

Research Article

COVID-19 and Pneumonia Diagnosis in X-Ray Images Using Convolutional Neural Networks

Rahib H. Abiyev  and **Abdullahi Ismail** 

Applied Artificial Intelligence Research Centre, Department of Computer Engineering, Near East University, North Cyprus, Mersin-10, Nicosia, Turkey

Correspondence should be addressed to Rahib H. Abiyev; rahib.abiyev@neu.edu.tr

Received 28 April 2021; Revised 6 July 2021; Accepted 5 November 2021; Published 24 November 2021

Academic Editor: Ali Ahmadian

Copyright © 2021 Rahib H. Abiyev and Abdullahi Ismail. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper proposes a Convolutional Neural Networks (CNN) based model for the diagnosis of COVID-19 and non-COVID-19 viral pneumonia diseases. These diseases affect and damage the human lungs. Early diagnosis of patients infected by the virus can help save the patient's life and prevent the further spread of the virus. The CNN model is used to help in the early diagnosis of the virus using chest X-ray images, as it is one of the fastest and most cost-effective ways of diagnosing the disease. We proposed two convolutional neural networks (CNN) models, which were trained using two different datasets. The first model was trained for binary classification with one of the datasets that only included pneumonia cases and normal chest X-ray images. The second model made use of the knowledge learned by the first model using transfer learning and trained for 3 class classifications on COVID-19, pneumonia, and normal cases based on the second dataset that included chest X-ray (CXR) images. The effect of transfer learning on model constriction has been demonstrated. The model gave promising results in terms of accuracy, recall, precision, and F1_score with values of 98.3%, 97.9%, 98.3%, and 98.0%, respectively, on the test data. The proposed model can diagnose the presence of COVID-19 in CXR images; hence, it will help radiologists make diagnoses easily and more accurately.

1. Introduction

COVID-19 is a respiratory infection that affects the human lungs, which has now been declared a pandemic that is affecting the entire globe. As of today, October 2, 2020, there were 34 million total cases, 23.9 million recoveries, and 1.02 million deaths reported to the World Health Organization [1]. The initial case of COVID-19 was detected in December 2019 in Wuhan, Hubei province, China [2], from where it started to propagate to other countries around the world. As the COVID-19 virus is transmittable, early detection is very important for both patients and the people around them, as the patient will get proper care, and other people will be protected. The best way to fight against the COVID-19 pandemic is the early diagnosis of patients infected by the virus as well to provide special care and treatments. Reverse transcription-polymerase chain reaction (RT-PCR) is commonly used in tests to diagnoses COVID-19, although this has low sensitivity in the early stage of the virus, which

could lead to further transmission [3]. This test kit is expensive and scarce, and therefore, for early diagnosis, chest X-ray (CXR) images and computer tomography (CT) scans are the best option for use in diagnosing any patient that shows symptoms of pneumonia.

Non-COVID-19 viral pneumonia is also one of the leading causes of death of people of young and old ages. According to the Centers for Disease Control and Prevention (CDC), over 1 million adult pneumonia patients are hospitalized, and almost 50,000 patients die every year from this disease in USA alone [4]. As stated by WHO, chest X-rays are the best available way of diagnosing pneumonia [5]. Pneumonia is a respiratory infection that affects the lungs that can be caused by bacteria, viruses, or fungi. Diagnosing pneumonia is considered a tedious task, even by expert radiologists, because its symptoms appear to be similar to other pathologies that affect the lungs. As a result, different algorithms have been designed for this purpose. Paper [6] used ANN for the identification of pneumonia

diseases. Paper [7] used NN, competitive NN, and a deep learning structure for the identification of chest diseases using the CXR dataset, where deep learning exhibited the best performance. CXR and CT are widely used for the diagnosis of lung-related repository diseases, especially pneumonia and COVID-19. Researches have shown the usefulness and benefits of using medical images in diagnosing pneumonia diseases and COVID-19. Feature extraction is one of the important aspects of the detection of pneumonia diseases. In [8], COVID-19 classification problem was solved using K-nearest neighbors (KNN) and support vector machines (SVM).

In recent years, deep learning models have proven to be a promising method in the field of medicine for the diagnosis of pathologies, including lung pathology, which is the focus of this study, and have also given very promising results in the diagnosis of other medical diseases [9–11]. A convolutional neural network was developed from the visual context studies neocognitron in 1980 by K. Fukushima [12]. In 1998, Yann Lecun, Leon Bottou, and Yoshua Bengio recorded a very important milestone in convolutional neural networks by introducing the architecture called LeNet-5 [13], which is now widely used for handwritten recognition tasks. For this reason, we used a convolutional neural network in this study to diagnose the presence of COVID-19 and pneumonia. As the diagnosis of those diseases can be a tedious task, even among expert radiologists, this study aims to help radiologists diagnose COVID-19 and pneumonia from chest X-ray images easily and within a short time.

Several different algorithms have been proposed to diagnose the presence of COVID-19 and pneumonia in chest X-ray images and computer tomography (CT) scan images using different approaches with deep learning models, some of which are described below.

Li et al. [2] used CheXNet, DenseNet, VGG19, MobileNet, InceptionV3, ResNet18, ResNet101, and squeezeNet architecture to train the 3-class classification using transfer learning. The paper used data augmentation techniques on the dataset consisting of chest X-rays from 423 COVID-19 cases, 1,485 viral pneumonia cases, and 1,575 normal cases, and their model attained an accuracy of 97.94%. Li et al. [3] proposed a network architecture called CovXNet to diagnose the presence of COVID-19, viral pneumonia, and bacterial pneumonia. Their dataset consisted of 1,583 normal X-ray images, 1,493 non-COVID-19 viral pneumonia X-ray images, 2,980 bacterial pneumonia X-ray images, and 305 COVID-19 X-ray images cases from different patients. Their model performance had an accuracy of 89.1%. Gunraj et al. [14] proposed a COVID-Net network to diagnose the presence of COVID-19 and non-COVID-19 pneumonia. They introduced a new dataset of 13,975 chest X-ray images from 13,870 patients, and their model attained a performance accuracy of 93.3%. However, the paper only presented accuracy as the performance metric for the 3-class classification. Han et al. [15] used an attention-based deep 3D multiple instance learning (AD3D-MIL) approach for the screening of COVID-19 pneumonia from other forms of viral pneumonia. The researchers used a dataset of computer tomography (CT) scans that included 230 CT scans of

COVID-19 from 79 patients, 100 CT scans of patients with pneumonia, and 130 CT scans from people who did not have pneumonia. They reported that their algorithm achieved an overall accuracy of 97.9%. Rajaraman et al. [16] presented an iteratively pruned deep learning model ensemble to detect COVID-19 in chest X-ray images. They trained two models in their research. The first one was trained to classify normal and abnormal chest X-Rays, while the second model was trained to classify COVID-19 and pneumonia cases by using the training weights of the first model with the help of the transfer learning method. They used the ensemble method to improve the prediction performance of their model and achieved an accuracy of 99.01%. Hammoudi et al. [17] presented tailored models for early-stage detection of COVID-19 pulmonary symptoms. Their models were trained with a dataset that included bacterial pneumonia, viral pneumonia, and normal chest X-ray images. Ko et al. [18] proposed a simple 2D deep learning framework called the first-track COVID-19 classification network (FCONet) to diagnose COVID-19 pneumonia on a single chest computer tomography (CT) scan image. They used the transfer learning approach with state-of-the-art deep learning models as the backbone for training the FCONet model. In all the pretrained FCONet models, ResNet50 FCONet had the highest performance results of 99.8%, 100%, and 99.87% for sensitivity, specificity, and accuracy on the test dataset, respectively.

