

## Article

# In Search of an Efficient and Reliable Deep Learning Model for Identification of COVID-19 Infection from Chest X-ray Images

Abul Kalam Azad , Mahabub-A-Alahi , Imtiaz Ahmed  and Mosabber Uddin Ahmed 

Department of Electrical and Electronic Engineering, University of Dhaka, Dhaka 1000, Bangladesh

\* Correspondence: azad@du.ac.bd (A.K.A.); mosabber.ahmed@du.ac.bd (M.U.A.)

**Abstract:** The virus responsible for COVID-19 is mutating day by day with more infectious characteristics. With the limited healthcare resources and overburdened medical practitioners, it is almost impossible to contain this virus. The automatic identification of this viral infection from chest X-ray (CXR) images is now more demanding as it is a cheaper and less time-consuming diagnosis option. To that cause, we have applied deep learning (DL) approaches for four-class classification of CXR images comprising COVID-19, normal, lung opacity, and viral pneumonia. At first, we extracted features of CXR images by applying a local binary pattern (LBP) and pre-trained convolutional neural network (CNN). Afterwards, we utilized a pattern recognition network (PRN), support vector machine (SVM), decision tree (DT), random forest (RF), and k-nearest neighbors (KNN) classifiers on the extracted features to classify aforementioned four-class CXR images. The performances of the proposed methods have been analyzed rigorously in terms of classification performance and classification speed. Among different methods applied to the four-class test images, the best method achieved classification performances with 97.41% accuracy, 94.94% precision, 94.81% recall, 98.27% specificity, and 94.86% F1 score. The results indicate that the proposed method can offer an efficient and reliable framework for COVID-19 detection from CXR images, which could be immensely conducive to the effective diagnosis of COVID-19-infected patients.



**Citation:** Azad, A.K.; M.-A.-A.; Ahmed, I.; Ahmed, M.U. In Search of an Efficient and Reliable Deep Learning Model for Identification of COVID-19 Infection from Chest X-ray Images. *Diagnostics* **2023**, *13*, 574. <https://doi.org/10.3390/diagnostics13030574>

Academic Editor: Ivan Fan Ngai Hung

Received: 8 November 2022

Revised: 8 December 2022

Accepted: 17 January 2023

Published: 3 February 2023



**Copyright:** © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

**Keywords:** image classification; chest X-ray images; local binary pattern; machine learning classifiers; support vector machine; pattern recognition network; convolutional neural network

## 1. Introduction

The severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2) is responsible for the devastating coronavirus disease 2019 (COVID-19) outbreak which rapidly spread across the world and took over 6.55 million human lives to date [1–3]. The SARS-CoV-2 is a single-stranded and approximately spherically shaped enveloped RNA virus with probable zoonotic origin belonging to the coronaviridae family [4–6]. Although the SARS-CoV-2 infection can occur at all ages of human beings, its symptoms and acuteness vary with age from asymptomatic (in cases of children and young people) to severe respiratory malfunction and organ failure (above 60 y) [2,7]. The most common COVID-19 symptoms are fever, fatigue, and dry cough, whereas, headache, nausea, sore throat, sputum production, hemoptysis, diarrhea, anorexia, and chest pain are noted as less common features of it [8–11]. The gold standard of COVID-19 diagnosis is based on the detection of SARS-CoV-2 nucleic acid at a molecular level by real-time reverse transcription polymerase chain reaction (RT-PCR) [12,13]. However, the laboratory-grade accuracy and specificity of RT-PCR can be adversely affected by the variability of type, method of handling, and collection time of the sample with respect to the disease stage which may lead to a low detection rate below 50% [14–17]. As the early sign of acute COVID-19 reveals symptoms of pneumonia, radiological imaging techniques such as chest X-ray (CXR), computed tomography (CT), and magnetic resonance imaging (MRI) serve as indispensable tools to negate the false negative RT-PCR detection [18–27]. Although CXR is preferred over the

