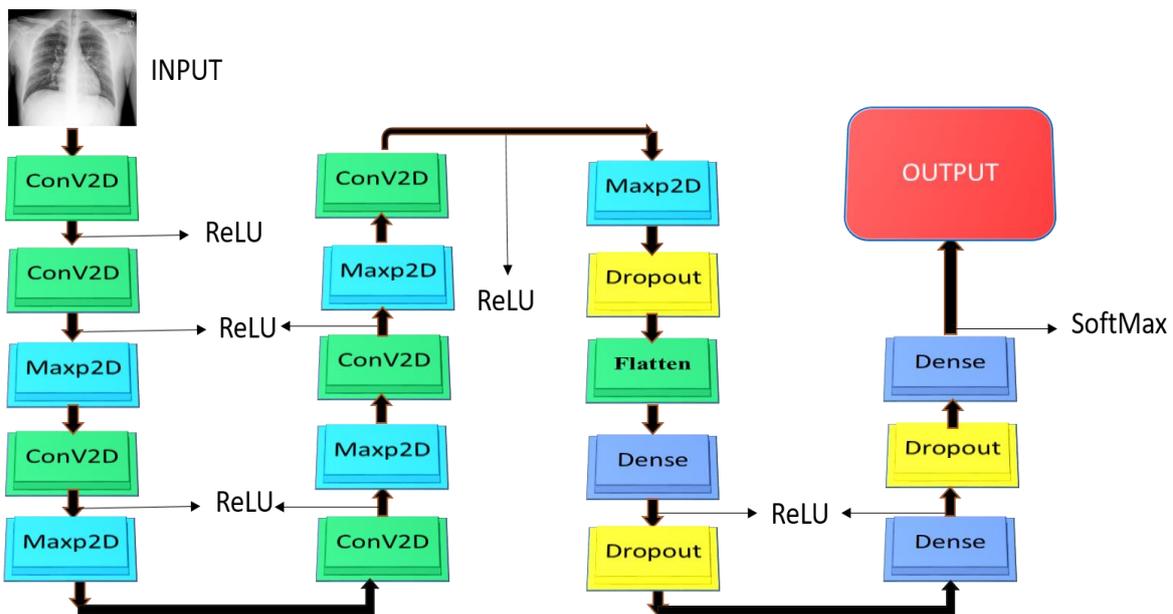


Fig. 2. Proposed ModCOVNN network.

## 2.5. Activation functions

In every convolutional and MaxPooling layer was used the rectifier linear unit (ReLU) activation function. A ReLU activation inspects whether the result is negative or positive. It transforms the negative values to zero and stores them in the transformed matrix to reduce instability in a Convnet. It also stores the positive values as-is in the output matrix. In the output layer of the three architectures of this study was used the SoftMax activation. A SoftMax function activates the neuron that has the highest probability in the output layer and classifies the input image to the class corresponding to that neuron. It transforms the inputs of each neuron to some probability values. The neuron with the highest probability gets selected as the predicted class.

## 2.6. Experimental Setup

In this study, three CNN-based machine learning models have been trained. The three models are VGG16, VGG19, and our proposed ModCOVNN. We have not used the pre-trained weights of VGG16 and VGG19. Instead, all three models have been trained from scratch. Due to the lack of computational horsepower, we have not used any graphical performance unit (GPU) to train the deep learning models. We have used the Keras library on top of TensorFlow for creating deep learning models. All machine learning models have been trained with a computer with a Core (TM) – i78550U CPU, 8GB ram, and a Windows 10 operating system.

## 2.7. Dataset

We have collected and merged these three types of images from various sources and then created our dataset. All these are open source. We tested whether there were any duplicate data in the dataset or not. After that, all the duplicate images have been removed. In this way, we have curated a dataset that is made up of 8670 chest x-rays. There are 1089 COVID-19 chest x-rays, 4905 normal x-rays, and 2676 pneumonia x-rays in the dataset. Initially, the dataset has been split into the training and test set. The training set contains 6740 images, while the test set has 1930 images distributed in three classes. During training, 10% of the training images have been set aside for validation. The class-wise distribution of the training, validation and test images is presented in Table 1 as it is seen that the dataset that has been used for the experiments is quite imbalanced. The training algorithm of ModCOVNN has been described in Fig. 3.