In [19], image feature descriptors, feed-forward NN, and CNN that used COVID-19 CXR images for the identification of diseases were presented. [20] presented a deep convolutional neural network based on Xception architecture called CorNet for the detection of COVID-19 infections using CXR images. [21] presented a clinical predictive model that uses deep learning and laboratory data for estimating COVID-19 diseases. The model was tested with 18 laboratory findings from 600 patients. [22] presented deep transfer learning for detecting COVID-19 diseases using CXR and CT images. In [23], a hybrid CNN was presented using an optimization algorithm for diagnosing COVID-19. [24] presented a deep learning assisted method for the diagnosis of COVID-19. A comparative study of eight CNN models was presented. [25] used deep learning to detect COVID-19 on CXR. The system was composed of three phases. In the first phase, the presence of pneumonia was detected, then COVID-19 and pneumonia were determined, and lastly, the localization of diseases was performed. The paper used 6,523 CXR images and obtained an accuracy of 97%. [26] presented an AI system for predicting COVID-19 pneumonia using CXR images. [27] used transfer learning and semi-supervised adversarial detection to classify COVID-19 from CT images. [28] introduced CSHNET to classify COVID-19 using transfer learning and discriminate correlation analysis. They achieved the highest sensitivity of 98.3% and reported that CSHNET outperformed 12 state-of-the-art COVID-19 detection models. [29] presented a patch-based CNN to diagnose COVID-19 in CXR images; their model achieved state-of-the-art performance.

In this paper, we present a Convolutional Neural Network (CNN) model, which was developed to help in

detecting COVID-19 and pneumonia cases in X-ray images with the aim of facilitating early diagnosis and preventing transmission of the virus to other people. The main contributions of this paper include the following: the structure of CNN for detecting COVID-19 and pneumonia cases is proposed; the training of CNN with imbalanced data is demonstrated using random sampling of data with data augmentation; the effect of transfer learning on construction of the final model has been demonstrated; a CNN was designed to diagnose COVID-19; accuracy precision and recall evaluation metrics were used to tackle the imbalance problem.

To diagnose COVID-19, two different datasets were utilized, where one contained only pneumonia and normal chest X-ray images, and the other one contained COVID-19, pneumonia, and normal chest X-ray images. Two models were proposed: the first model was trained on pneumonia and normal cases chest X-ray images, while the second model made use of the knowledge that was learned from the first model and trained on COVID-19, pneumonia, and normal cases chest X-ray images. The transfer learning approach was utilized to transfer the weight/knowledge of the first model to the second model for COVID-19, pneumonia, and normal class classification.

The remainder of the paper is arranged as follows. The methodology and system architecture are presented in Section 2. In Section 3, we describe the experimental study and present the simulation results obtained in Section 4. Finally, we conclude our research work in Section 5 of the paper.

2. The Method

2.1. System Architecture. In this study, the design of the identification system that will correctly diagnose the presence of COVID-19 and viral pneumonia in X-ray images was carried out. The architecture of the designed system is summarized in Figure 1, which includes the following steps:

- (1) Preparation of the research dataset that includes pneumonia and normal X-ray images and partitioning of the images into training, validation, and testing sets.
- (2) Preprocessing of input images including scaling the images, data augmentation, and data resampling.
- (3) The use of constructed CNN structure for calculating the output of the targeted/proposed model using chest X-ray images of pneumonia and normal cases.
- (4) Calculation of output classes
- (5) Comparison of current outputs with the desired classes and calculation of the loss function.
- (6) Use of the loss function and training algorithm to adjust the parameters of CNN
- (7) Use of steps 3–6 for all datasets and all epochs.
- (8) The use of a pretrained/source model (first model), which was trained with a preprocessed dataset

including only pneumonia and normal X-ray images as the base model architecture for the second model implementation of transfer learning techniques.

- (9) Preparation of a research dataset that includes COVID-19, pneumonia, and normal X-ray images and partitioning the images as training, validation, and testing sets.
- (10) Preprocessing the input images including scaling the images, data augmentation, and data resampling.
- (11) The use of the constructed CNN structure for calculating the output of the targeted/proposed model using chest X-ray images of COVID-19, pneumonia, and normal cases.
- (12) Use of the training algorithm for updating the CNN parameters of the pretrained model.
- (13) Use of steps 10–12 for all datasets and all epochs.
- (14) The targeted model output is a decision-making part of the model architecture that determines whether the X-ray image is of COVID-19, pneumonia, or a normal case.

2.2. Transfer Learning. The transfer learning approach was used in developing the model. In the paper, two models were trained using two different datasets. The first model was trained on a dataset that contained only normal and pneumonia cases. The second model used the first model as the base model; in other words, it used the knowledge learned by the first model to train on the dataset that contained COVID-19, pneumonia, and normal cases. Here, we demonstrated the effect of transfer learning in the final model construction.

Transfer learning is the method of reusing machine learning model, which was previously trained with a large amount of data in a particular task, to train a new classifier of a similar or different task by adjusting the model hyperparameters or freezing all or some of the layers of the already trained model, as shown in Figure 2. Transfer learning helps with the training of a convolutional neural network with a small-sized dataset. In transfer learning, the new model that is built with the pretrained model does not need to be trained with a large dataset to perform well. Since all the basic features are already learned in the pretrained model, the time required for training is significantly less than that needed to train from scratch or train without using the transfer learning method. The memory and computational resources will also be reduced compared to training from scratch. Many pretrained algorithms are trained with large datasets like the ImageNet dataset [30], which has over 15 million images from around 22,000 categories. Those pretrained algorithms are largely used as the base in the transfer learning method. However, in this research, the pretrained model used was trained with the chest X-ray images (Pneumonia) dataset [31], not the ImageNet dataset.

Transfer learning is the most widely used method in computer vision-related tasks. Transfer learning helps

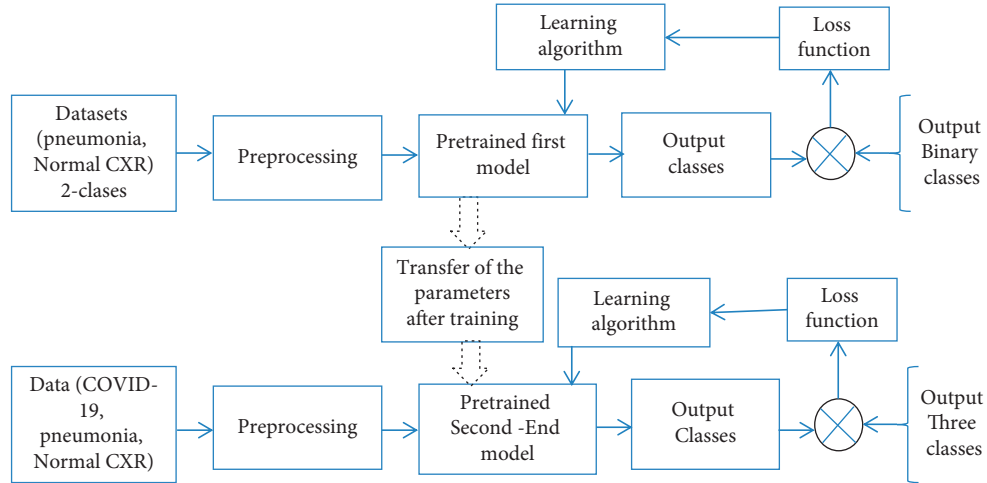


FIGURE 1: The overall architecture of the proposed system.

