

EMCNet: Automated COVID-19 diagnosis from X-ray images using convolutional neural network and ensemble of machine learning classifiers

Prottoy Saha ^{*}, Muhammad Sheikh Sadi, Md. Milon Islam

Department of Computer Science and Engineering, Khulna University of Engineering & Technology, Khulna, 9203, Bangladesh



ARTICLE INFO

Keywords:
COVID-19
Convolutional neural network
Ensemble of classifiers
Automatic diagnosis
X-ray images

ABSTRACT

Recently, coronavirus disease (COVID-19) has caused a serious effect on the healthcare system and the overall global economy. Doctors, researchers, and experts are focusing on alternative ways for the rapid detection of COVID-19, such as the development of automatic COVID-19 detection systems. In this paper, an automated detection scheme named EMCNet was proposed to identify COVID-19 patients by evaluating chest X-ray images. A convolutional neural network was developed focusing on the simplicity of the model to extract deep and high-level features from X-ray images of patients infected with COVID-19. With the extracted features, binary machine learning classifiers (random forest, support vector machine, decision tree, and AdaBoost) were developed for the detection of COVID-19. Finally, these classifiers' outputs were combined to develop an ensemble of classifiers, which ensures better results for the dataset of various sizes and resolutions. In comparison with other recent deep learning-based systems, EMCNet showed better performance with 98.91% accuracy, 100% precision, 97.82% recall, and 98.89% F1-score. The system could maintain its great importance on the automatic detection of COVID-19 through instant detection and low false negative rate.

1. Introduction

The recent coronavirus, caused by Severe Acute Respiratory Syndrome Coronavirus 2 (SARS-CoV-2) [1], was first identified in Wuhan, China, in January 2020 [2]. The World Health Organization (WHO) named it COVID-19 in February 2020 [3]. Initially, though it was not considered a severe case, the infection and rapid death rate of people have now collapsed the world's healthcare system. The WHO declared this pandemic as a Public Health Emergency of International Concern on January 30, 2020 [4]. A total of 25, 297, 383 coronavirus cases, 6,825, 774 active cases, and 848,561 deaths have been reported since August 30, 2020 around the world [5]. From developed countries (China, USA, Italy etc.) to underdeveloped countries, all are fighting against this severe pandemic in spite of a shortage of facilities, insufficient healthcare systems, and improper diagnostic approaches. The signs of being infected by COVID-19 are fever, cough, and severe respiratory problems [6].

One of the most commonly used tests for COVID-19 detection is real-time reverse transcription polymerase chain reaction (rRT-PCR) [7]. This test obtains DNA by reverse transcription and then uses PCR to amplify DNA for analysis. Thus, it can detect COVID-19 as this virus only

carries RNA [8]. However, this test has several limitations. It takes from a few hours to 2 days to detect COVID-19. Moreover, the test kit is not widely available. It is also not very reliable [9] because it can provide false negative results, which means it can misclassify COVID-19 patients as uninfected. Thus, the healthcare system and doctors are facing major problems in fighting against this pandemic. Given the contagious nature of the virus, it is spreading rapidly. Many countries have asked people to stay home and declared whole areas on "lockdown" to reduce the spreading rate and prevent this outbreak with a shortage of ventilators, ICUs, and vaccines [10].

However, RT-PCR and quarantine are not enough to prevent this outbreak, so researchers are trying to develop alternative solutions. They have focused on radiological image analysis for the diagnosis of COVID-19 [11]. CT scans or chest radiology image analysis can play an important role in this case. Researchers have discovered that COVID-19-infected chest X-ray or CT images have some distinct features such as ground glass opacity or vague darkened points, which can help in detecting COVID-19 [12]. Radiologists can analyze these images and help doctors in the early detection of COVID-19. However, in a pandemic, we need a more updated, available, and quick system. Manual analysis of the image may be a time-consuming process; in light

* Corresponding author.

E-mail addresses: prottoy@cse.kuet.ac.bd (P. Saha), sadi@cse.kuet.ac.bd (M.S. Sadi), milonislam@cse.kuet.ac.bd (Md.M. Islam).

of COVID-19's rapid spreading situation, the "time" parameter is critical. More developed, more advanced (i.e., automatic) COVID-19 detection systems are needed for diagnosis with the reduction of excessive pressure on radiologists' services.

In the healthcare system, deep learning has opened a new door [13]. With the contribution of deep neural network, the healthcare system has shown great advancement for automatic disease detection such as chest disease detection [14], cancer cell detection [15], tumor detection [16], and genomic sequence analysis [17]. Kieu et al. [18] proposed a deep learning-based model to detect the abnormal density of chest X-ray images. The system used three CNN models (CNN-128F, CNN-64L, and CNN-64R) to detect normal or abnormal density of chest X-ray images. The dataset included 400 chest X-ray images with 300 training images and 100 test images. The images were labeled 0 for normal and 1 for abnormal. Extensive experiments presented 96% accuracy for this system. Many researchers are currently working to build a deep neural network-based automated COVID-19 detection system [19]. They have developed several deep neural network models for feature extraction from COVID-19-infected chest X-ray images, thereby ensuring good accuracy, sensitivity, and specificity. Most of them have trained their model on a small dataset due to a lack of COVID-19 image sources. Therefore, these studies need further enhancement with proper training and testing models for real-time use.

In this study, a new system named EMCNet was proposed, which aims to detect COVID-19 by using chest X-ray images. The system has built a convolutional neural network (CNN) consisting of 20 layers. The model has a simple structure compared with pre-trained models. CNN extracts unique and distinct features from X-ray images. Subsequently, machine learning (ML) models are trained to develop a binary classifier (COVID-19 vs. Normal) with these feature vectors. Finally, all these models are combined to develop a voting ensemble of classifiers for better decision-making than any individual classifier.

The major contributions of the paper are as follows:

- i) The paper has proposed a simple CNN model that extracts deep features from chest X-ray images to detect patients with COVID-19.
- ii) For classification, an ensemble of classifiers is developed. Any one ML model can be selected for classification. However, the individual model showing excellent performance for one test set does not guarantee that it always provides better results for all other test sets. Different models can show varied results for varying datasets. Hence, four ML classifiers were combined to develop an ensemble of classifiers, which ensures better results for the dataset of various sizes and resolutions.
- iii) The models were trained and tested with a dataset of 1320 images where recent developing systems have conducted their research with comparatively small COVID-19 datasets.
- iv) The model showed excellent performance with 98.91% accuracy, 100% precision, 97.82% recall, and 98.89% F1-score.

The remaining paper is summarized as follows. Section 2 represents recent research for the detection of COVID-19 from CT scans or chest X-ray radiology images by using a deep learning network. Section 3 provides a full description of dataset, data preprocessing, proposed models of EMCNet, and their structures. Section 4 provides performance analysis of EMCNet with respect to evaluation metrics and comparative evaluation of EMCNet with existing systems as well. Finally, Section 5 concludes the paper.

2. Related work

Since the beginning of the COVID-19 outbreak, researchers have been trying their best to find automated COVID-19 detection systems using ML or deep neural networks. In this section, recent studies related to deep neural network-based COVID-19 detection and ensemble of ML

classifiers are described. Table 1 represents a comparative study of recent research.

Abbas et al. [20] proposed a new deep CNN model DeTraC based on chest X-ray images. The system simplified the local structure of the dataset by applying the class-decomposition layer. Subsequently, training of the system was performed using the pre-trained ResNet model. Finally, the class-composition layer was used to fine-tune the final classification. The dataset included 105 chest X-ray images of COVID-19, 11 images of SARS virus disease, and 80 normal chest X-ray images. Extensive experiments achieved 95.12% accuracy, 97.91% sensitivity, and 91.87% specificity for this system. Loey et al. [21] proposed a generative adversarial network with a deep neural network. The dataset consisted of 69 COVID-19 chest X-ray images, 79 pneumonia bacteria-infected X-ray images, 79 pneumonia virus-infected X-ray images, and 79 normal chest X-ray images. The research was conducted using pre-trained models. Among these models, Googlenet achieved 80.6% accuracy for the classification of four classes. Alexnet obtained 85.2% accuracy in three class scenarios, and Googlenet achieved 100% accuracy in two class scenarios. Rahimzadeh et al. [22] developed several deep neural networks for the automatic classification of COVID-19. The pre-trained models used for the system were Xception and ResNet50V2. The dataset consisted of 180 X-ray images of COVID-19 patients, 6054 images of pneumonia-affected patients and 8851 images of normal patients. Extensive experiments showed 91.4% average accuracy for all classes.