sophisticated CT and MRI in terms of cost, equipment availability, measurement time, and radiation dose, the proper diagnosis of COVID-19 from it poses significant challenges in many parts of the world, arising from the scarcity of experienced radiologists for visual inspection of minute features such as ground-glass opacity and abnormalities of bilateral and interstitial nature in the radiographs relevant to COVID-19 pneumonia [9,11,23,28]. Moreover, the similarity of CXR images of other viral infections and respiratory diseases with COVID-19 makes it very challenging even for expert radiologists [8,21]. Artificial intelligence (AI)-based deep machine learning (ML) offers tremendous support in revealing subtle and hidden image features which may not be apparent in the original CXR and thereby facilitating the automation of image classification and COVID-19 disease diagnosis problem [29]. Moreover, the abundance of publicly available CXR image datasets provided tremendous facilitation in training ML algorithms [30].

Numerous attempts have been made to standardize the detection models for COVID-19 based on CXR images as a complement to RT-PCR using an AI-based DL approach [31–33]. The success of CXR image-based COVID-19 diagnosis with DL potentially relies on the reliable detection of different abnormalities of interstitial and bilateral nature as well as ground-glass opacity irregularities in CXR radiographs [34]. The local binary pattern (LBP) operator is a very popular feature extraction method for CXR radiographs due to its simplicity and low computational cost. The LBP operator divides the images into several parts and assigns binary patterns depending on the texture of the pixel neighbor to extract the feature. On the other hand, a sophisticated and computationally intense DL-based convolutional neural network (CNN), due to its inherent spatial feature extraction and classification capabilities, has emerged as a leading candidate for CXR-based COVID-19 detection [35–38]. With the aid of transfer learning, efficient training of deep CNN with a relatively small CXR dataset sparked its widespread usage in the ML research community to differentiate COVID-19 from non-COVID-19 pneumonia from CXR images [39].

Many of the CXR image-based detection techniques use off-the-rack deep neural networks such as VGG-16, VGG-19, Inception-V3, MobileNet-V2, ResNet-18, ResNet-50, DenseNet-121, CapsNet, and EfficientNet. The CNN network prefers large datasets to avoid bias problem and hence the size of the dataset is crucial for the reliability of result presented in the different literature. For a small dataset containing 125 CXR images, 98.08% accuracy in COVID-19 classification was reported for binary classes in Ref. [40]. Ismael et al., reported an accuracy of 92.63% in COVID-19 prediction working on a CXR image set containing 180 COVID-19 and 200 normal patients' images, where they used a pre-trained ResNet50 model for feature extraction and support vector machine (SVM) with a linear kernel for classification [41]. Afshar et al., used similar two-class CXR images with CapsNet to obtain an accuracy of 98.3% [42]. Frid-Adar et al., achieved 98% accuracy using the ResNet50 over a two-class dataset of no lung opacity and lung opacity [43]. For a large two-class CXR dataset with COVID-19 pneumonia (5805) and non-COVID-19 pneumonia (5300) images, Zhang et al., managed to report 92% accuracy using the DenseNet-121 [44]. Vinod and Prabakaran used a pre-trained CNN in combination with a decision tree classifier to obtain a precision score of 88% on a CXR dataset containing ~300 two-class images [45].

For a three-class classification problem, the achieved accuracy was 95% over a CXR dataset containing normal (104), non-COVID-19 pneumonia (80), and COVID-19 pneumonia (99) [46]. Bukhari et al., reported overall accuracy of 98.24% using 278 images in ResNet-50 with three different image classes comprised of normal (93), COVID-19 (89), and non-COVID-19 pneumonia (96) [47]. Tartaglione et al., achieved a maximum 85% accuracy over 5857 CXR images with three classes comprising normal (1583), bacterial pneumonia (2780), and viral pneumonia (1493) [48]. For a different three-class CXR image dataset containing normal (1314), community-acquired pneumonia (2004), and COVID-19 pneumonia (204), the reported maximum accuracy was 98.71% [49]. Kana et al., obtained 99% accuracy for a dataset with normal (2487), bacterial/viral pneumonia (2507), and COVID-19 (161) [50]. On a similar three-class problem, 91.92% overall accuracy of prediction was achieved in Ref. [44]. Bassi and Attux reported an accuracy of 99.94% for a small