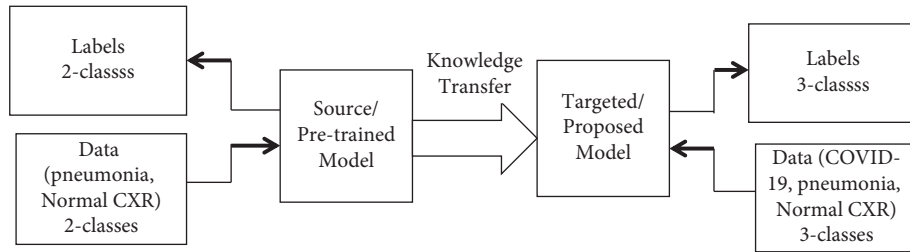


FIGURE 2: Transfer learning structure.

reduce the training time of a model as well as the computational cost, prevents or reduces model overfitting, allows the training of a large CNN with a small amount of data, and increases/boosts the performance of a model.

The network parameters are determined as a result of training the network.

3. Experimental Study

3.1. Dataset. The datasets that were used in this research study were downloaded from the Kaggle dataset repository. The first dataset (CXR images (Pneumonia)) [31] contains 5,856 X-ray images in JPEG format, anterior-posterior chest X-ray images of pediatric patients ranging from one to five years old, which were selected from the Guangzhou Women and Children's Medical Center. The chest X-ray images were screened to remove all X-ray images that were unreadable or have low-quality scans. Two expert radiologists checked and evaluated the diagnosis (Label). The third expert radiologist also rechecked the validity of the labels to avoid errors. The dataset was categorized into three parts, namely, the training, validation, and testing part, and each part was subcategorized into two classes, normal and pneumonia class. The dataset contained bacterial and viral pneumonia, which were considered and labeled as pneumonia class. A total of 4,185 chest X-ray images were used as training data, 1,047 chest X-ray images for validation, and 624 images for testing, where 390 were pneumonia cases and 234 were

normal cases. All the chest X-ray images from these three parts were both from pneumonia and normal classes. The second dataset (COVID-19 Radiography database) contained COVID-19, pneumonia, and normal cases chest X-ray images. This dataset was created by a team of researchers from Qatar University, the University of Dhaka in Bangladesh, and collaborators from Pakistan and Malaysia in collaboration with medical doctors [2]. The dataset contains 219 COVID-19 chest X-ray images, 1,341 normal chest X-ray images, and 1345 viral pneumonia chest X-ray images cases. Information about the datasets is given in Table 1 and 2. A sample of chest X-ray images from COVID19, pneumonia, and normal classes is shown below (Figure 3).

3.2. Data Imbalance Problem. As the dataset used in this research was imbalanced, and the convolutional neural network used can be severely affected by this imbalance by creating bias among the classes, we used the method of transfer learning explained in Section 2.2 of this paper to overcome this problem of model bias [31]. We used the data augmentation technique to deal with this problem, as well as a sampling approach, to address the imbalanced dataset. Through data resampling, we improved the frequency of minority classes and decreased the number of majority classes. We attempted to take the same number of instances for both classes. These operations are called oversampling and undersampling. In this way, by using random sampling,

TABLE 1: The first dataset split for training, validation, and testing.

Cases	Training data	Validation data	Test data	Total
Normal	1,067	282	234	1,583
Pneumonia	3,118	765	390	4,273
Total	4,185	1,047	624	5,856

TABLE 2: The second dataset split for training, validation, and testing.

Cases	Training sets	Validation set	Testing set	Total
Normal	1,082	118	141	1,341
COVID-19	186	12	21	219
Pneumonia	1,084	132	129	1,345
Total	2,352	262	291	2,905

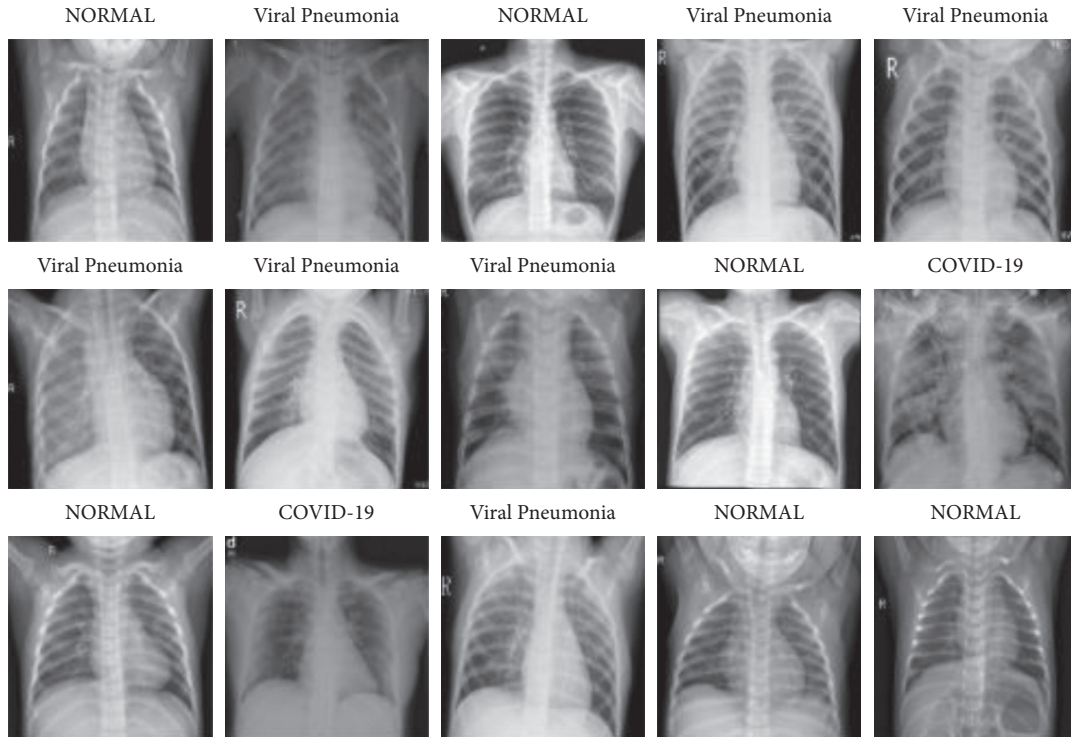


FIGURE 3: COVID-19/pneumonia/normal chest X-ray images.

the number of minority classes is increased to solve the data imbalance problem. In this approach, we do not lose information. We trained two models, where the first model was trained with the pneumonia dataset, and its weight was used as the initial weight in the second model that diagnoses the presence of COVID-19. This helped the model easily learn the important features of each class with a small number of images despite the data imbalance. Evaluation metrics were the second method used in tackling the data imbalance problem; we used precision and recall to evaluate our model. We adjust our model hyperparameter in such a way that the model avoids bias by making the precision and recall as close as possible.