Oh et al. [23] proposed a patch-based CNN with a relatively small amount of trainable parameters. The system selected the final classification decision by majority voting from results at various path locations. The dataset consisted of 15,043 images including 180 images of COVID-19 cases, 6012 pneumonia images, and 8851 images of normal patients. The extensive experiment showed 88.9% accuracy, 83.4% precision, 85.9% recall, and 84.4% F1-score. Zhang et al. [24] proposed a new deep learning-based model (anomaly detection model) based on chest X-ray images. The system consisted of three modules: a backbone network to extract high-level features of chest X-ray images, a classification head to generate a classification score, and an anomaly detection head to generate a scalar anomaly score. The dataset included 100 chest X-ray images for COVID-19 and 1431 images for pneumonia cases. Extensive experiments presented 96% sensitivity and 70.65% specificity for this system. Apostolopoulos et al. [25] introduced several pre-trained CNN models for COVID-19 detection. The system used transfer learning for diagnosis with a relatively small COVID-19 dataset. The experiment was conducted with two datasets. The first dataset contained 1427 X-ray images including 224 COVID-19 infected cases, 700 images of bacterial pneumonia cases, and 504 normal cases. The second dataset contained 224 COVID-19 X-ray images, 700 bacterial and viral pneumonia images, and 504 normal cases. Among the pre-trained models, MobileNetV2 achieved 96.78% accuracy, 98.66% sensitivity, and 96.46% specificity for the second dataset.

Mahmud et al. [26] proposed a new deep learning model named CovXNet using chest X-ray images for automatic detection of COVID-19 and other pneumonia cases. The system introduced depthwise convolution to extract high-level features from chest X-ray images. First, they trained their model with normal and (viral/bacterial) pneumonia chest X-ray images. Learning of this step was transferred with additional layers, and training was performed with COVID-19 chest X-ray images and pneumonia cases. The system used a stacking algorithm and gradient-based discriminative localization. The dataset consisted of 305 images of COVID-19 cases, 1493 viral pneumonia images, 2780 bacterial pneumonia images, and 1583 images of normal patients. Extensive experiments presented 97.4% accuracy for COVID-19 versus normal cases and 90.2% accuracy for classification among COVID-19, normal, and viral and bacterial pneumonia cases. Tsiknakis et al. [27] developed a deep learning-based model to diagnose COVID-19 from chest X-ray images. The pre-trained model Inception-V3 was introduced, and transfer learning was combined due to the limitation of the COVID-19

Table 1

Comparative study of recent researches related to COVID-19 detection and ensemble of ML classifiers.

Author	Sources of Dataset	Dataset Details	Model	Ensemble	Accuracy (%)
Abbas et al. [20]	COVID-19 and SARS images: [31]; Normal images [32,33]:	196 (COVID-19 = 105, Normal = 80, SARS = 11)	ResNet	No	95.12%
Loey et al. [21]	COVID-19 images: [31]; Normal and Pneumonia images [34]:	306 (COVID-19 = 69, Normal = 79, Bacteria pneumonia = 79, Virus pneumonia = 79)	Googlenet	No	80.56
Rahimzadeh et al. [22]	COVID-19 images: [31]; Normal and Pneumonia images [35]:	15,085 (COVID-19 = 180, Normal = 8851, Pneumonia = 6054)	Xception + ResNet50V2	No	91.4
Oh et al. [23]	COVID-19 images: [31]; Normal and Pneumonia images [36]:	15,043 (COVID-19 = 180, Normal = 8851, pneumonia = 6012)	Patch based CNN	No	88.9
Zhang et al. [24]	COVID-19 and images: [31]; Normal images [37]:	1531 (COVID-19 = 100, Normal = 1431)	18 layer residual CNN	No	72.31%
Apostolopoulos et al. [25]	COVID-19 images: [31,38]; Normal and Pneumonia images [34]:	1442 (COVID-19 = 224, Normal = 504, Pneumonia = 714)	MobileNet V2	No	96.78
Mahmud et al. [26]	COVID-19 images: Sylhet Medical College, BD; Normal and Pneumonia images [39]:	6161 (COVID-19 = 305, Normal = 1583, Bacteria pneumonia = 2780, Virus pneumonia = 1493)	CNN with Depthwise dilated convolution	No	90.2
Tsiknakis et al. [27]	COVID-19 images: [31]; Normal and Pneumonia images [34,35]:	572 (COVID-19 = 122, Normal = 150, Bacteria pneumonia = 150, Virus pneumonia = 150)	Inception V3	No	76
Sethy et al. [28]	COVID-19 images: [31,38]; Normal and Pneumonia images [34]:	381 (COVID-19 = 127, Normal = 127, Bacteria pneumonia = 63, Virus pneumonia = 64)	Resnet50 + SVM	No	95.33
Horry et al. [29]	COVID-19 images: [31]; Normal and Pneumonia images [37]:	60,838 (COVID-19 = 115, Normal = 60,361, pneumonia = 322)	VGG 19	No	81
Hasan et al. [30]	[40]	768 (Diabetic = 268, non-Diabetic = 500)	KNN, DT, RF, Naive Bayes, AB, XB	Yes	72.26

dataset. The dataset included 122 images for COVID-19, 150 for bacterial pneumonia, 150 for viral pneumonia, and 150 for normal cases. The system demonstrated 100% accuracy, 99% sensitivity, and 100% specificity for COVID-19 versus common pneumonia binary classification and 76% accuracy, 93% sensitivity, and 87% specificity for four classes' classification. Sethy et al. [28] proposed a system where nine different pre-trained models extracted features from chest X-ray images. The system used SVM to classify COVID-19 using extracted features. A total of 381 chest X-ray images were used as a dataset for the experiment. Among all the models, ResNet50 was considered best for feature extraction. The system obtained an accuracy of 95.33% and F1-score of 95.34% using ResNet50 and SVM. Horry et al. [29] introduced a scheme where four different pre-trained models were used. The dataset contained 115 samples of COVID-19, 322 pneumonia cases, and 60,361 healthy case samples. Among the models, VGG16 and VGG19 performed best. VGG19 achieved 81% accuracy for COVID-19 versus pneumonia classification. Hasan et al. [30] developed a new ensemble model based on ML classifiers for diabetic prediction. The system proposed six ML models: k-nearest neighbor, decision trees, random forest, AdaBoost, Naive Bayes, and XGBoost. The system used weighted ensembling. Weights were calculated from the area under ROC curve of the ML models. The system conducted this experiment with 268 diabetic patients and 500 non-diabetic patients. Extensive experiment achieved 78.9% sensitivity and 93.4% specificity for this system.

3. Methodology

To develop an automatic COVID-19 detection system named EMC-Net, data were collected from multiple resources. The collected images were then resized. Data normalization was performed on the dataset to prevent overfitting and facilitate generalization. The dataset was partitioned into three different sets: training set, validation set, and testing set. With the training and validation sets, the proposed CNN model was trained. The experiment was run up to different epochs such as 50, 60, and 100. After running the network for 50 epochs, the network accuracy started saturating. With 50 epochs, the model achieved the expected training accuracy and validation accuracy. The first fully connected layer (FCL) of the model was selected to extract features. The feature vector was extracted from each training image with this layer. The feature vectors of all these images were fed into four ML classifiers. To tune the best hyperparameters of ML classifiers, the grid search technique was applied on each ML classifier. Finally, all ML classifiers were

combined to develop an ensemble of classifier, which predicts class labels based on the majority vote of ML classifiers. The performance of the proposed system was evaluated in terms of confusion matrix, precision, recall accuracy, and F1-score. The overall system architecture of EMC-Net is highlighted in Fig. 1.

3.1. Description of datasets

In the proposed EMCNet, datasets were collected from multiple sources. First, 660 positive COVID-19 chest X-ray images were collected from Github repository developed by Cohen et al. [31]. This repository contains X-ray images of positive COVID-19, negative COVID-19, and pneumonia cases. The images of this repository have been gathered from open sources and hospitals. Though the complete metadata of all patients are not described in the source, the average age of the infected patients was around 55 years. At the time of carrying out the experiment, the Cohen database contained 500 chest X-ray images of positive COVID-19-infected patients. Negative COVID-19 images included other viral and bacterial pneumonia (MERS, SARS, and ARDS). The proposed system considered only COVID-19 disease and healthy cases. The proposed system did not consider other viral and bacterial pneumonia diseases. Thus, negative images were not used from this repository. Only 500 positive images were selected from here for COVID-19 cases. A total of 1800 COVID-19 chest X-ray images were collected from another Github source [41], SIRM database [42], TCIA [43], radiopaedia.org [44], and Mendeley [45]. A total of 2300 COVID-19 X-ray images were collected from these sources. Normal chest X-ray images were obtained from the Kaggle repository [46] and NIH chest X-ray images [47]. From this source, 2300 images of normal chest X-ray images were collected. The samples are shown in Fig. 2. Thus, the dataset of the proposed EMCNet contained a total of 4600 images. The dataset was partitioned into 70%:20%:10% for three different sets: training set with 3220 images, validation set with 920 images, and test set with 460 images. The partition of the dataset into training, validation, and testing sets is described in Table 2.

3.2. Data preprocessing

All sample images were of various sizes. Therefore, the images were resized to 224 × 224 pixels. Data normalization was performed for better learning of the system. Thus, the dataset became ready to be fed into the CNN network and to train the model.