dataset with 439 images [51]. The VGG-19 produced an accuracy of 89.3% for a dataset having normal (300), pneumonia (30), and COVID-19 (260) CXR images [52]. Luz et al., performed classification with EfficientNet on a dataset containing 13,569 CXR images and obtained 93.9% accuracy [53]. The use of a deep CNN DeTraC produced 93.1% accuracy in COVID-19 detection using a small dataset composed of normal (80), SARS (11), and COVID-19 (105) CXR images [54]. Another variant of deep CNN, known as COVID-Net achieved a prediction accuracy of 83.5% on a larger dataset containing normal (358), non-COVID-19 pneumonia (8066), and COVID-19 pneumonia (5538) CXR images [22]. The deep Bayes-SqueezeNet was capable of producing 98.3% COVID-19 prediction accuracy which utilized a dataset with normal (1583), non-COVID-19 pneumonia (4290), and COVID-19 (76) CXR images [55]. Heidari et al., used three classes of CXR images with normal (2880), non-COVID-19 pneumonia (5179), and COVID-19 pneumonia (415) to obtain an accuracy of 94.5% [56]. The classification accuracy can be increased to 97.8% with the VGG-16 CNN model with a similar-sized dataset [57]. Vinod et al., used Covix-Net to yield 96.8% accuracy on a three-class CXR database containing a total of 9000 images divided into normal (3000), COVID-19 (3000), and pneumonia (3000) [58].

In recent times, few investigations emerged with four-class classification capabilities in COVID-19 identification from CXR images. Hussain et al., used a small CXR dataset with normal (138), bacterial pneumonia (145), non-COVID-19 viral pneumonia (145), and COVID-19 (130) and managed to produce 79.52% accuracy [59]. The deep CNN termed as CoroNet was able to produce 89.6% over a CXR dataset composed of normal (310), bacterial pneumonia (330), viral pneumonia (327), and COVID-19 (284) [60]. The use of deep ResNet improved the accuracy of the four-class classification to 92.1% for a very small dataset consisting of 450 CXR images [61]. Attaullah et al., combined patients' symptoms with a total of 800 four-class CXR images and obtained an accuracy of 78.88% [62]. The estimation of the uncertainty in the deep CNN with a Bayesian approach can improve the reliability of the accuracy measurements in a four-class classification problem [63]. The pre-trained and fine-tuned ResNet-50 architecture have been shown to achieve 96.23% accuracy for four-class CXR dataset containing normal (1203), non-COVID-19 viral pneumonia (660), bacterial pneumonia (931), and COVID-19 pneumonia (68) [64]. It is hard to find rigorous and extensive studies of COVID-19 identification within the framework of four-class classification problems using different ML algorithms on a relatively large dataset in the existing literature.

To that cause, here we used different ML algorithms on a comparatively large dataset comprising of 5360 CXR images containing four different classes, i.e., COVID-19, normal, lung opacity, and viral pneumonia, each of which contains 1340 images. The CXR image feature extractions were performed using local binary pattern (LBP) and pre-trained CNN. We used LBP-based PRN, LBP-based SVM, LBP-based DT, LBP-based RF, LBP-based KNN, and CNN-based SVM for image class identification. For the reliable performance analysis of LBP-PRN, a variety of six different training algorithms were used. The performance of SVM classifiers was assessed for nineteen different pre-trained CNNs in feature extraction. The classification performance provided by the ensemble configuration of the three best CNN-based SVM classifiers selected from these aforementioned nineteen different CNNs was also evaluated in this study. Overall, we believe that the results presented here have established an efficient and reliable CNN-based SVM framework for COVID-19 detection from CXR images.