The details of the dataset for COVID-19 detection.

Class	No. of training images	No. of validation images	No. of test images	Total
COVID-19	642	71	376	1089
Pneumonia	1728	192	756	2676
Normal	3697	410	798	4905
Total	6067	673	1930	8670

### Algorithm: ModCOVNN training

```

1. begin
2.     resize the images to 224x224
3.     apply image augmentation techniques
4.     pass the training images through two convolutional layers
5.     pass the outputs of last convolutional layer through a max-pooling layers
6.     for i = 1 to 3
7.         begin
8.             pass the outputs of last layer to a convolutional layer
9.             pass the outputs of last layer to a max-pooling layer
10.        endfor
11.    add a dropout layer
12.    vectorize the neurons
13.    for i = 1 to 2
14.        begin
15.            pass the outputs of last layer to a dense layer
16.            pass the outputs of last layer to a dropout layer
17.        endfor
18.    pass the outputs of last layer through a dense layer with SoftMax activation
19. end

```

Fig. 3. The training of the model using the ModCOVNN network.

## 2.8. Image Data Transformation

Image data augmentation helps a deep learning model to achieve improved performances. It also helps a classifier to generalize well. In this study, different image augmentation strategies have been applied. Three transformation techniques have been used for image augmentation. Those are – rotating, zooming, and flipping. In Fig. 4, an overview of image data augmentation has been shown.