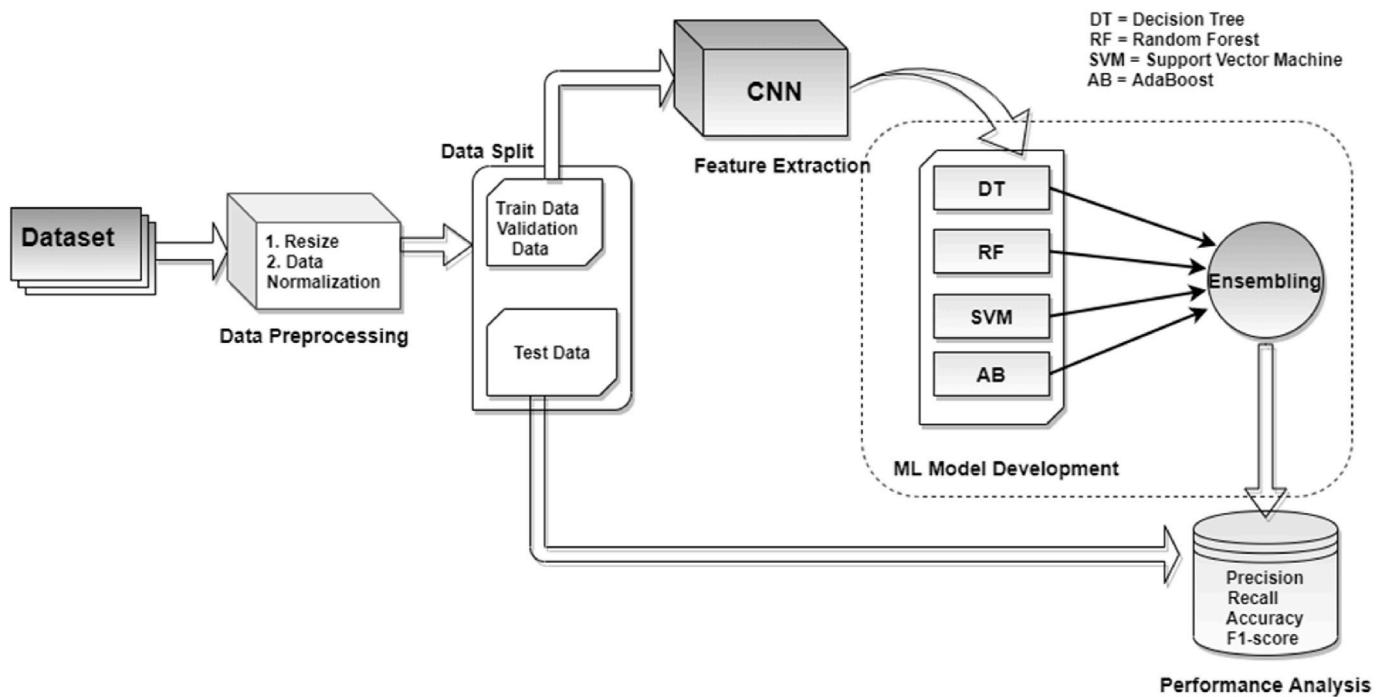


Fig. 1. Architecture of EMCNet.

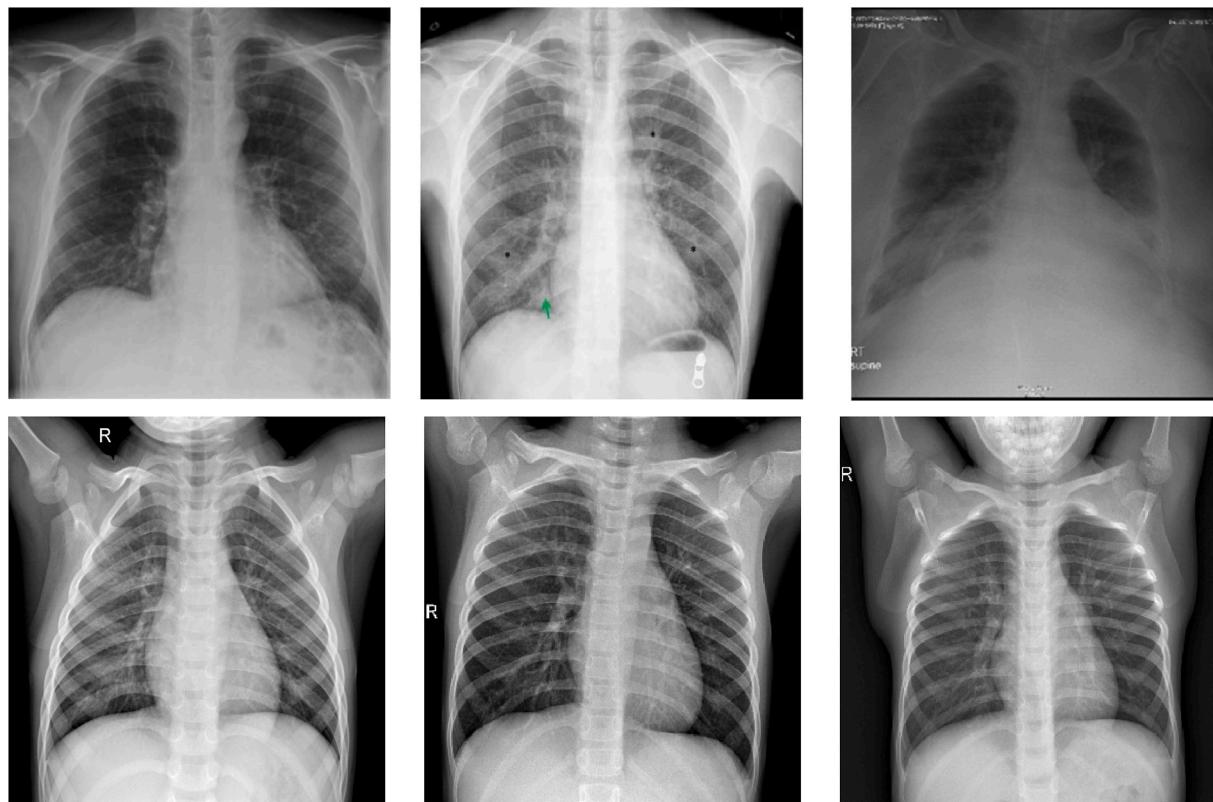


Fig. 2. First three images of the first row are samples of COVID-19 X-ray images. The rest of the images are of normal chest X-ray images.

3.3. Development of the proposed EMCNet

There are several deep transfer learning-based architectures for feature extraction from images such as AlexNet, VGG 16, Inception, and ResNet-50. ResNet-50 is a 50 layer residual network with the identity connection between the layers. ResNet-50's architecture has four stages.

The first layer is convolutional layer with filter of size 7×7 and stride of 2. The second layer is max-pooling layer with filter of size 3×3 and stride of 2. Stage 1 of the network then begins. It consists of three residual blocks each containing three layers. In all three layers of block 1, 64, 64 and 256 kernels are used. In the residual block, convolution is performed with stride 2. Thus, the height and width of an image are

Table 2

Partition of the dataset into training, validation and testing set.

Dataset	COVID-19	Normal	Total
Training	1610	1610	3220
Validation	460	460	920
Testing	230	230	460
Total	2300	2300	4600

reduced to half from one stage to another, but the channel width is doubled. Another pre-trained model VGG-16 has six stages. In the first two stages, there are two convolutional layers with one max-pooling layer of stride 2. In the next three stages, there are three convolutional layers with one max-pooling layer of stride 2. The last stage has three FCLs. The convolutional layers use a filter of size 3×3 and stride of 1. The number of filters, starting from 64, doubles in each stage except for stage 5.

Transfer learning is the method of taking the weights of a pre-trained model and using the previously learned features to make a decision of a new class label. A network is used in transfer learning that is pre-trained in the imangenet dataset, and this network learnt to identify high-level features of images in the initial layers. For transfer learning, a few dense layers were added at the end of the pre-trained network. The network then learns what combinations of features will help identify the features in new data collection. In EMCNet, the CNN model was inspired from the pre-trained model VGG-16 by giving priority to the simplicity of the model. In the proposed CNN, a maximum of two convolutional layers were considered in each stage, and the dropout layer was added to prevent overfitting of the model.

To develop EMCNet, the CNN model was used to extract the features for each training image. The feature set was then passed to ML classifiers to train them. Finally, all these models were combined to develop an ensemble of classifiers. A detailed explanation is shown as follows.

3.3.1. Proposed CNN architecture

The proposed EMCNet has developed a simple CNN network to extract features. In general, CNN has three different layers: convolutional layer, pooling layer, and FCL. The layer types of the proposed model and their explanation are shown in Table 3. To train the model, RGB images (size $224 \times 224 \times 3$) are fed into the CNN model. Here, the number of channels (n_c) in input image is 3. The first layer is the convolutional layer. This layer takes filters, which are also called "kernel" of size $(f_h \times f_w)$, where $f_h (= 3)$ is the filter height, and $f_w (= 3)$ is the filter width. Normally, the filter height (f_h) and width (f_w) remain the same.

This filter can be called "feature identifier." With these filters, the layer obtains low-level features such as edge and curve. The more convolutional layers are added, the better the model is able to extract deep features from images; thus, the model can determine the complete characteristics of images. More convolutional layers have been added in the model step by step. The filter performs convolution operation with a sub area of the image. Convolution operation means element-wise multiplication and summation with the filter and pixel values of the image. The values of filter are called weights or parameters. These weights have to be learnt by training the model. The sub-area of the image is named "receptive field." The filter starts convolution from the beginning of the image. It then is continuously shifted across the whole image by a certain amount of unit and performs convolution until the whole image is covered. One operation outputs one single value. Convolution throughout the image outputs an array or metric of values. The operation is expressed as shown in (1).

$$C = I * F = \sum \sum I(i+m, j+n)F(f_h, f_w) \quad (1)$$

where I is the input, and F is the filter of size $(f_h \times f_w)$. The operation is represented by the operator (*).