## 2. Materials and Methods

Here, we classified four-class CXR images of COVID-19 infected patients (COVID-19 class), healthy persons (normal class), persons with lung opacity (lung opacity class), and viral pneumonia infected patients (viral pneumonia class) using LBP- and CNN-based feature extraction from the CXR images. The LBP-extracted features were subsequently used to train the PRN, SVM, DT, RF, and KNN-based machine learning classifiers. Moreover,

the pre-trained CNN-derived features were subjected to SVM-based classification. The functional diagram of such feature extraction-based classifiers is depicted in Figure 1.

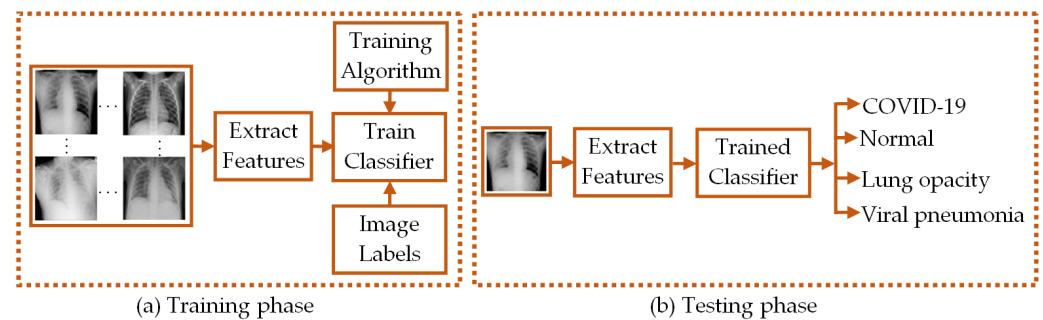


**Figure 1.** Functional diagram depicting the (a) training and (b) testing phases of classifiers.

In both training and testing phases of the classification process, feature extraction algorithms were used to provide the necessary CXR image features. The classifiers are trained with the features obtained in the training phase and the trained classifiers are used to classify the CXR images based on the image features obtained in the testing phase.

### 2.1. Dataset of CXR Images

The dataset of CXR images used in this study has been collected from a public source [65]. Table 1 outlines a brief description of the number of CXR images in the dataset along with the number of images used in this study.

**Table 1.** Total number of chest X-ray images per-class and per-fold.

| Class           | Number of Total CXR Images |      |                   |                  |
|-----------------|----------------------------|------|-------------------|------------------|
|                 | Dataset                    | Used | Training per-Fold | Testing per-Fold |
| COVID-19        | 3616                       | 1340 | 1072              | 268              |
| Normal          | 10,192                     | 1340 | 1072              | 268              |
| Lung opacity    | 6012                       | 1340 | 1072              | 268              |
| Viral pneumonia | 1345                       | 1340 | 1072              | 268              |

We have utilized a total of 5360 CXR images from the four different classes comprising of COVID-19, normal, lung opacity, and viral pneumonia, each of which contains an equal number of 1340 images as shown in Table 1. Four sample CXR images from each class are shown in Figure 2.

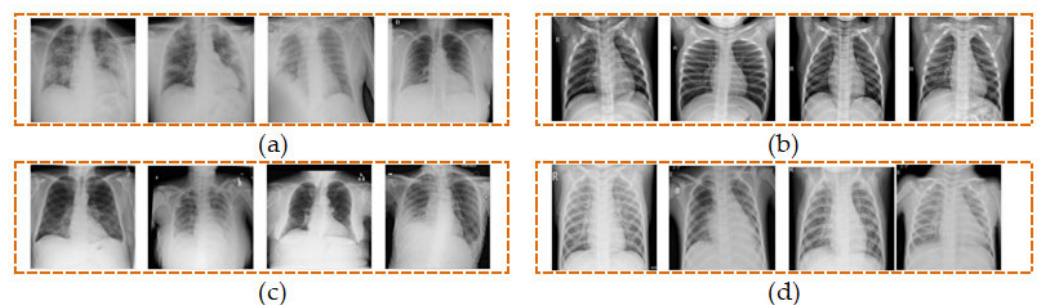


**Figure 2.** Samples of CXR images from four classes of (a) COVID-19, (b) normal, (c) lung opacity, and (d) viral pneumonia.