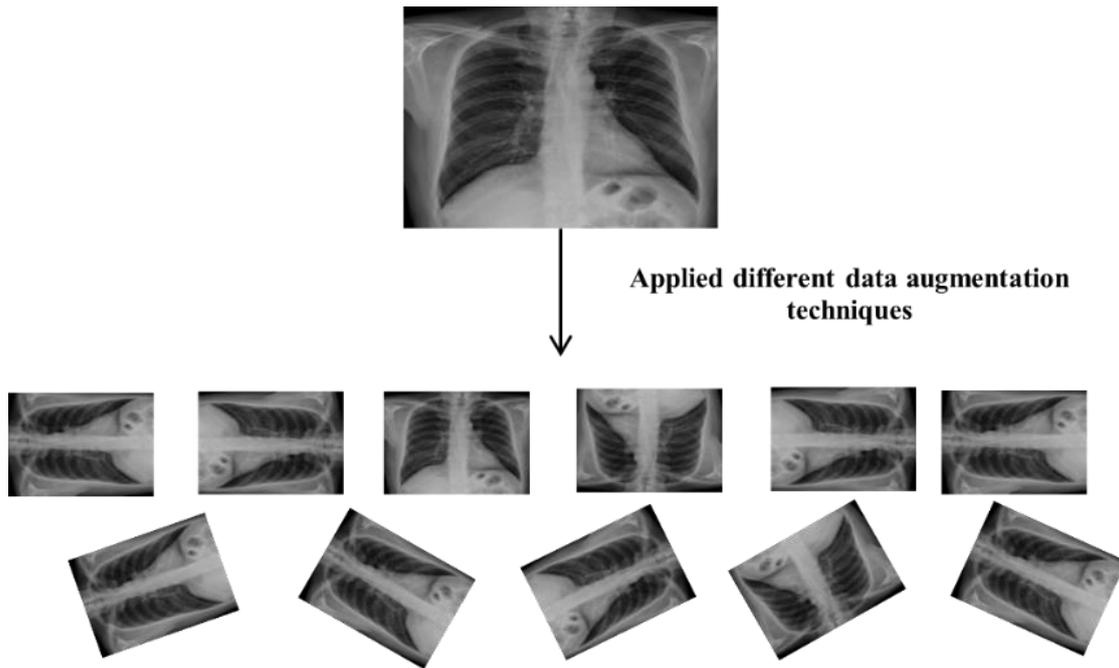


Fig. 4. Image data augmentation of chest X-ray dataset

## 2.9. Optimizers, Loss Function, Input Size, Batch Size, and Evaluation Metrics

The models have been trained from scratch in the experimentations with the optimizer named Adam (Adaptive Moment Estimation). Adam is a stochastic optimization approach that measures parameter-specific adaptive learning rates. Since the classification task of this study is a multi-class classification task, categorical cross-entropy has been used as the loss function. The images that we collected are from three different sources. Therefore they have different dimensions. For this reason, before feeding to the networks, the images have been resized into a fixed dimension of  $224 \times 224$ . The x-rays have been three-channelled (red, green, blue), and each pixel's value ranges from 0 to 255; as the neural networks work better with the numbers ranging from 0 to 1, the pixel values have been rescaled by dividing them by 255. Another important hyperparameter is batch size. The performance of a trained model depends on batch size. Usually, the models with bigger batch sizes demonstrate better performances than those with smaller batch sizes. Nevertheless, the increasing batch size is not always possible for the limitation of computational resources. With the setup of this study, a batch size of 64 has been used for the three models during training. Each model has been trained for 50 epochs. After the completion of training, the trained models have been tested with the holdout test images.

The performances of the models have been measured based on Precision, Recall, F1-score, and Accuracy. Precision is what is positive in the fraction of data points classified as positive. A precision of 1.0 means that any non-negative (positive) data point classified is positive. It is also known as positive predictive value. Precision can be calculated by dividing the number of true positives by the model's total number of data points classified positively.

$$Precision = \frac{TP}{TP+FP} \quad (4)$$

A measure of completeness is called Recall. It can be calculated by dividing the number of true positive data points by the total positive data points.

$$Recall = \frac{TP}{P} \quad (5)$$

The harmonic average of the precision and the Recall is the F1 score. It is defined as follows:

$$F1\_score = \frac{2 \times precision \times recall}{precision + recall} \quad (6)$$

And accuracy of a machine learning classification model is a way to calculate how much an algorithm correctly classifies a data point out of all the data points. It is the portion of the correctly predicted data points to the total input samples. It is defined as follows:

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \quad (7)$$

Here  $TP$ ,  $TN$ ,  $FP$ ,  $FN$ ,  $P$ , and  $N$  are true positive, true negative, false positive, false negative, the total number of positive instances, and the total number of negative instances, respectively. While calculating these evaluation metrics, the COVID-19 class has been considered as the positive class.

### 3. Results and Discussion

This study has demonstrated outstanding performances in detecting COVID-19 from X-rays. Our proposed ModCOVNN method has shown better performances than the other two implemented convolutional neural networks – VGG16 and VGG19. During training, the models' training accuracies have been observed. The ModCOVNN model has obtained 98.28% training accuracy. On the other hand, VGG16 and VGG19 have gained training accuracies of 97.14% and 97%, respectively.

The models have been trained for fifty epochs. During training, after each epoch, a model has been validated against the validation images. Based on the training and validation of each epoch, the training accuracy versus validation accuracy graph has been plotted. Similarly, the training loss vs validation loss graph has been plotted. This has been done to observe the stability of the models during training. It is important to watch whether the training of a model has been proper or not. In Fig. 5, two graphs corresponding to our proposed model have been illustrated. Fig. 5(a) shows the comparison of training accuracy with the validation accuracy over the fifty epochs that the model has been trained. It indicates that the model has demonstrated good performance on the validation images with respect to training images. The upward slope of both training accuracy and validation accuracy indicates the proper training of the model. Similarly, Fig. 5(b) depicts the downward trend of training and validation loss, which is a positive sign. On the other hand, the increasing validation accuracy, decreasing validation loss over fifty epochs also indicate that the trained model is expected to perform well in the testing phase.