3.3. CNN Model. The main aim of this research is to develop a model that will correctly diagnose the presence of COVID-19 and viral pneumonia in X-ray images. The transfer

learning approach was used in developing the model, and two models were trained in two different datasets, where the first model was trained on a dataset that contained only normal and pneumonia cases, and the second model used the first model as the base model; in other words, it used the knowledge learned by the first model to train on the dataset that contained COVID-19, pneumonia, and normal cases. The training and testing of the model were performed using Keras and TensorFlow framework in a Python programming language with a tensor processing unit (TPU) as an accelerator in Kaggle kernel.

CNN, which is used for the detection of diseases, includes a set of layers that implement feature extraction and classification operations. CNN is a kind of neural network that consists of convolutional layers, activation functions, pooling and flattening layers, which are used in the extraction of features from the input data and fully connected

networks that are used for classification [32]. Figure 4 depicts the structure of the CNN used in this research.

The convolutional layers of the network are used to learn and extract features from an image. A convolutional layer is where the input image pixels values that have some weight and height are multiplied or convoluted with convolutional filters or kernels. The convolutional result output dimension will have fewer dimensions than before the convolution operation. Filters, stride, and padding types are the hyperparameters of convolutional layers that have to be set. The convolution operation performed by the convolutional layer can be represented by the mathematical formula below, where X is the input, ω is the filter's parameter, and b is the bias.

$$\sum_{i=0}^{i=n} X * \omega + b. \quad (1)$$

Here, two or more convolutional layers (see Figure 4) can be used for the extraction of features. The output dimension of the convolution of input image pixels with a convolution filter or kernel can be calculated using the below mathematical formula, where n is the image dimension, p is the padding type, 0 denotes no padding and 1 padding, f is the filter or kernel size, and s is the stride value.

$$\lfloor \frac{n + 2p - f}{s} + 1 \rfloor. \quad (2)$$

The activation function is a function that decides whether an output of a neural network layer will pass to the next layer or not. The activation function is added at the end of each layer of a neural network (NN). There are linear and nonlinear activation functions, which are used to decide on the output of each neural network layer. Nonlinear activation functions are the functions that are mostly used in neural networks and deep learning algorithms. The ReLU activation function is used in the output activation y of the convolutional layers, which can be represented mathematically as $y = \text{ReLU}(y)$. These output activations are then inputted to the pooling layer.

The pooling layer is a layer that comes after the convolutional layer, where its main purpose is to shrink or reduce the input image size. The number of parameters, memory usage, and computational power will be reduced by the pooling layer, and it also helps in reducing the risk of model overfitting. As in the convolutional layers, the pooling layers also have some hyperparameters that must be set. The size of the pooling layer, the padding, and the stride are the hyperparameters in each pooling layer that must be set. Pooling layers are not like convolutional layers that do some convolutional operations as they just use aggregate functions to aggregate the input. The commonly used aggregation functions are max aggregation, min aggregation, and average aggregation, also known as max-pooling, min-pooling, and average-pooling, respectively.

The CNN flattening layer is used to restructure the obtained output activation into a one-dimensional array. The output of this layer can be formulated as $u_{\text{flatten}}^l = \text{flatten}(u_{ij}^l)$. This output vector is given to the Dense

layers as its input $D(u_{\text{flatten}}^l)$. This layer is one of the most used layers in a neural network; it is fully connected by neurons in the network layers, where each neuron is connected to all the previous layer neurons in the network. The input of each dense layer comes from the previous layer of the network. It performs the dot product of the input tensor x and the weight matrix w , and then the bias vector b is added, and the activation is performed on the output value, that is, $y = f(wx + b)$. A softmax activation function is used in the last layer of a neural network. This activation function gives a value to the input tensors in relation to their weight, and if all the values are added, they will sum up to one. The softmax activation function is most frequently used in both binary and multiclass classification problems. The mathematical equation for the softmax activation function is given as

$$f(x) = \frac{e^x}{\sum_{j=1}^n e^{x_j}}. \quad (3)$$

The CNN is trained using the output signals and patterns of an image. A loss function is used for training the network.

$$L = \frac{1}{N} \sum_{i=1}^N E(\theta; o^{(i)}, p^{(i)}). \quad (4)$$

Here, θ is the parameter of CNN, o - outputs, and p -target patterns. The network parameters are trained with the loss function using a stochastic gradient algorithm. The Adam optimizer with transfer learning and categorical cross-entropy loss is used in the model training.

Adam stands for Adaptive Moment Estimation. This optimizer encapsulates the concepts of the momentum optimizer and RMSProp optimizer by monitoring both the exponential decay average of the previous gradient and the exponential decaying average of the previous square gradient, respectively. The Adam optimizer is the best among the other optimizers, because the algorithm is very fast and also converges rapidly [33]. The equation below shows the Adam optimization algorithm.

$$\theta_t \leftarrow \frac{\theta_{t-1} - \alpha \cdot m_t}{(\sqrt{\vartheta_t} + \varepsilon)}, \quad (5)$$

where m_t and ϑ_t are bias-corrected first moment and second-row moment estimates, respectively, $\varepsilon = 10^{-8}$ is a numerical term, α is the learning rate, and θ_t is the weight parameter in the t -th iteration. For more details, the reader may refer to [33].

Model hyperparameters help estimate model parameters. The right combination of hyperparameters allows the model to maximize model performance. They can often be set by heuristics or manually. We searched for the best value by trial and error. In this way, we constructed a distribution for each parameter by trying different combinations of hyperparameters values manually to find the best parameters for the high model performance.

Table 3 shows the overall architecture of the proposed model. The model architecture received an X-ray image data of size $180 \times 180 \times 3$ as input data; it has two convolutional



FIGURE 4: CNN structure.

TABLE 3: The proposed model architecture.

Layer (type)	Output shape	Parameters #
Conv2d_2 (conv2d)	(180, 180, 16)	448
Conv2d_3 (conv2d)	(180, 180, 16)	2320
Max_pooling2d_1	(90, 90, 16)	0
Sequential (sequential)	(45, 45, 32)	2160
Sequential_1 (sequential)	(22, 22, 64)	7392
Sequential_2 (sequential)	(11, 11, 128)	27072
Dropout (dropout)	(11, 11, 128)	0
Sequential_3 (sequential)	(5, 5, 256)	103296
Dropout_1 (dropout)	(5, 5, 256)	0
Flatten (flatten)	(6400)	0
Sequential_4 (sequential)	(512)	3279360
Sequential_5 (sequential)	(128)	66176
Sequential_6 (sequential)	(64)	8512
Dense_7 (dense)	(512)	33280
Dense_8 (dense)	(3)	1539

layers followed by one max-pooling layer, where the resulting output of the convolutional layer called activation is reduced, then three sequential layers, followed by a MaxPooling layer. Each sequential layer before the flattening layer includes two convolution layers: the Batch normalization layer and the MaxPooling layer. The sequential layer is followed by a dropout layer, then the sequential layer, and another dropout layer. The outputs of the dropout layer are entered into the Flatten layer. The latter is followed by three sequential layers, one dense layer, and then an output layer, which performs the classification. Each sequential layer after the flatten layer includes two dense layers, one batch normalization layer, and one dropout layer. The output layer that makes the classification uses a softmax activation

function. All other layers use the “ReLU” activation function, which is responsible for the probabilistic decision on the input images of all the classes that the network trained on filter of size 3 for all the convolutional layers. The Adam optimizer with default parameters and categorical cross-entropy loss is used, and in the model training, 20 epochs resulted from the use of the early stopping method for avoiding model overfitting, where a default learning rate of 0.001 and batch size of 32 were used.