## 2.2. Extraction of Features from CXR Images

The feature extraction process maps the most significant information of image to a much reduced-size feature vector. In this work, we have demonstrated the use of LBP operator and pre-trained CNN to extract features from CXR images.

### 2.2.1. Extraction of Features from CXR Images Using LBP Operator

The extraction of image features using LBP operator has found extensive applications in the field of image processing [66–69]. The basic principle of LBP operator was first presented in Ref. [66] to describe the texture of an image. This operator works by thresholding the gray levels of neighborhood pixels compared to that of their central pixel in a local circular region. The thresholded values are then summed up in a clockwise direction after being weighted by powers of 2 to obtain the gray levels of the central pixel. The LBP value of a given pixel is given by [66]:

$$L_{P,R}(u, v) = \sum_{m=0}^{P-1} F(g_m - g_c) 2^m, \text{ where } F(x) = \begin{cases} 1 & x \geq 0 \\ 0 & x < 0 \end{cases} \quad (1)$$

In Equation (1),  $P$  is the total number of neighborhoods of the central pixel in a region of the image having radius  $R$ ,  $g_m$  stands for gray level of the neighborhood pixel and  $g_c$  represents the gray level of the central pixel within the region considered in the image.

The mechanism of feature extraction using LBP is depicted in Figure 3. In this illustration, a local region having radius  $R = 1$  is considered to attain an image segment of  $3 \times 3$  pixels. There are 8 pixels ( $P = 8$ ) in this region excluding the central pixels. Then, the thresholding operation is performed through which the pixel value is set to 0 if the gray level of any neighboring pixel is less than that of the center pixel which is 90 in this illustration. Otherwise, it is set to 1. The binary values obtained through this thresholding operation are then weighted by powers of 2 sequentially in a clockwise fashion. These weighted values are finally summed up to obtain the LBP value of the central pixel. The process is repeated to obtain the LBP values of central pixels of other local regions in the whole image.

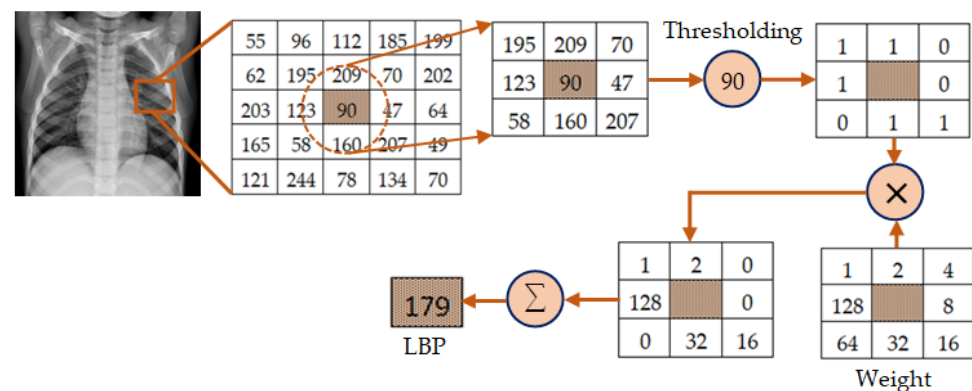


**Figure 3.** Illustration of the mechanism of feature extraction using LBP for  $P = 8$  and  $R = 1$ .

For an  $M \times N$  image, a total of  $2^P$  local binary patterns  $L_{P,R}$  are obtained from Equation (1) which are then represented as a histogram vector  $I$  of length  $0 \leq l \leq (2^P - 1)$  as given by

$$I[l] = \frac{\sum_{u=0}^{M-1} \sum_{v=0}^{N-1} L_{P,R}(u, v)}{M \times N} \quad (2)$$

The  $L_{P,R}$  operator given by Equation (1) generates  $2^P$  different local binary patterns. These local binary patterns vary in accordance with the rotation of image. To avoid the