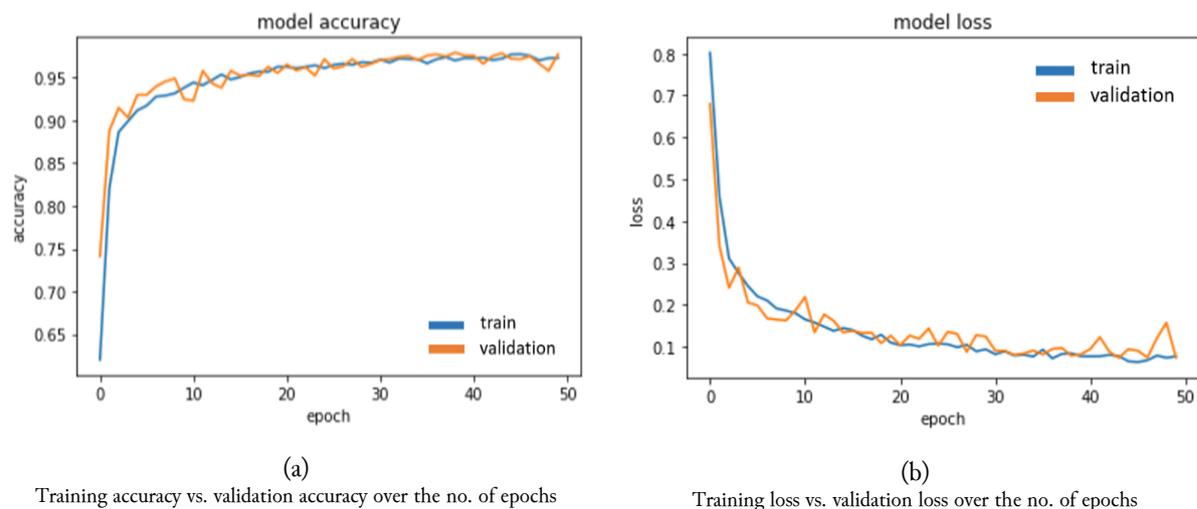


Fig. 5. ModCOVNN accuracy and loss comparison

Just like training accuracy, the proposed ModCOVNN has also outperformed VGG16 and VGG19 in terms of test accuracy. It has achieved 98% accuracy on the test images. Among the two VGG models, the VGG16 model has been better than the VGG19 model in terms of training accuracy. A similar trend

in performance has also been observed in these two models in terms of test accuracy. The VGG16 model has yielded 96% test accuracy, while VGG19 has yielded a test accuracy of 92%.

In this study, the main goal has been to screen out the COVID-19 patients from chest X-rays. Therefore, we are more interested in the proper recognition of COVID-19 cases. Classifying a chest x-ray of a COVID-19 patient as COVID-19 is more important than classifying a chest X-ray of a normal patient as normal. Another important matter is to consider is the class imbalance in the dataset. The COVID-19 class has way fewer number images than the other two classes. In this scenario, the model's overall accuracy can be higher, but it can perform poorly in the class where detection is more important. To determine whether our proposed model is performing consistently on the most important class, precision, recall, and f1-score of the model considering COVID-19 as the positive class have also been measured. To do that, a confusion matrix has been constructed first. The rows of the confusion matrix denote the ground truth class, and the columns contain the information about the predicted classes by ModCOVNN. Then, with the help of that, precision and Recall have been calculated. In Table 2, the confusion matrix of the proposed ModCOVNN has been shown. The model has achieved 98.4% recall and 98.14% precision. On the other hand, the VGG16 model has achieved 95.83% and 95.41% precision and recall, respectively. The VGG19 model has achieved the lowest precision and recall performance among the three models trained in this study. It has been able to demonstrate the precision of 92.5% and Recall of 91.67%. The f1-scores of the models depend on the models' precision and recall. In the f1-score, both precision and Recall are given equal importance. The ModCOVNN has achieved an f1-score of 98.27%. On the other hand, two VGG models have obtained f1-scores of 95.62% and 92.08%, respectively.

Table 2.

Confusion matrix for ModCOVNN model

		Ground truth class		
		<i>Covid</i>	<i>Normal</i>	<i>Pneumonia</i>
Predicted class	<i>Covid</i>	370	4	2
	<i>Normal</i>	5	742	9
	<i>Pneumonia</i>	2	15	781

This study has obtained outstanding performance in detecting COVID-19 from chest X-rays. Our proposed model can act as a decision support system for the screening of COVID-19 patients. Such a data-driven machine learning model can also help to mitigate the spread of Coronavirus by identifying the patients who are contracted with the virus and isolating them. In this study, we have proposed a novel convolutional neural network-based architecture that has superior performance to VGG16 and VGG19 models in the classification challenge. The reason why our model has worked so well is because of its not-so-deep and not so shallow architecture. The proposed ModCOVNN has six convolutional layers as opposed to thirteen and sixteen convolutional layers in VGG16 and VGG19 architectures, respectively. Not using the VGG16 and VGG19 models pre-trained on the ImageNet dataset might also have caused comparatively worse performance than the proposed ModCOVNN. Our proposed model has also been better than the other two deep learning models with respect to all of the four evaluation metrics that have been used in this study. The models of this study have also been able to tackle the imbalance problem of the used dataset. The models have been able to demonstrate good performance in the COVID-19 class despite having the lowest number of instances in that class. The proposed ModCOVNN has shown better precision and Recall in the COVID-19 class than the other two classes. Better performance in the desired class has also increased the reliability of the model's performance in a real-world environment. However, a balanced dataset can always help a machine learning model to achieve better performance. Adding more X-ray images to the COVID-19 class can improve a model's accuracy, precision, and Recall. Another way to handle the class imbalance problem is to use the Generative Adversarial Network (GAN). Using GAN, new samples of the minority class can be synthesized and can be added to the minority class to reduce the imbalance.