The amount by which the filter is shifted is determined by another parameter called stride. For the model, stride is set to 1 for all convolutional layers. The more stride increases, the more spatial dimension (height and width) of input volume decreases. High stride value for the minimum overlap of the receptive field can create some problems such as the receptive field can go beyond the input volume and the dimension can be reduced. To overcome these problems, padding is used. "Zero padding" (also called "same" padding) pads the input with zero around the border and keeps the output volume dimension the same as the input dimension. If stride size is 1, then the size of zero padding is determined by (2).

$$\text{ZeroPadding} = \frac{f - 1}{2} \quad (2)$$

where f is the filter height or width; here, both height and width are equal in size. For the proposed EMCNet, "valid padding" was used instead of zero padding, so the output dimension was not the same as the input dimension, and it shrank after convolution.

Multiple filters in the convolutional layer have been used for extracting multiple features. In the first layer, 32 filters were used. The number of filters increased gradually in the next layers, from 32 to 128, 128 to 512, and so on. The output volume is called "activation map" or "feature map." The output shape of all layers is shown in Table 4. The

Table 3

CNN layers and their detail explanation.

Layer	Filter Size	Pool Size	Stride	Padding	Number of Filters	Dropout Threshold	Activation
Conv2D	3×3	–	1	Valid	32	–	Relu
Conv2D	3×3	–	1	Valid	128	–	Relu
MaxPooling2D	–	2×2	2	–	–	–	–
Dropout	–	–	–	–	–	.25	–
Conv2D	3×3	–	1	Valid	64	–	Relu
MaxPooling2D	–	2×2	2	–	–	–	–
Dropout	–	–	–	–	–	.25	–
Conv2D	3×3	–	1	Valid	128	–	Relu
MaxPooling2D	–	2×2	2	–	–	–	–
Dropout	–	–	–	–	–	.25	–
Conv2D	3×3	–	1	Valid	512	–	Relu
MaxPooling2D	–	2×2	2	–	–	–	–
Dropout	–	–	–	–	–	.25	–
Conv2D	3×3	–	1	Valid	512	–	Relu
MaxPooling2D	–	2×2	2	–	–	–	–
Dropout	–	–	–	–	–	.25	–
Flatten	–	–	–	–	–	–	–
FCL	–	–	–	–	64	–	Relu
Dropout	–	–	–	–	–	.25	–
FCL	–	–	–	–	2	–	Sigmoid

Table 4
Model summary of the proposed CNN model.

Number of Layers	Layer (Type)	Output Shape	Parameter
1	conv2d_6 (Conv2D)	[222, 222, 32]	896
2	conv2d_7 Conv2D)	[220, 220, 128]	36,992
3	max_pooling2d_5 (MaxPooling2D)	[110, 110, 128]	0
4	dropout_6 (Dropout)	[110, 110, 128]	0
5	conv2d_8 (Conv2D)	[108, 108, 64]	73,792
6	max_pooling2d_6 (MaxPooling2D)	[54, 54, 64]	0
7	dropout_7 (Dropout)	[54, 54, 64]	0
8	conv2d_9 (Conv2D)	[52, 52, 128]	73,856
9	max_pooling2d_7 (MaxPooling2D)	[26, 26, 128]	0
10	dropout_8 (Dropout)	[26, 26, 128]	0
11	conv2d_10 (Conv2D)	[24, 24, 512]	590,336
12	max_pooling2d_8 (MaxPooling2D)	[12, 12, 512]	0
13	dropout_9 (Dropout)	[12, 12, 512]	0
14	conv2d_11 (Conv2D)	[10, 10, 512]	2,359,808
15	max_pooling2d_9 (MaxPooling2D)	[5, 5, 512]	0
16	dropout_10 (Dropout)	[5, 5, 512]	0
17	flatten_1 (Flatten)	[12,800]	0
18	dense_2 (Dense)	[64]	819,264
19	dropout_11 (Dropout)	[64]	0
20	dense_3 (Dense)	[2]	65

size of output volume is determined by (3), (4), and (5).

$$O_h = \frac{I_h - f_h + 2P}{S} + 1 \quad (3)$$

$$O_w = \frac{I_w - f_w + 2P}{S} + 1 \quad (4)$$

$$O_n = N_f \quad (5)$$

where I_h is the input height, I_w is the input width, f_h is the filter height, f_w is the filter width, S is the stride size, P is the padding and N_f is the number of filters. For our 1st convolutional layer.

$$I_h = 224, I_w = 224, f_h = 3, f_w = 3, S = 1, P = 0 \text{ and } N_f = 32.$$

From (3), (4), and (5), the following values can be obtained.

$$O_h = \frac{224 - 3 + 0}{1} + 1 = 222$$

$$O_w = \frac{224 - 3 + 0}{1} + 1 = 222$$

$$O_n = 32$$

Non-linear activation is applied on convolution output. Convolutional layer has performed linear computation (element-wise multiplication and summation). Thus, nonlinearity is introduced on the linear operation with this activation. Activation ReLU (rectified linear unit) is applied on convolution output. The function for ReLU operation is shown in (6).

$$\text{ReLU}(X) = \max(0, X) \quad (6)$$

Here, X is the convolution operation output. ReLU mitigates all negative output by zero. In the proposed model, the motive to use ReLU is that it enhances the nonlinearity of the model and helps make the computational time faster without affecting model accuracy. It also reduces the vanishing gradient problem where lower layers are trained very slowly.

After two convolution layers, the maxpooling layer is used. This layer reduces spatial dimension (height and width) of input. In the proposed

model, the layer takes a filter of size 2×2 , and its stride is equal to 2. The filter convolves around the input volume and outputs the maximum value of the receptive field. The observation that works for using this layer is that a specific feature's relative location with respect to other features is more important than its exact location. It prevents overfitting and reduces the number of weights, which result in reduction of computational cost. The dropout layer is then used. This layer drops out some activation randomly by setting them to zero. This layer ensures that the model can predict actual class label of an image, despite some activations being dropped out. Thus, the model should not be too fitted to train the dataset. The dropout layer helps prevent overfitting. Dropout layers have been applied with a threshold of 0.25. The flatten layer converts a 2D feature map into a 1D feature vector, which is fed to an FCL. To date, deep features of images have been extracted. With FCL, the classification task of COVID-19 is performed from a 1D long feature vector. In the proposed CNN, FCL consists of 64 neurons. The first FCL passes output activation to the second FCL. Finally, in the output layer, the model predicts the class label between COVID-19 versus normal cases through "softmax" activation. The proposed CNN architecture is illustrated in Fig. 3.

3.3.2. ML and ensembling techniques

Features were picked up from first FCL of CNN. Form Table 4, the first FCL was "dense_2" with 64 neurons. Thus, the feature vector was of dimension (1×64) . Deep features were extracted from each training image, so there were 64 features for each image. The training set contained 3220 images, and new input set dimension to train our ML classifiers was (3220×64) . The input data were passed to the ML models to train them. The grid search algorithm was used to tune hyperparameters of ML classifiers. Fivefold cross-validation was applied on the models to learn best classifiers. The ML classifiers were combined to develop an ensemble of classifiers. The ensemble of classifiers combined the results from four different ML classifier models and used the final decision of the class label based on majority voting. It helped improve performance and showed ideally better performance than any single classifier. For ensembling, "hard voting" technique was applied where the classifier used decisions based on the largest sum of votes from other models. The training phase of the ML classifiers is shown in Fig. 4.

3.4. Metrics for performance evaluation

For the performance evaluation of EMCNet, the metrics used in this paper were accuracy, precision, recall, and F1-score. The formulas for deriving the values of these metrics are shown in (7), (8), (9), and (10).

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

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

$$\text{Recall} = \frac{TP}{TP + FN} \quad (9)$$

$$\text{F1-score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (10)$$

In the above equations, TP means "true positive," which represents the successful prediction of actual class label of COVID-19 cases by the system. TN means "true negative," which describes the successful prediction of the actual class label of normal cases by the system. FP means "false positive," which indicates that the model misclassifies normal cases as COVID-19, but they are actually not COVID-19-infected patients. FN represents "false negative," which highlights that model misclassifies COVID-19 cases as normal.