3.4. Data Augmentation. Data augmentation is the process of transforming the training data to generate more artificial training data from the original data. In this research study, the focus is on problems related to the computer vision domain; therefore, we used image data augmentation techniques. To augment data, or in other words, to generate artificial training image data from original training data, especially the minority class, some simple image processing techniques were used on the image data, including cropping, shifting, zooming, flipping, color alteration, and more, as shown in Figure 5.

Deep learning algorithms are known to consume large amounts of data. For any deep learning algorithm to produce an accurate result, it needs a huge amount of data to train from, and this high demand for data by deep learning algorithms makes it difficult for them to be applied in areas or domains where data is expensive to obtain. All domains do have a large amount of data to train a deep learning algorithm, but in this research, we used the data augmentation technique to bridge the gap. This helped train a deep learning algorithm on a small amount of data to achieve a good performance without overfitting the majority class of the data.

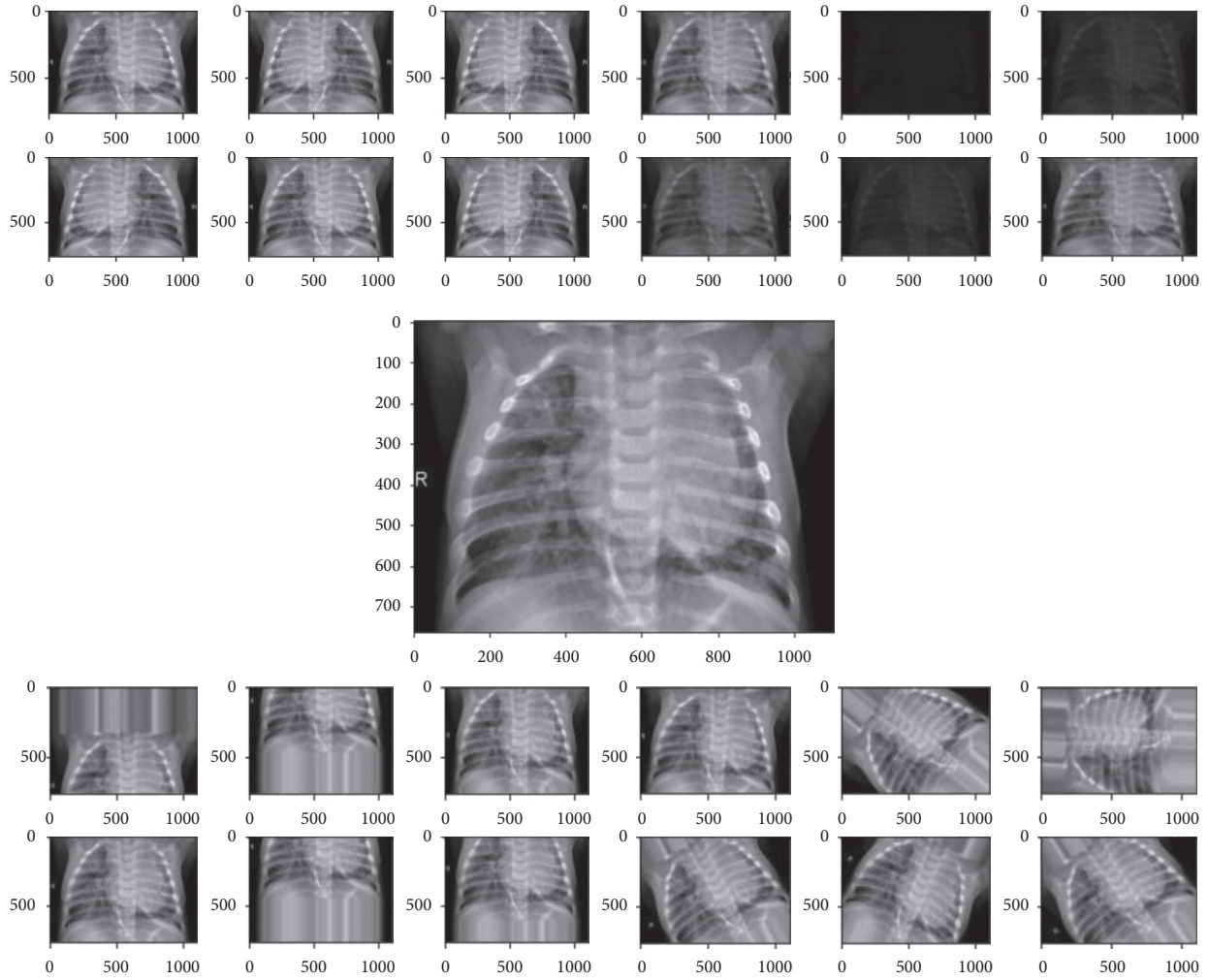


FIGURE 5: Original image at the center, color transform image at bottom left, rotate image at bottom right, flipped image at top left, and color transform image at top right.

3.5. Evaluation Metrics. Different evaluation metrics are used to evaluate the performance of different models regarding the problem at hand. Some evaluation metrics are superior for measuring the performance of regression models, while others are more suitable for classification models. As previously mentioned, many different types of evaluation metrics exist, but in this research, accuracy, recall, precision, F1 score, and specificity were used in evaluating the models' performances.

Accuracy is an evaluation metric that is used for measuring the performance of classification or regression algorithms. Accuracy can be a problematic or misleading performance metric when used to evaluate a model that is trained on unbalanced data. For this evaluation metric to provide a good and reliable performance measure, the data to be used in training the model must be balanced. Accuracy is computed by summing up the true positive (TP) and true negative (TN) classes divided by the summation of true positive, true negative, false positive (FP), and false negative (FN) classes, as shown in the formula below.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}. \quad (6)$$

Recall is another evaluation metric that is used to measure the performance of the classifier. Recall is a correctly classified class from the classification model. Recall is computed by dividing the true positive class by the sum of the true positive class and false-negative class, as shown in the formula below.

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}. \quad (7)$$

Precision is also an evaluation metric usually used together with the recall evaluation metric to measure the performance of classification algorithms. Precision is a positive prediction result that corresponds to the correctly classified class. Precision is computed by dividing the true positive class by the sum of the true positive and false positive classes, as shown in the formula below.

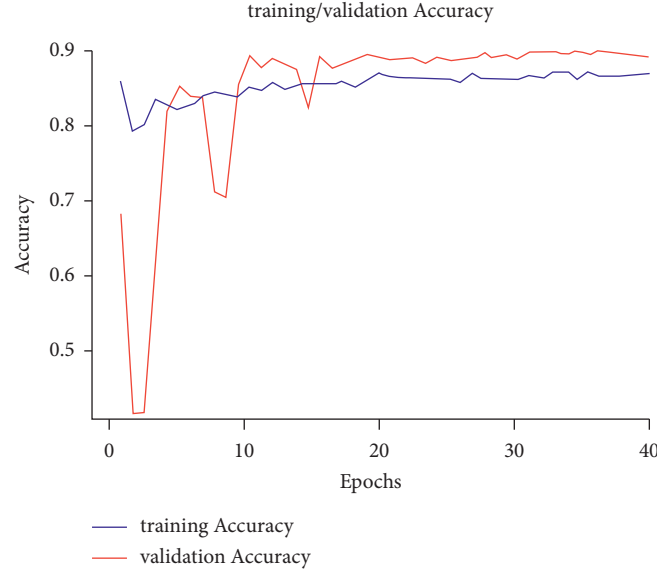


FIGURE 6: First model accuracy per epoch.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}. \quad (8)$$