This study has also proved that using chest X-ray, proper detection of COVID-19 in a patient is possible. Chest X-rays can be used instead of the PCR test and antigen test. Using X-rays can also be helpful in the early diagnosis of COVID-19, which can save lives. Although the models generally have good performances, there are also a few limitations of this study. Only two of the popular convolutional neural network-based architectures (VGG16 and VGG19) have been used for the classification task. Other CNN-based architectures like ResNet [27], Inception [28], InceptionResNetV2 [29], Xception [30], DenseNet [31], Network in network [32] etc. have not been used. Using those deeper and sophisticated architectures could have added another dimension in the detection of COVID-19 from chest X-ray images. Other different methods of image classification, like ensemble methods, and hidden Markov models, have not been explored for X-ray image classification in this study.

This study has achieved better performances than many other existing works in this domain of recognizing COVID-19 from chest X-rays. In Table 3, the performance of this study has been compared with three other existing works. From Table 3, it is apparent that the proposed ModCOVNN has a better result than others with 98.08% test accuracy. In comparison, the other works have not been able to cross the 97% test accuracy mark. Among those three, the best performance achieving work has been done by I. D. Apostolopoulos and T. A. Mpesiana [15] with 96.78%. The other two have test accuracies of 89.60% [1] and 95% [26], respectively.

Table 3. Performance comparison of this study with other notable works

Works	Methods	Dataset	Test Accuracy
A. I. Khan et al. [1]	CoroNet (based on Xception CNN architecture)	1251 (Normal: 310, Pneumonia : 657, Covid : 284)	89.60%
I. D. Apostolopoulos and T. A. Mpesiana [15]	VGG-19 (model with transfer learning)	1442 (Normal: 504 Pneumonia: 714 Covid: 224)	96.78%
A. Waheed [26]	CovidGAN	1124 (Normal: 721 Covid: 403)	95%
This study	Proposed method (ModCOVNN)	8760 (COVID-19: 1089, Pneumonia: 2676, normal: 4905)	98.08%

#### 4. Conclusion

In this study, we have proposed a novel method to detect COVID-19 from chest X-rays. We have conducted the experiments in three phases. First, two popular CNN models have been used for image classification. After that, a new CNN-based architecture ModCOVNN has been introduced for COVID-19 detection. Finally, a comparative analysis of three models has been conducted to determine which model is more reliable in the described domain-specific task. The analysis shows that our proposed architecture has better performance amongst three trained machine learning models. Giving light to the limitations of this study, new works can be proposed to handle the scarcity of COVID-19 chest X-rays. Other techniques can be applied to increase the model performance and obtain a better decision support system to recognize COVID-19 in a patient. Another area worth exploring in this field of biomedical image classification is to work to increase the explainability and interpretability of deep learning models. How a deep learning model decides a given input can be confusing to the general stakeholders. Works addressing this issue would surely make deep learning models more usable and acceptable in various domain-specific tasks. However, this study has proven that our proposed ModCOVNN can be used in a reliable manner to screen out the COVID-19 patients easily.

### Declarations

**Author contribution.** All authors contributed equally to the main contributor to this paper. All authors read and approved the final paper.

**Funding statement.** None of the authors have received any funding or grants from any institution or funding body for the research.