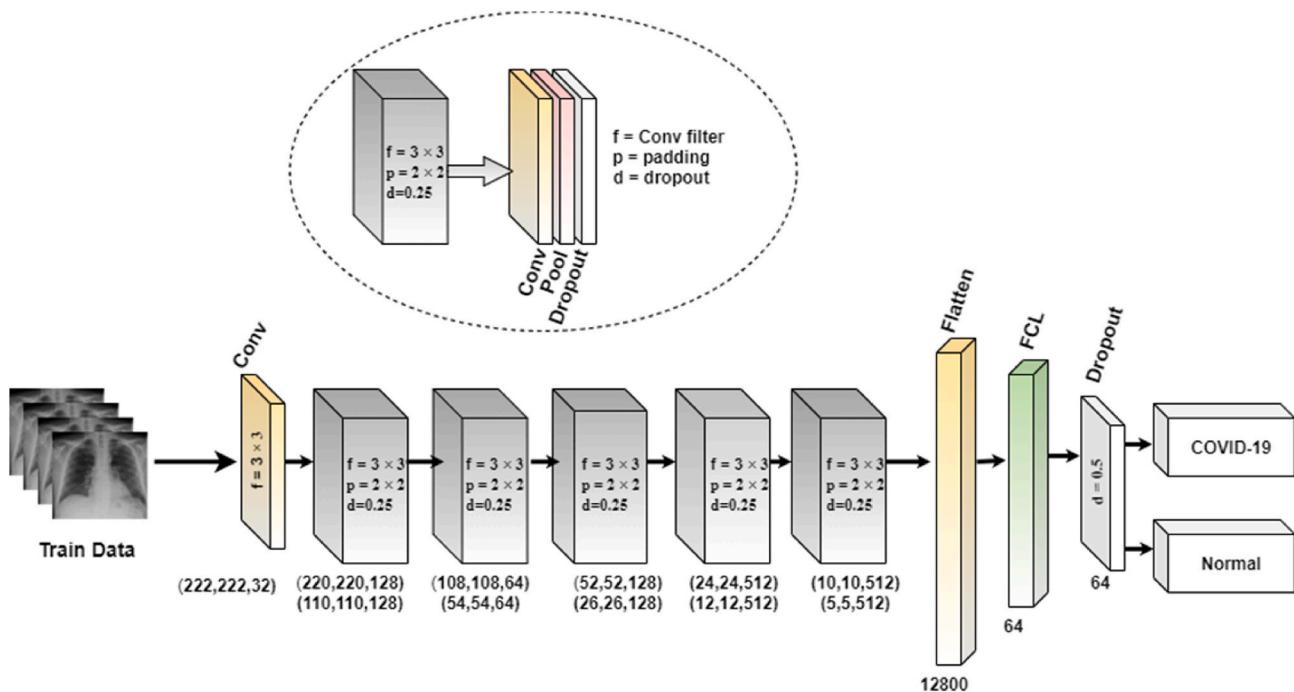


Fig. 3. Proposed CNN architecture.

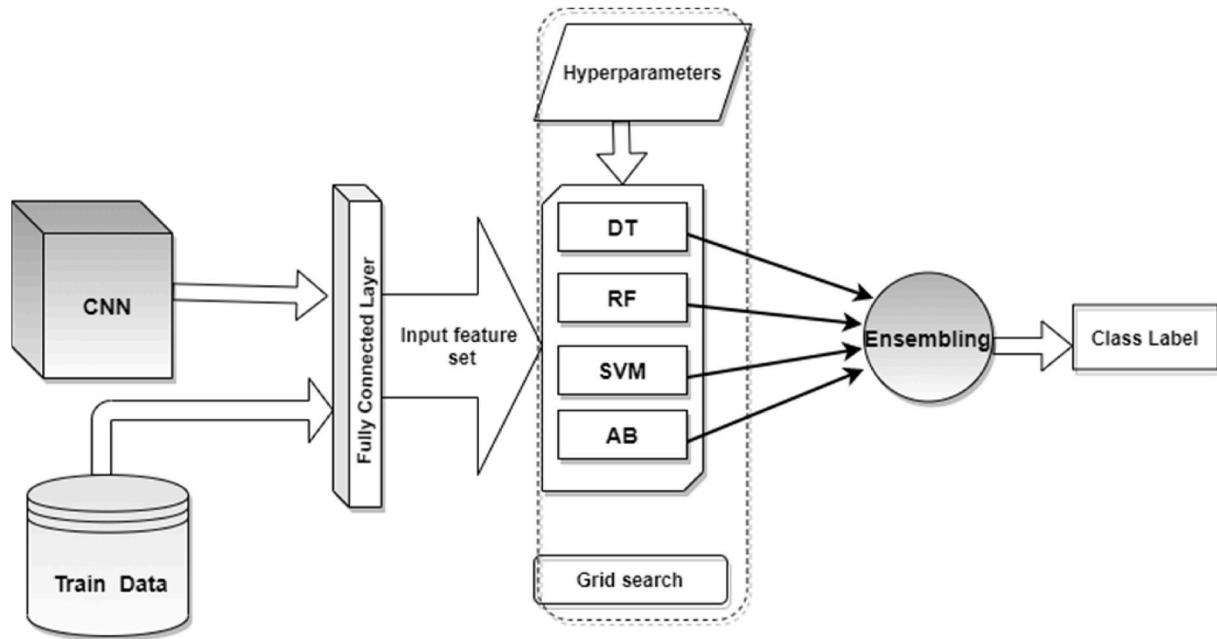


Fig. 4. Training process of ML classifiers and ensemble of classifiers.

4. Experimental analysis

EMCNet extracts features from images with CNN and classifies COVID-19 with an ensemble of four different ML classifiers. The CNN model was trained up to 50 epochs. Hyperparameters of the ML classifiers were tuned with the grid search technique. Performance of the proposed CNN, ML classifiers and ensemble of classifiers for each class label was analyzed in terms of confusion matrix, ROC curve, precision, recall, accuracy, and F1-score.

4.1. Experimental setup

The experiment was run on Google Collaboratory. Google Collaboratory is a Jupyter notebook-based cloud service for disseminating knowledge and work on machine learning. It offers a completely optimized runtime for deep learning and free-of-charge access to a stable GPU.

4.2. Results analysis

Fig. 5 shows the training accuracy and validation accuracy with respect to epochs. The CNN model was run up to 50 epochs. The

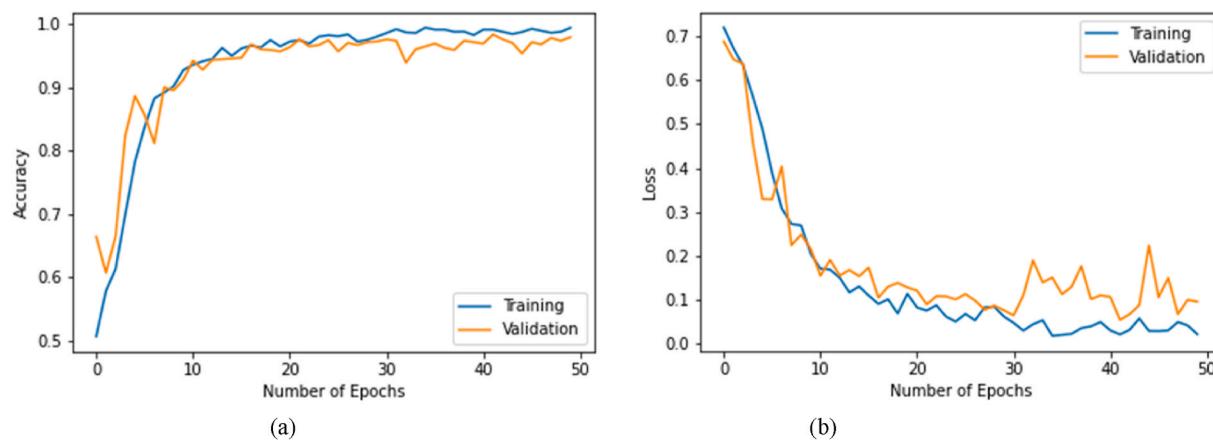


Fig. 5. Performance analysis of the CNN used in EMCNet. (a) Training and Validation Accuracy (b) Training and Validation Loss.

maximum obtained accuracy for training was 99.97%, and that for validation was 98.33%. These values indicated that our model learnt well and could correctly classify COVID-19 versus normal cases. The training loss was 0.0021, and the validation loss was 0.0958.

With extracted features, four ML classifiers were trained. The grid search technique was applied to tune hyperparameters. The tuned hyperparameters' values of the models are shown in Table 5. To evaluate the model, several performance metrics were considered for the efficient diagnosis of COVID-19. Fig. 6 describes the confusion matrices for EMCNet with six classifiers from CNN to an ensemble of classifiers. In the test set, there were 230 COVID-19 images and 230 normal X-ray images. In the confusion matrices, actual cases were placed along rows, and predicted cases were placed along columns. For CNN, among 230 COVID-19 cases, the model detected 222 cases and misclassified 8 cases as normal. The model predicted the exact class label of all normal cases. For RF (where extracted features were used to learn the model), the model detected 221 cases and misclassified 9 cases as normal among 230 COVID-19 cases. DT predicted the exact class label of 220 COVID-19 cases and misclassified 10 cases as normal. AB predicted the exact class label of 222 COVID-19 cases and misclassified 8 cases as normal. All these three models predicted the exact class label of all normal cases. For SVM, among 230 COVID-19 cases, the model detected 223 cases and misclassified 7 cases as normal. Among normal cases, the model detected 221 cases and misclassified 9 cases as COVID-19 positive. Finally, our ensemble of classifiers, which used the decision based on majority voting of four ML classifiers, classified 225 COVID-19 cases and 230 normal cases.