The F1 score evaluation metric is used to measure the performance of a classifier by combining the recall and precision evaluation metrics as one single performance measure evaluation metric. The F1 score evaluation metric is computed by multiplying the product of the precision and recall by 2 and dividing by the sum of recall and precision evaluation metrics. The formula below shows how the F1 score evaluation metric is computed.

$$\text{F1 Score} = \frac{2 \times \text{precision} \times \text{recall}}{\text{precision} + \text{recall}}. \quad (9)$$

4. Results

In this section of the paper, we present our research test results, which include the model performance metrics and figures showing the activation map, the Grad_CAM of our model, and we also compare our results with some other state-of-the-art results in the area. Before the training, all images were resized to $180 \times 180 \times 3$ and then normalized for easy and faster training. To avoid overfitting, we used batch normalization layers and dropout layers in our model architecture, and also the transfer learning method. We used the data augmentation technique to deal with the training problem, and simple image processing techniques like cropping, color transformation, horizontal flipping, and rotation were used to produce artificial data. The training was accomplished with imbalanced data. We used a data sampling approach to handle the imbalanced data set. By data resampling, we improved the frequency of minority classes. By applying the random data sampling approach, we attempted to take the same number of instances for both classes. Using transfer learning, the learning time of the model

is decreased, the computational cost is reduced, model overfitting is prevented or reduced, the training of a large CNN with a small amount of data is allowed, and the performance characteristics of the model are also improved.

Figures 6 and 7 show the training accuracy and loss for the first model, respectively. The model is trained to the extent that it can learn the important basic X-ray image features and can differentiate pneumonia cases from normal X-ray images with good accuracy. However, the model plot below shows signs of model underfitting, and this is because the first dataset used in the model training was small. Therefore, this issue can be resolved by adding the training data, and further tuning the hyperparameters of the model.

Figures 8 and 9 present the training and validation plots of the model accuracy and loss against each epoch. As shown on the plot, the model achieved high performance with only 20 epochs. It can be clearly seen that the model accuracy was high, and the model loss was very low. This is because of the use of our first model by transfer learning method, which helped reduce the training time and also attain high performance with a small amount of training data and number of epochs.

Table 4 shows the results of the first and second models on the test data. The results given in the table show the model's evaluation using test data. The table provided the recall, precision, F1 score, and accuracy, which were 0.96, 0.82, 0.88, and 0.85, obtained by the first model, and 0.979, 0.982, 0.980, and 0.982, obtained by the second model, respectively. The second model performance can be maximized by tuning the model hyperparameters, the addition of more training data, and using data augmentation techniques. However, with this result, we are able to get what we need to build the main model that classifies COVID-19 and pneumonia cases from chest X-ray images.

Figure 10 shows the activation map of the convolutional layers of the network. The activation maps show what a

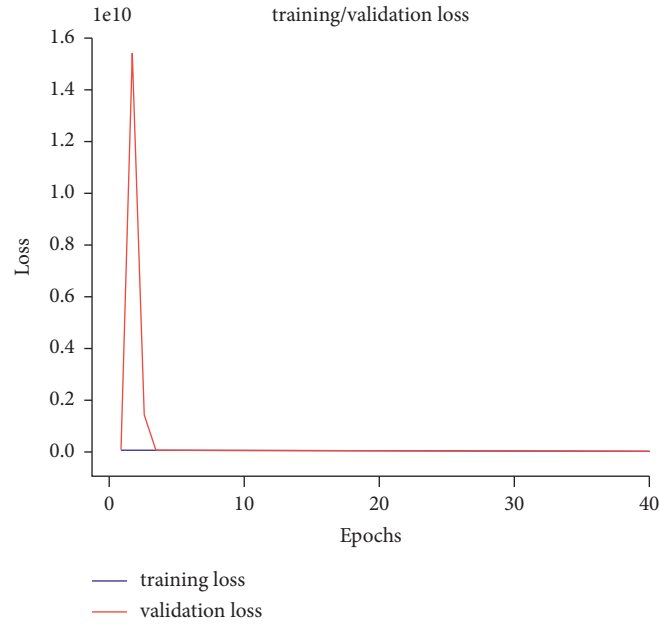


FIGURE 7: First model loss per epoch.



FIGURE 8: Second model accuracy per epoch.

network learned at a particular layer. For instance, the Conv2d_6 layer shows the high-level features learned by the network, and the sequential_30 layer shows low-level features, which are more specific features of the classification classes in the data. The deeper the network, the more the specific features learned for each class by the network.

Figure 11 shows the Grad_CAM of COVID-19 and viral pneumonia cases. The Grad_CAM helps us visualize where exactly the model looks to perform the prediction on each X-ray image. The first 2 X-ray images are for COVID-19 cases, and the last 2 X-ray images are for viral pneumonia cases. In all the X-ray images, the part that is rainbow/yellowish is the most important part of the image used by the

model in making decisions, while the parts with purple color are less important for the model in terms of decision making.

The results of the simulation are compared with the results of other deep learning structures used for the detection of COVID-19 in the same dataset as we used in our research. Table 5 presents a comparison of the performances of different models based on deep learning structures. The comparative results demonstrate the efficiency of the designed CNN models in diagnosing COVID-19.

This research is limited to the use of only a small number of chest X-ray images from COVID-19 cases due to the lack of availability of open access CXR images. Even with the small amount of CXR images that were available, our



FIGURE 9: Second model loss per epoch.

TABLE 4: Models test results.

	Epoch	Recall	Precision	F1 score	Accuracy	Loss
First model	46	0.96	0.82	0.88	0.85	0.44
Second model	20	0.979	0.982	0.980	0.982	0.045

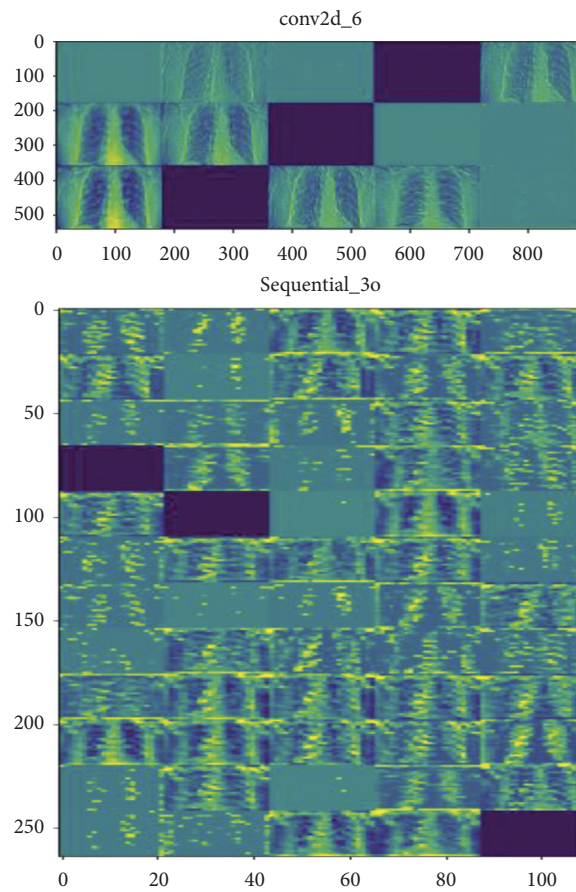


FIGURE 10: The activation map in some convolutional layers of the network.