**Conflict of interest.** The authors declare no conflict of interest.

**Additional information.** No additional information is available for this paper.

### References

- [1] 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," *Comput. Methods Programs Biomed.*, vol. 196, 2020, doi: [10.1016/j.cmpb.2020.105581](https://doi.org/10.1016/j.cmpb.2020.105581).
- [2] "Rolling updates on coronavirus disease (COVID-19)." Available at: <https://www.who.int/>
- [3] K. Tolksdorf, S. Buda, E. Schuler, L. H. Wieler, and W. Haas, "Influenza-associated pneumonia as reference to assess seriousness of coronavirus disease (COVID-19)," *Eurosurveillance*, 2020, doi: [10.2807/1560-7917.ES.2020.25.11.2000258](https://doi.org/10.2807/1560-7917.ES.2020.25.11.2000258).
- [4] G. Grasselli, A. Pesenti, and M. Cecconi, "Critical Care Utilization for the COVID-19 Outbreak in Lombardy, Italy," *JAMA*, 2020, doi: [10.1001/jama.2020.4031](https://doi.org/10.1001/jama.2020.4031).
- [5] T. Mahmud, M. A. Rahman, and S. A. Fattah, "CovXNet: A multi-dilation convolutional neural network for automatic COVID-19 and other pneumonia detection from chest X-ray images with transferable multi-receptive feature optimization," *Comput. Biol. Med.*, 2020, doi: [10.1016/j.combiomed.2020.103869](https://doi.org/10.1016/j.combiomed.2020.103869).
- [6] H. A. Rothan and S. N. Byrareddy, "The epidemiology and pathogenesis of coronavirus disease (COVID-19) outbreak," *Journal of Autoimmunity*. 2020, doi: [10.1016/j.jaut.2020.102433](https://doi.org/10.1016/j.jaut.2020.102433).
- [7] C. C. Lai, T. P. Shih, W. C. Ko, H. J. Tang, and P. R. Hsueh, "Severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2) and coronavirus disease-2019 (COVID-19): The epidemic and the challenges," *International Journal of Antimicrobial Agents*. 2020, doi: [10.1016/j.ijantimicag.2020.105924](https://doi.org/10.1016/j.ijantimicag.2020.105924).
- [8] E. Mahase, "Coronavirus covid-19 has killed more people than SARS and MERS combined, despite lower case fatality rate," *BMJ*, 2020, doi: [10.1136/bmj.m641](https://doi.org/10.1136/bmj.m641).
- [9] W. Guan *et al.*, "Clinical Characteristics of Coronavirus Disease 2019 in China," *N. Engl. J. Med.*, 2020, doi: [10.1056/nejmoa2002032](https://doi.org/10.1056/nejmoa2002032).
- [10] R. Punia, L. Kumar, M. Mujahid, and R. Rohilla, "Computer vision and radiology for COVID-19 detection," 2020, doi: [10.1109/INCET49848.2020.9154088](https://doi.org/10.1109/INCET49848.2020.9154088).
- [11] H. V Huff and A. Singh, "Asymptomatic Transmission During the Coronavirus Disease 2019 Pandemic and Implications for Public Health Strategies," *Clin. Infect. Dis.*, 2020, doi: [10.1093/cid/ciaa654](https://doi.org/10.1093/cid/ciaa654).
- [12] X. Xie, Z. Zhong, W. Zhao, C. Zheng, F. Wang, and J. Liu, "Chest CT for Typical Coronavirus Disease 2019 (COVID-19) Pneumonia: Relationship to Negative RT-PCR Testing," *Radiology*, 2020, doi: [10.1148/radiol.2020200343](https://doi.org/10.1148/radiol.2020200343).
- [13] A. Abbas, M. M. Abdelsamea, and M. M. Gaber, "Classification of COVID-19 in chest X-ray images using DeTraC deep convolutional neural network," *Appl. Intell.*, 2020, doi: [10.1007/s10489-020-01829-7](https://doi.org/10.1007/s10489-020-01829-7).
- [14] 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," *Sci. Rep.*, 2020, doi: [10.1038/s41598-020-76550-z](https://doi.org/10.1038/s41598-020-76550-z).
- [15] I. D. Apostolopoulos and T. A. Mpesiana, "Covid-19: automatic detection from X-ray images utilizing transfer learning with convolutional neural networks," *Phys. Eng. Sci. Med.*, 2020, doi: [10.1007/s13246-020-00865-4](https://doi.org/10.1007/s13246-020-00865-4).
- [16] A. Narin, C. Kaya, and Z. Pamuk, "Automatic detection of coronavirus disease (COVID-19) using X-ray images and deep convolutional neural networks," *arXiv*. 2020. doi: [10.1007/s10044-021-00984-y](https://doi.org/10.1007/s10044-021-00984-y)