The evaluation matrices precision, recall, accuracy, and F1-score of

classifiers for each class (COVID-19 vs. Normal) are shown in Table 6, and their graphical view is presented in Fig. 7. The CNN model achieved 100% precision, 96.52% recall, 96.52% accuracy, and 98.23% F1-score for COVID-19 classes and 96.64% precision, 100% recall, 100% accuracy, and 98.29% F1-score for normal classes. The DT model achieved 100% precision, 95.65% recall, 95.65% accuracy, and 97.78% F1-score with COVID-19 cases and 95.83% precision, 100% recall, 100% accuracy, and 97.87% F1-score with normal cases. The RF model achieved 100% precision, 96.09% recall, 96.09% accuracy, and 98% F1-score with COVID-19 cases and 96.23% precision, 100% recall, 100% accuracy, and 98.08% F1-score with normal cases. The SVM model achieved 96.12% precision, 96.96% recall, 96.96% accuracy, and 96.54% F1-score with COVID-19 cases and 96.93% precision, 96.09% recall, 96.09% accuracy, and 96.51% F1-score with normal cases. The AB model achieved 100% precision, 96.52% recall, 96.52% accuracy, and 98.23% F1-score with COVID-19 cases and 96.64% precision, 100% recall, 100% accuracy, and 98.29% F1-score with normal cases. Finally, the proposed ensemble of classifiers returned 100% precision, 97.83% recall, 97.83% accuracy, and 98.90% F1-score with COVID-19 cases and 97.87% precision, 100% recall, 100% accuracy, and 98.92% F1-score with normal cases.

An ROC curve is plotted with true positive rate (TPR) along the x-axis, and false positive rate (FPR) is plotted along the y-axis. The formulas for obtaining TPR and FPR are shown in (11) and (12).

$$TPR = \frac{TP}{TP + FN} \quad (11)$$

$$FPR = \frac{FP}{TN + FP} \quad (12)$$

The higher the area under the curve (AUC) of ROC is, the greater the model is considered to be efficient for medical diagnosis. The ROC curves of our classifiers are plotted in Fig. 8. For CNN, the AUC was 0.9832. For RF, the AUC was 0.9812. For DT, the AUC was 0.9792. For SVM, the AUC was 0.9653. For AB, the AUC was 0.9832. Finally, the ensemble of classifiers achieved an AUC of 0.9894. Thus, the model could contribute efficiently in detecting COVID-19 cases from chest X-ray images.

The overall system's precision, recall, accuracy, and F1-score of classifiers are shown in Table 7. As shown in this table, CNN achieved 100% precision, 96.52% recall, 98.26% accuracy, and 98.22% F1-score for automatic diagnosis of COVID-19. For the RF model, precision was 100%, recall was 96.09%, accuracy was 98.04%, and F1-score was 98%. For the DT model, precision was 100%, recall was 95.65%, accuracy was 97.82%, and F1-score was 97.77%. For the SVM model, precision was 96.12%, recall was 96.96%, accuracy was 96.52%, and F1-score was 96.54%. For the AB model, precision was 100%, recall was 96.52%,

Table 5
Tuned hyperparameters of ML classifiers.

ML classifiers	Hyperparameters
Random Forest	Bootstrap: True (method for sampling - with or without replacement) max depth: 100 (max number of levels in decision tree) max features: 2 (maximum features for splitting) min samples leaf: 4 (minimum data allowed in a leaf node) min samples split: 10 (minimum data allowed for split) n estimators: 100 (number of trees)
Support Vector Machine	C: 1 (Regularization parameter) Kernel: rbf (Specifies the kernel type- 'linear', 'poly', 'rbf', 'sigmoid') Gamma: 0.1 (Kernel coefficient for 'rbf', 'poly' and 'sigmoid'.)
Decision Tree	max leaf nodes: 10 min samples split: 4 Criterion: entropy max depth: 6
AdaBoost	n_estimators: 250

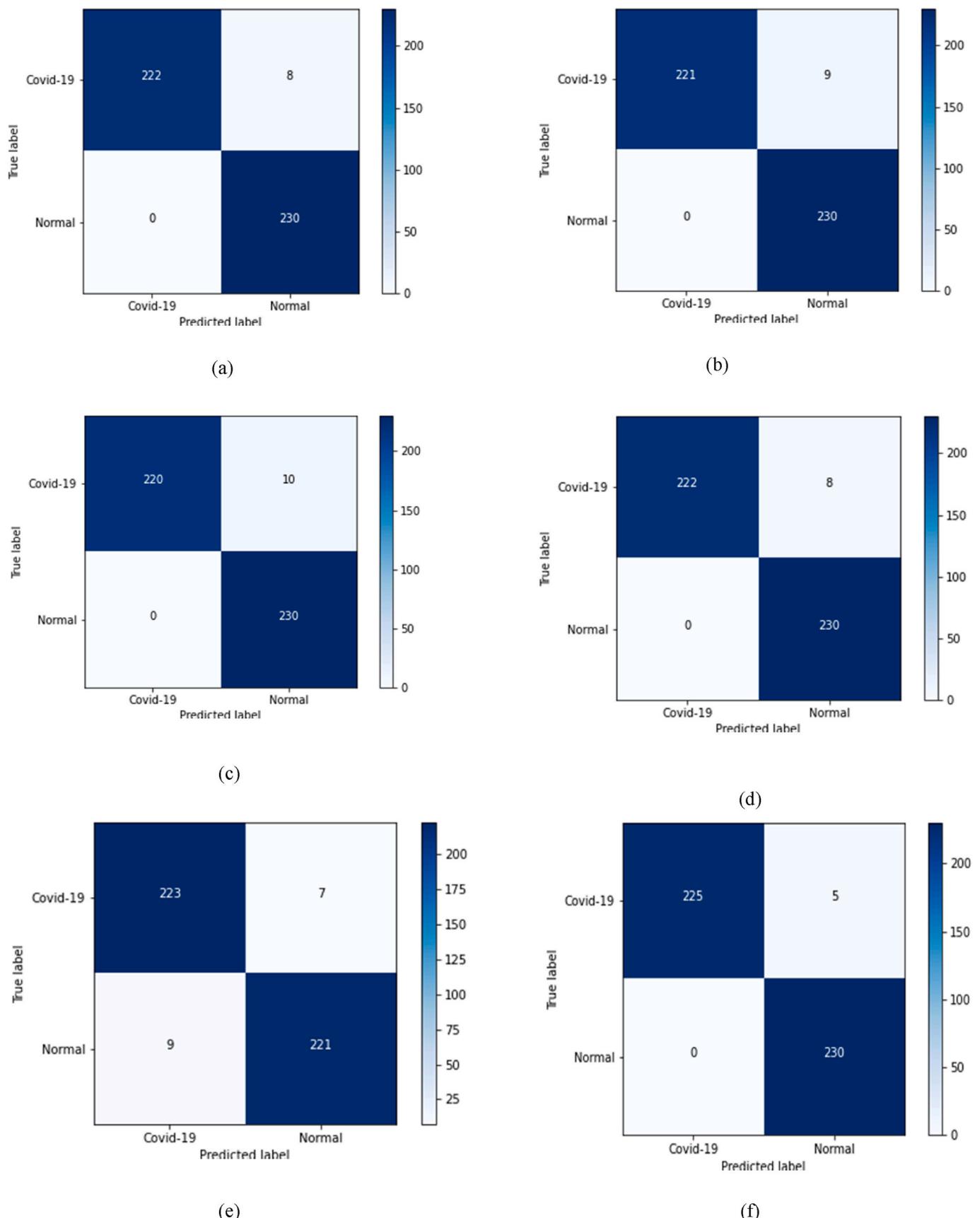


Fig. 6. Confusion matrix representation for the classifiers used in EMCNet (a) CNN (b) RF (c) DT (d) AB (e) SVM (f) Ensemble of Classifiers.

Table 6

Performance evaluation of the classifiers of EMCNet based on each class label.

Models	Class	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
CNN	COVID	96.52	100	96.52	98.23
	Normal	100	96.64	100	98.29
DT	COVID	95.65	100	95.65	97.78
	Normal	100	95.83	100	97.87
RF	COVID	96.09	100	96.09	98.00
	Normal	100	96.23	100	98.08
SVM	COVID	96.96	96.12	96.96	96.54
	Normal	96.09	96.93	96.09	96.51
AB	COVID	96.52	100	96.52	98.23
	Normal	100	96.64	100	98.29
Ensembling	COVID	97.83	100	97.83	98.90
	Normal	100	97.87	100	98.92

accuracy was 98.26%, and F1-score was 98.22%. Finally, our ensemble of classifiers achieved 100% precision, 97.82% recall, 98.91% accuracy, and 98.89% F1-score. Among the four ML models, recall was greater for SVM and AB. These two models played a key role in decision making of COVID-19 cases for ensembling. However, the rest of the ML models (except SVM) correctly predicted all normal cases. Thus, these three models contributed to correctly classify all normal cases of COVID-19.

4.3. Comparison of the EMCNet with the state of the art

The comparative performance evaluation of EMCNet with other recent research for automatic detection of COVID-19 is shown in Table 8. The systems proposed in Ref. [21,24,27], and [23] obtained accuracy of 72.31%, 76%, 80.56%, and 88.9%, respectively. The systems shown in Ref. [20,22], and [28] have improved the accuracy level to 91.4%, 95.12%, and 95.33%, respectively. The system proposed in Ref. [25] shows that their system can detect COVID-19 with 96.78% accuracy. The accuracy achieved by EMCNet outperformed all these models. EMCNet achieved 98.91% accuracy, which demonstrated that it might be an effective tool for the automatic detection of COVID-19 from chest X-ray images.