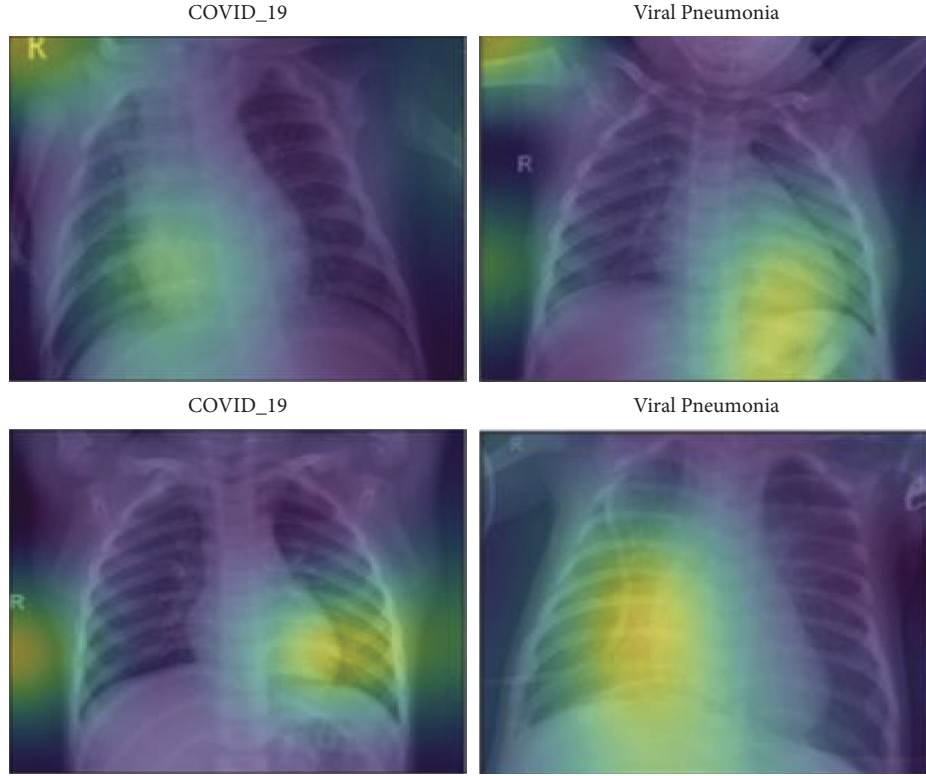


FIGURE 11: Shows the Grad_CAM of COVID-19 and viral pneumonia cases.

TABLE 5: Performance comparison of different models trained on the same data.

Authors (year)	Recall (%)	Precision (%)	Specificity (%)	F1 score (%)	Accuracy (%)
Chowdhury et al. (2020) [34]	97.94	97.95	98.80	97.94	97.94
Cavallo et al. (2020) [35]	—	—	90	—	90.8
Lin & Lee. (2020) [36]	92.6	89.7	—	—	93.1
Rajpal et al. (2020) [37]	—	—	—	95	94.4
Echtioui et al. (2020) [38]	88.33	91	—	89.66	91.34
Pham (2020) [39]	—	95	98	96	95
Neural et al. (2020) [40]	—	—	100	—	96.9
Alotaibi (2020) [41]	97.42	97.42	—	97.23	98.3
This paper	97.94	98.27	—	98.0	98.28

proposed model was able to diagnose the presence of COVID-19 and pneumonia cases with good performance. Likewise, the comparison of our results with those of other researchers with other research results is limited to only studies that used the same dataset, but not the same experimental setup like the hardware used in training the models.

5. Conclusion

The paper was aimed at developing a convolutional neural network (CNN) model that will help in the early diagnosis of COVID-19 and non-COVID-19 viral pneumonia cases using chest X-ray images in the difficult period caused by the pandemic. Due to the lack of COVID-19 chest X-ray image case data, we have shown how the transfer learning approach could be used to bridge this gap. As explained, two convolutional neural network (CNN) models were trained on

two different datasets; the first model was trained for binary classification (pneumonia/normal) on the first dataset that only contained pneumonia cases and normal chest X-ray images. On the other hand, with the help of the transfer learning method, the second model used the first model as the base model and was trained on the second dataset that contained COVID-19, pneumonia, and normal cases chest X-ray images for 3-class classification (COVID-19/pneumonia/normal). After the model was implemented, the results achieved in terms of the diagnosis of COVID-19 and pneumonia were 98.3%, 97.9%, 98.3%, and 98.0% for accuracy, recall, precision, and F1_scores, respectively; hence, the proposed model proved to be efficient in diagnosing COVID-19 and pneumonia cases. A convolutional neural network (CNN) is known as a black box; therefore, a class activation map of some convolutional layers was shown to help understand what the model learns at a particular layer. Grad_CAM was also shown to help us identify where exactly

the model is looking at on the chest X-ray images to perform the classification task. At the end of the paper, some of the state-of-the-art results were compared with the results of this research, and this work archived higher performance for accuracy, recall, precision, and F1_score than the others.

Data Availability

The data used to support the findings of this study have been deposited in the KAGGLE repository ([DOI: 10.17632/rschjbr9sj.2])