- [17] T. Ozturk, M. Talo, E. A. Yildirim, U. B. Baloglu, O. Yildirim, and U. Rajendra Acharya, "Automated detection of COVID-19 cases using deep neural networks with X-ray images," *Comput. Biol. Med.*, 2020, doi: [10.1016/j.combiomed.2020.103792](https://doi.org/10.1016/j.combiomed.2020.103792).
- [18] X. Xu *et al.*, "A Deep Learning System to Screen Novel Coronavirus Disease 2019 Pneumonia," *Engineering*, 2020, doi: [10.1016/j.eng.2020.04.010](https://doi.org/10.1016/j.eng.2020.04.010).
- [19] O. Stephen, M. Sain, U. J. Maduh, and D. U. Jeong, "An Efficient Deep Learning Approach to Pneumonia Classification in Healthcare," *J. Healthc. Eng.*, 2019, doi: [10.1155/2019/4180949](https://doi.org/10.1155/2019/4180949).
- [20] P. K. Sethy, S. K. Behera, P. K. Ratha, and P. Biswas, "Detection of coronavirus disease (COVID-19) based on deep features and support vector machine," *Int. J. Math. Eng. Manag. Sci.*, 2020, doi: [10.33889/IJMEMS.2020.5.4.052](https://doi.org/10.33889/IJMEMS.2020.5.4.052).
- [21] E. E. D. Hemdan, M. A. Shouman, and M. E. Karar, "COVIDX-Net: A Framework of Deep Learning Classifiers to Diagnose COVID-19 in X-Ray Images," *arXiv*. 2020. Available at: <https://ui.adsabs.harvard.edu/>
- [22] L. Li *et al.*, "Using Artificial Intelligence to Detect COVID-19 and Community-acquired Pneumonia Based on Pulmonary CT: Evaluation of the Diagnostic Accuracy," *Radiology*, 2020, doi: [10.1148/radiol.2020200905](https://doi.org/10.1148/radiol.2020200905).
- [23] X. Xu *et al.*, "Deep learning system to screen coronavirus disease 2019 pneumonia," *arXiv*. 2020. doi: [10.1016/j.eng.2020.04.010](https://doi.org/10.1016/j.eng.2020.04.010)
- [24] L. Brunese, F. Mercaldo, A. Reginelli, and A. Santone, "Explainable Deep Learning for Pulmonary Disease and Coronavirus COVID-19 Detection from X-rays," *Comput. Methods Programs Biomed.*, 2020, doi: [10.1016/j.cmpb.2020.105608](https://doi.org/10.1016/j.cmpb.2020.105608).
- [25] K. Simonyan and A. Zisserman, "VGG-16," *arXiv Prepr.*, 2014. Available at: [Google Scholar](https://scholar.google.com/).
- [26] A. Waheed, M. Goyal, D. Gupta, A. Khanna, F. Al-Turjman, and P. R. Pinheiro, "CovidGAN: Data Augmentation Using Auxiliary Classifier GAN for Improved Covid-19 Detection," *IEEE Access*, 2020, doi: [10.1109/ACCESS.2020.2994762](https://doi.org/10.1109/ACCESS.2020.2994762).
- [27] K. He, X. Zhang, S. Ren, and J. Sun, "ResNet," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2016. Available at: [Google Scholar](https://scholar.google.com/).
- [28] C. Szegedy *et al.*, "Going deeper with convolutions," 2015, doi: [10.1109/CVPR.2015.7298594](https://doi.org/10.1109/CVPR.2015.7298594).
- [29] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, "Inception-v4, inception-ResNet and the impact of residual connections on learning," 2017. Available at: [Google Scholar](https://scholar.google.com/).
- [30] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," 2017, doi: [10.1109/CVPR.2017.195](https://doi.org/10.1109/CVPR.2017.195).
- [31] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," 2017, doi: [10.1109/CVPR.2017.243](https://doi.org/10.1109/CVPR.2017.243).
- [32] M. Lin, Q. Chen, and S. Yan, "Network in network," 2014. Available at: [Google Scholar](https://scholar.google.com/).