5. Conclusions

The world is currently suffering tremendously because of the COVID-19 pandemic. Many people have already died due to the lack of proper treatment, insufficient facilities, or absence of early detection. EMCNet can contribute to the infected patients with automatic detection of

COVID-19 from chest X-ray images. With CNN, EMCNet extracts high-level features from X-ray images. The ensemble model classifies COVID-19 versus normal cases with high accuracy. The dataset contains a large amount of COVID-19 images compared with other recent research. An extensive experiment shows greater performance with 98.91% accuracy, 100% precision, 97.82% recall, and 98.89% F1-score. Considering all these experimental results, EMCNet can play an important role as a helping hand of doctors and may be used as an alternate of manual radiology analysis for the automatic detection of COVID-19 in this recent pandemic. Although EMCNet has some limitations (e.g., it can misclassify some COVID-19-positive cases as negative), it can function as an alternate of manual radiology analysis and aid doctors in the automatic detection of COVID-19 from chest X-ray images. EMCNet is highly capable of relieving pressure on frontline doctors and nurses;

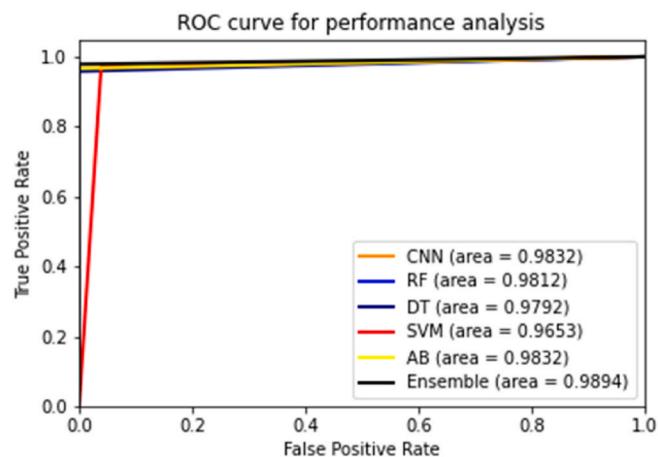


Fig. 8. ROC curve for the classifiers of EMCNet.

Table 7

Performance evaluation of the classifiers used in EMCNet.

Classifiers	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
CNN	98.26	100	96.52	98.22
RF	98.04	100	96.09	98.00
DT	97.82	100	95.65	97.77
SVM	96.52	96.12	96.96	96.54
AB	98.26	100	96.52	98.22
Ensemble	98.91	100	97.82	98.89

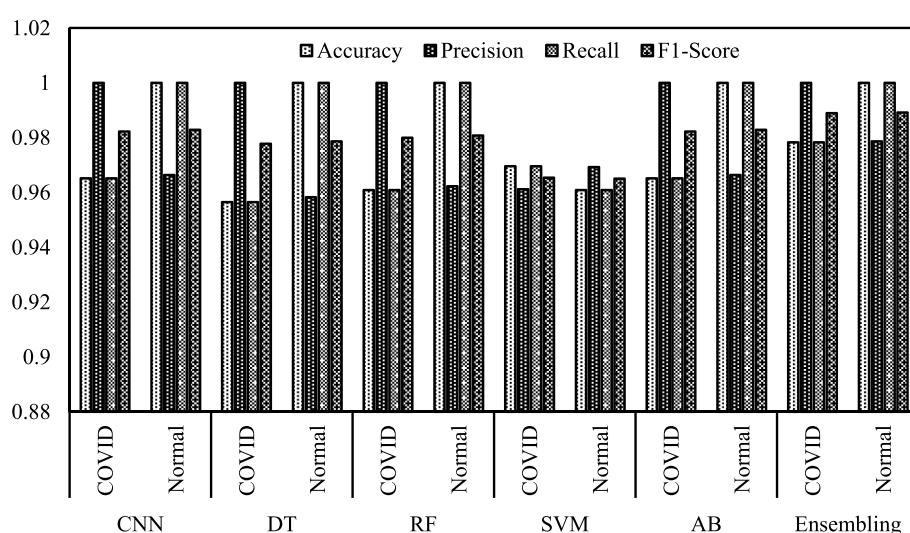


Fig. 7. Graphical representation of the performance of the models used in EMCNet for each class label.

Table 8
Performance analysis of EMCNet in comparison with recent works.

Author	Sources of Dataset	Dataset Details	Model	Accuracy (%)
Zhang et al. [24]	COVID-19 and images: [31]; Normal images [37]:	1531 (COVID-19 = 100, Normal = 1431)	18 layer residual CNN	72.31
Tsiknakis et al. [27]	COVID-19 images: [31]; Normal and Pneumonia images [34, 35]:	572 (COVID-19 = 122, Normal = 150, Bacteria pneumonia = 150, Virus pneumonia = 150)	Inception V3	76
Loey et al. [21]	COVID-19 images: [31]; Normal and Pneumonia images [34]:	306 (COVID-19 = 69, Normal = 79, Bacteria pneumonia = 79, Virus pneumonia = 79)	Googlenet	80.56
Oh et al. [23]	COVID-19 images: [31]; Normal and Pneumonia images [36]:	15,043 (COVID-19 = 180, Normal = 8851, pneumonia = 6012)	Patch based CNN	88.9
Rahimzadeh et al. [22]	COVID-19 images: [31]; Normal and Pneumonia images [35]:	15,085 (COVID-19 = 180, Normal = 8851, Pneumonia = 6054)	Xception + ResNet50V2	91.4
Abbas et al. [20]	COVID-19 and SARS images: [31]; Normal images [32, 33]:	196 (COVID-19 = 105, Normal = 80, SARS = 11)	ResNet	95.12
Sethy et al. [28]	COVID-19 images: [31, 38]; Normal and Pneumonia images [34]:	381 (COVID-19 = 127, Normal = 127, Bacteria pneumonia = 63, Virus pneumonia = 64)	Resnet50 + SVM	95.33
Apostolopoulos et al. [25]	COVID-19 images: [31, 38]; Normal and Pneumonia images [34]:	1442 (COVID-19 = 224, Normal = 504, Pneumonia = 714)	MobileNet V2	96.78
EMCNet	COVID-19 images: [31, 41–45]; Normal images [46, 47]:	4600 (COVID-19 = 2300, Normal = 2300)	CNN + Ensemble of ML Classifiers	98.91

improving early diagnostics and treatment; and helping control the epidemic.

Declaration of competing interest

The authors declare no conflicts of interest.

Acknowledgement

We are grateful to all data hosting providers for their support, and storage capacity. to deliver datasets for our research. We are also thankful to the teachers and staffs of the Department of Computer

Science and Engineering (CSE), Khulna University of Engineering & Technology (KUET), Bangladesh and other friends for their continuous encouragement and suggestions.