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

- [1] WHO, Coronavirus Disease (COVID-19) Dashboard, WHO, Geneva, Switzerland, <https://covid19.who.int/>.
- [2] Q. Li, X. Guan, P. Wu et al., "Early transmission dynamics in Wuhan, China, of novel coronavirus-infected pneumonia," *New England Journal of Medicine*, vol. 382, no. 13, pp. 1199–1207, 2020.
- [3] L. Li, L. Qin, Z. Xu et al., "Using artificial intelligence to detect COVID-19 and community-acquired pneumonia based on pulmonary CT: evaluation of the diagnostic accuracy," *Radiology*, vol. 296, no. 2, pp. E65–E71, 2020.
- [4] Pneumonia Home CDC, <https://www.who.int/news-room/fact-sheets/detail/pneumonia>.
- [5] World Health Organization, *Standardization of Interpretation of Chest Radiographs for the Diagnosis of Pneumonia in Children/World Health Organization Pneumonia Vaccine Trial Investigators' Group*, WHO, Geneva, Switzerland, 2001.
- [6] S. Pattar, "Detection and classification of lung disease—pneumonia and lung cancer in chest radiology using artificial neural network," *International Journal of Scientific and Research Publications*, vol. 5, no. 1, pp. 2250–3153, 2015.
- [7] R. H. Abiyev and M. K. S. Ma'aitah, "Deep convolutional neural networks for chest diseases detection," *Journal of Healthcare Engineering*, vol. 2018, Article ID 4168538, 11 pages, 2018.
- [8] R. M. Pereira, L. O. Teixeira, D. Bertolini, C. N. Silla, and Y. M. G. Costa, "Covid-19 identification in chest X-ray images on flat and hierarchical classification scenarios," *Computer Methods and Program in Biomedicine*, vol. 194, 2020.
- [9] R. Abiyev, M. Arslan, J. B. Idoko, B. Sekeroglu, and A. Ilhan, "Identification of epileptic eeg signals using convolutional neural networks," *Applied Science*, vol. 10, no. 12, 2020.
- [10] R. H. Abiyev and M. Arslan, "Head mouse control system for people with disabilities," *Expert System*, vol. 37, no. 1, pp. 1–14, 2020.
- [11] I. J. Bush, R. Abiyev, and M. Arslan, "Impact of machine learning techniques on hand gesture recognition," *Journal of Intelligent & Fuzzy Systems*, vol. 37, no. 3, pp. 4241–4252, 2019.
- [12] K. Fukushima, "Neocognitron: a self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position," *Biological Cybernetics*, vol. 36, no. 4, pp. 193–202, 1980.
- [13] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.
- [14] H. Gunraj, L. Wang, and A. Wong, "COVIDNet-CT: a tailored deep convolutional neural network design for detection of COVID-19 cases from chest CT images," *Frontiers in Medicine*, vol. 7, pp. 1–12, 2020.
- [15] Z. Han, B. Wei, Y. Hong et al., "Accurate screening of COVID-19 using attention-based deep 3D multiple instance learning," *IEEE Transactions on Medical Imaging*, vol. 39, no. 8, pp. 2584–2594, 2020.
- [16] S. Rajaraman, J. Siegelman, P. O. Alderson, L. S. Folio, L. R. Folio, and S. K. Antani, "Iteratively pruned deep learning ensembles for COVID-19 detection in chest X-Rays," *IEEE Access*, vol. 8, pp. 115041–115050, 2020.
- [17] K. Hammoudi, H. Benhabiles, M. Melkemi et al., "Deep learning on chest X-ray images to detect and evaluate pneumonia cases at the era of COVID-19," *Journal of Medical Systems*, pp. 1–10, 2021.
- [18] H. Ko, H. Chung, W. S. Kang et al., "COVID-19 pneumonia diagnosis using a simple 2d deep learning framework with a single chest CT image: model development and validation," *Journal of Medical Internet Research*, vol. 22, no. 6, pp. 1–13, 2020.
- [19] S. Varela-Santos and P. Melin, "A new approach for classifying coronavirus COVID-19 based on its manifestation on chest X-rays using texture features and neural networks," *Information Sciences*, vol. 545, pp. 403–414, 2021.
- [20] A. I. Khan, J. L. Shah, and M. Bhat, "CoroNet: a deep neural network for detection and diagnosis of covid-19 from chest X-ray images," *Computer Methods and Programs in Biomedicine*, vol. 196, 2020.
- [21] T. B. Alakus and I. Turkoglu, "Comparison of deep learning approaches to predict COVID-19 infection," *Chaos, Solitons and Fractals*, vol. 140, 2020.
- [22] H. Panwar, P. K. Gupta, M. K. Siddiqui, R. Morales-Menendez, P. Bhardwaj, and V. Singh, "A deep learning and grad-CAM based color visualization approach for fast detection of COVID-19 cases using chest X-ray and CT-Scan images," *Chaos, Solitons & Fractals*, vol. 140, no. August, Article ID 110190, 2020.
- [23] D. Ezzat, A. E. Hassanien, and H. A. Ella, "An optimized deep learning architecture for the diagnosis of COVID-19 disease based on gravitational search optimization," *Applied Soft Computing Journal*, vol. 98, Article ID 106742, 2020.
- [24] S. R. Nayak, D. R. Nayak, U. Sinha, V. Arora, and R. B. Pachori, "Application of deep learning techniques for detection of COVID-19 cases using chest X-ray images: a comprehensive study," *Biomedical Signal Processing and Control*, vol. 64, Article ID 102365, 2021.
- [25] A. Oluwasanmi, M. U. Aftab, Z. Qin et al., "Transfer learning and semisupervised adversarial detection and classification of COVID-19 in CT images," *Complexity*, vol. 2021, Article ID 6680455, 11 pages, 2021.
- [26] F. Dorr, H. Chaves, M. M. Serra et al., "COVID-19 pneumonia accurately detected on chest radiographs with artificial intelligence," *Intelligence-Based Medicine*, vol. 3-4, Article ID 100014, 2020.
- [27] R. H. Abiyev, M. Arslan, and J. C. Idoko, "Sign language translation using deep convolutional neural networks," *KSII Transactions on Internet and Information Systems*, vol. 14, no. 2, pp. 631–653, 2020.
- [28] S. H. Wang, D. R. Nayak, D. S. Guttery, X. Zhang, and Y. D. Zhang, "COVID-19 classification by CCSHNet with deep fusion using transfer learning and discriminant correlation analysis," *Information Fusion*, vol. 68, pp. 131–148, 2021.

- [29] Y. Oh, S. Park, and J. C. Ye, "Deep learning COVID-19 features on CXR using limited training data sets," *IEEE Transactions on Medical Imaging*, vol. 39, no. 8, pp. 2688–2700, 2020.
- [30] O. Russakovsky, J. Deng, H. Su et al., "ImageNet large scale visual recognition challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015.
- [31] D. S. Kermany, M. Goldbaum, W. Cai et al., "Identifying medical diagnoses and treatable diseases by image-based deep learning," *Cell*, vol. 172, no. 5, pp. 1122–1131, 2018.
- [32] R. H. Abiyev, M. Arslan, and J. B. Idoko, "Sign language translation using deep convolutional neural networks," *KSII Transactions on Internet and Information System*, vol. 14, no. 2, pp. 631–653, 2020.
- [33] D. P. Kingma and B. A. Jimmy, "A method for stochastic optimization," in *Proceedings of the International Conference on Learning Representations (ICLR 2015)*, San Diego, CA, USA, May 2015, <https://arxiv.org/abs/1412.6980>.
- [34] M. E. H. Chowdhury, T. Rahman, A. Khandakar et al., "Can AI help in screening viral and COVID-19 pneumonia?" *IEEE Access*, vol. 8, pp. 132665–132676, 2020.
- [35] A. U. Cavallo, J. Troisi, M. Forcina et al., "Texture analysis in the evaluation of Covid-19 pneumonia in chest X-ray images: a proof of concept study," *Current Medical Imaging*, vol. 17, no. 9, 2021.
- [36] T. C. Lin and H. C. Lee, "Covid-19 chest radiography images analysis based on integration of image preprocess, guided grad-CAM, machine learning and risk management," in *Proceedings of the ACM International Conference Proceeding Series*, pp. 281–288, Kamakura City, Japan, August 2020.
- [37] S. Rajpal, A. Rajpal, N. Lakhyani, and N. Kumar, "COV-ELM classifier: an extreme learning machine based identification of COVID-19 using chest X-Ray images," 2020, <https://arxiv.org/abs/2007.08637>.
- [38] A. Echtioui, W. Zouch, M. Ghorbel, C. Mhiri, and H. Hamam, "Detection methods of COVID-19," *SLAS Technology*, vol. 25, no. 6, pp. 566–572, 2020.
- [39] T. D. Pham, "Classification of COVID-19 chest X-rays with deep learning: new models or fine tuning," *Health Information Science and System*, vol. 9, 2021.
- [40] T. C. Mertýüz, H. C. Mertýüz, H. C. Taşar, and H. C. Yakut, "Covid-19 disease diagnosis from radiology data with deep learning algorithms," in *Proceedings of the 2020 4th International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT)*, pp. 1–14, Istanbul, Turkey, 2020.
- [41] A. Alotaibi, "Transfer learning for detecting Covid-19 cases using chest X-ray images," *International Journal of Machine Learning and Networked Collaborative Engineering*, vol. 4, no. 1, pp. 21–29, 2020.