References

- [1] Lai CC, Shih TP, Ko WC, Tang HJ, Hsueh PR. Severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2) and coronavirus disease-2019 (COVID-19): the epidemic and the challenges. *Int J Antimicrob Agents* 2020;55(3). <https://doi.org/10.1016/j.ijantimicag.2020.105924>.
- [2] Guan W, et al. Clinical characteristics of coronavirus disease 2019 in China. *N Engl J Med* 2020;382(18):1708–20. <https://doi.org/10.1056/NEJMoa2002032>.
- [3] Sun P, Lu X, Xu C, Sun W, Pan B. Understanding of COVID-19 based on current evidence. *J Med Virol* 2020;92(6):548–51. <https://doi.org/10.1002/jmv.25722>.
- [4] Bhagavathula AS, Aldhaleei WA, Rahmani J, Mahabadi MA, Bandari DK. Knowledge and perceptions of COVID-19 among Health care workers: cross-sectional study. *JMIR Public Heal. Surveill.* 2020;6(2):e19160. <https://doi.org/10.2196/19160>.
- [5] Worldometer. Coronavirus update (live): cases and deaths from COVID-19 virus pandemic. Worldometers; 2020. <https://www.worldometers.info/coronavirus/%0Ahttps://www.worldometers.info/coronavirus/>. [Accessed 30 August 2020].
- [6] Zu ZY, et al. Coronavirus disease 2019 (COVID-19): a perspective from China. *Radiology* 2020;296(2):E15–25. <https://doi.org/10.1148/radiol.2020200490>.
- [7] Tahamtan A, Ardebili A. Real-time RT-PCR in COVID-19 detection: issues affecting the results. *Expert Rev Mol Diagn* 2020;20(5):453–4. <https://doi.org/10.1080/14737159.2020.1757437>.
- [8] Corman VM, et al. Detection of 2019 novel coronavirus (2019-nCoV) by real-time RT-PCR. *Euro Surveill* 2020;25(3). <https://doi.org/10.2807/1560-7917.ES.2020.25.3.2000045>.
- [9] Lauer SA, et al. The incubation period of coronavirus disease 2019 (CoVID-19) from publicly reported confirmed cases: estimation and application. *Ann Intern Med* 2020;172(9):577–82. <https://doi.org/10.7326/M20-0504>.
- [10] Corum J, Zimmer C. Coronavirus vaccine tracker - the New York Times. The New York Times; 2020. accessed Aug. 30, 2020, <https://www.nytimes.com/interactive/2020/science/coronavirus-vaccine-tracker.html>.
- [11] Islam MM, Karray F, Alhajj R, Zeng J. A review on deep learning techniques for the diagnosis of novel coronavirus (COVID-19). 2020 [Online]. Available: <https://arxiv.org/abs/2008.04815>; 2020.
- [12] Xia W, Shao J, Guo Y, Peng X, Li Z, Hu D. Clinical and CT features in pediatric patients with COVID-19 infection: different points from adults. *Pediatr Pulmonol* 2020;55(5):1169–74. <https://doi.org/10.1002/ppul.24718>.
- [13] Rastgarpour M, Shanbehzadeh J. Application of AI techniques in medical image segmentation and novel categorization of available methods and tools. In: IMECS 2011 - international Multi Conference of engineers and computer scientists 2011, 1; 2011. p. 519–23.
- [14] Hwang EJ, et al. Development and validation of a deep learning-based automated detection algorithm for major thoracic diseases on chest radiographs. *JAMA Netw. open* 2019;2(3):e191095. <https://doi.org/10.1001/jamanetworkopen.2019.1095>.
- [15] Cireşan DC, Giusti A, Gambardella LM, Schmidhuber J. Mitosis detection in breast cancer histology images with deep neural networks. *Lect Notes Comput Sci* 2013; 8150 LNCS(PART 2):411–8. https://doi.org/10.1007/978-3-642-40763-5_51.
- [16] Mohsen H, El-Dahshan E-SA, El-Horbaty E-SM, Salem ABM. Classification using deep learning neural networks for brain tumors. *Futur. Comput. Informatics J.* 2018;3(1):68–71. <https://doi.org/10.1016/j.fcij.2017.12.001>.
- [17] Quang D, Xie X, DanQ: a hybrid convolutional and recurrent deep neural network for quantifying the function of DNA sequences. *Nucleic Acids Res* 2016;44(11). <https://doi.org/10.1093/nar/gkw226>.
- [18] Kieu PN, Tran HS, Le TH, Le T, Nguyen TT. Applying Multi-CNNs model for detecting abnormal problem on chest x-ray images. In: Proceedings of 2018 10th international conference on knowledge and systems engineering, KSE 2018; 2018. p. 300–5. <https://doi.org/10.1109/KSE.2018.8573404>.
- [19] Islam MZ, Islam MM, Asraf A. A combined deep CNN-lstm network for the detection of novel coronavirus (COVID-19) using X-ray images. *Informatics in Medicine Unlocked Aug.* 2020;20:100412.
- [20] Abbas A, Abdelsamea MM, Gaber MM. Classification of COVID-19 in chest X-ray images using DeTraC deep convolutional neural network. *Appl Intell* 2020. <https://doi.org/10.1007/s10489-020-01829-7>.
- [21] Loey M, Smarandache F, Khalifa NEM. Within the lack of chest COVID-19 X-ray dataset: a novel detection model based on GAN and deep transfer learning. *Symmetry* 2020;12(4). <https://doi.org/10.3390/SYM12040651>.
- [22] Rahimzadeh M, Attar A. A modified deep convolutional neural network for detecting COVID-19 and pneumonia from chest X-ray images based on the concatenation of Xception and ResNet50V2. *Informatics Med. Unlocked* 2020;19. <https://doi.org/10.1016/j.imu.2020.100360>.
- [23] Oh Y, Park S, Ye JC. Deep learning COVID-19 features on CXR using limited training data sets. *IEEE Trans Med Imag* 2020;39(8):2688–700. <https://doi.org/10.1109/TMI.2020.2993291>.
- [24] Zhang J, Xie Y, Li Y, Shen C, Xia Y. COVID-19 screening on chest X-ray images using deep learning based anomaly detection. 2020. *aeXiv.2002.12338 vol. 1*.
- [25] Apostolopoulos ID, Mpesiana TA. Covid-19: automatic detection from X-ray images utilizing transfer learning with convolutional neural networks. *Phys. Eng. Sci. Med.* 2020;43(2):635–40. <https://doi.org/10.1007/s13246-020-00865-4>.
- [26] Mahmud T, Rahman MA, Fattah SA. 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;122. <https://doi.org/10.1016/j.combiomed.2020.103869>.
- [27] Tsiknakis N, et al. Interpretable artificial intelligence framework for COVID-19 screening on chest X-rays. *Exp. Ther. Med.* 2020. <https://doi.org/10.3892/etm.2020.8797>.
- [28] Sethy PK, Behera SK, Ratha PK, Biswas P. Detection of coronavirus disease (COVID-19) based on deep features and support vector machine. *Int. J. Math. Eng. Manag. Sci.* 2020;5(4):643–51. <https://doi.org/10.33889/IJMEMS.2020.5.4.052>.
- [29] Horry M, et al. X-ray image based COVID-19 detection using pre-trained deep learning models. 2020. <https://doi.org/10.31224/osf.io/wx89s>.
- [30] Hasan MK, Alami MA, Das D, Hossain E, Hasan M. Diabetes prediction using ensembling of different machine learning classifiers. *IEEE Access* 2020;8: 76516–31. <https://doi.org/10.1109/ACCESS.2020.2989857>.
- [31] Cohen JP, Morrison P, Dao L, Roth K, Duong TQ, Ghassemi M. COVID-19 image data collection: prospective predictions are the future. 2020 [Online]. Available: <http://arxiv.org/abs/2006.11988>.
- [32] Candemir S, et al. Lung segmentation in chest radiographs using anatomical atlases with nonrigid registration. *IEEE Trans Med Imag* 2014;33(2):577–90. <https://doi.org/10.1109/TMI.2013.2290491>.
- [33] Jaeger S, et al. Automatic tuberculosis screening using chest radiographs. *IEEE Trans Med Imag* 2014;33(2):233–45. <https://doi.org/10.1109/TMI.2013.2284099>.
- [34] Kermany DS, et al. Identifying medical diagnoses and treatable diseases by image-based deep learning. *Cell* 2018;172(5):1122–31. <https://doi.org/10.1016/j.cell.2018.02.010>. e9.
- [35] RSNA pneumonia detection challenge | Kaggle. accessed Aug. 30, 2020, <https://www.kaggle.com/c/rsna-pneumonia-detection-challenge>.
- [36] Jaeger S, Candemir S, Antani S, Wang Y-XJ, Lu P-X, Thoma G. Two public chest X-ray datasets for computer-aided screening of pulmonary diseases. *Quant Imaging Med Surg* 2014;4(6):475–7. <https://doi.org/10.3978/j.issn.2223-4292.2014.11.20>.
- [37] Wang X, Peng Y, Lu L, Lu Z, Bagheri M, Summers RM. ChestX-ray 8: hospital-scale chest X-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases. 2017. <https://doi.org/10.1109/CVPR.2017.369>.
- [38] Dadario AM. COVID-19 X rays | Kaggle. accessed Aug. 30, 2020, <https://www.kaggle.com/andrewmvd/conv19-x-rays>; 2020.
- [39] Kermany D, Zhang K, Goldbaum M. Large dataset of labeled optical coherence tomography (OCT) and chest X-ray images. 2018. Mendeley Data, vol. 3.
- [40] Smith JW, Everhart JE, Dickson WC, Knowler WC, Johannes RS. Using the ADAP learning algorithm to forecast the onset of diabetes mellitus. 1988.
- [41] COVID 19 chest X-ray [Online]. Available: <https://github.com/agchung>.
- [42] COVID-19 DATABASE | SIRM [Online]. Available: <https://www.sirm.org/en/category/articles/covid-19-database/>.
- [43] Frederick Nat Lab. The cancer imaging archive (TCIA). The Cancer Imaging Archive; 2018. p. 1 [Online]. Available: <https://www.cancerimagingarchive.net/about-the-cancer-imaging-archive-tcia/%0Ahttps://www.cancerimagingarchive.net/%0Ahttps://www.cancerimagingarchive.net/%0Ahttp://www.cancerimagingarchive.net/>.
- [44] Gaillard F. Radiopaedia.org, the wiki-based collaborative Radiology resource. Radiopaedia.org; 2014 [Online]. Available: <http://radiopaedia.org/>.
- [45] Alqudah AM, Qazan S. Augmented COVID-19 X-ray images dataset, 4; 2020. <https://doi.org/10.17632/2FXZ4PX6D8.4>.
- [46] Mooney P. Chest X-ray images (pneumonia) | Kaggle. Kaggle.com; 2018. <https://www.kaggle.com/paultimothymooney/chest-xray-pneumonia%0Ah> <https://www.kaggle.com/paultimothymooney/chest-xray-pneumonia%0Ah> <https://data.mendeley.com/datasets/rscbjbr9sj/2>. [Accessed 30 August 2020].
- [47] NIH chest X-rays. Kaggle; 2018 [Online]. Available: <https://www.kaggle.com/nih-chest-xrays/data>.