**Table 2.** Feature layer of different pre-trained CNNs.

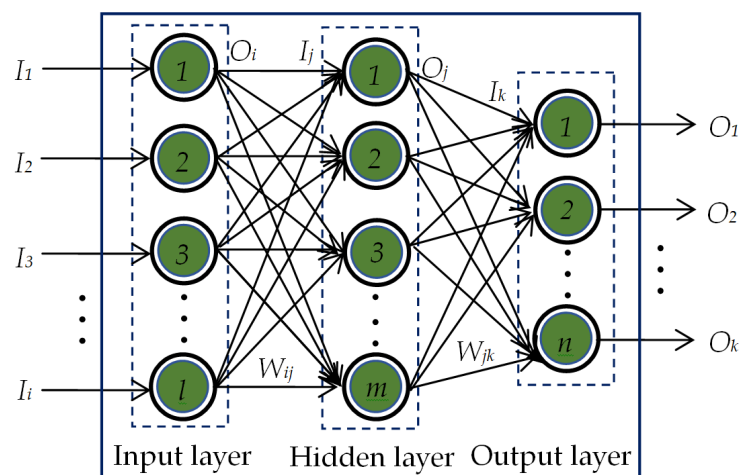
| Pre-Trained CNN    | Name of the Feature Layer                         |
|--------------------|---------------------------------------------------|
| SqueezeNet         | "pool10"                                          |
| GoogLeNet          | "loss3-classifier"                                |
| ResNet-18          | "fc1000"                                          |
| ShuffleNet         | "node202"                                         |
| MobileNet-v2       | "Logits"                                          |
| AlexNet            | "fc8"                                             |
| ResNet-50          | "fc1000"                                          |
| EfficientNet-b0    | "efficientnet-b0   model   head   dense   MatMul" |
| DarkNet-19         | "avg1"                                            |
| ResNet-101         | "fc1000"                                          |
| Inception-v3       | "predictions"                                     |
| VGG-16             | "fc8"                                             |
| Xception           | "predictions"                                     |
| VGG-19             | "fc8"                                             |
| NasNet-Mobile      | "predictions"                                     |
| DarkNet-53         | "conv53"                                          |
| InceptionResNet-v2 | "predictions"                                     |
| DenseNet-201       | "fc1000"                                          |
| NasNet-Large       | "predictions"                                     |

### 2.3. Classification of CXR Images

In this study, the classification of CXR images has been performed based on the extracted features of CXR images. For such classification, we have utilized several widely used classifiers, i.e., PRN, SVM, DT, RF, and KNN.

#### 2.3.1. Classification of CXR Images Using PRN Classifier

The pattern recognition network (PRN) used in this study is a feedforward neural network comprising the input layer, a single hidden layer and the output layer. The PRN has been trained first by using known input-output patterns to optimize the weights of interconnection between the neurons in its different layers. The basic architecture of the PRN used is shown in Figure 5.

**Figure 5.** The basic architecture of a single-hidden-layer PRN.

The output of the  $i$ th neuron in the input layer equals to the input to the  $i$ th neuron of the same input layer, i.e.,

$$O_i = I_i, i = 1, 2, 3, \dots, l \quad (4)$$

The input  $I_j$  to the  $j$ th neuron in the hidden layer is calculated to be the weighted sum of  $O_i$  and  $W_{ij}$ , where  $W_{ij}$  is the weight connecting  $i$ th neuron in the input layer and

$j$ th neuron in the hidden layer. This  $I_j$  is then passed through the activation function. In this work, hyperbolic tangent sigmoid function is used for the neurons in the hidden layer. Thus, the output of  $j$ th neuron in the hidden layer yields

$$O_j = \frac{2}{1 + e^{-2I_j}} - 1, \text{ where } I_j = \sum_{i=1}^l (W_{ij}O_i) \text{ and } j = 1, 2, 3, \dots, m \quad (5)$$

Similarly, the input  $I_k$  to the  $k$ th neuron in the output layer is calculated to be the weighted sum of  $O_j$  and  $W_{jk}$ , where  $W_{jk}$  is the weight connecting  $j$ th neuron in the hidden layer and  $k$ th neuron in the output layer. Initially, the output of the  $k$ th neuron in the output layer is computed by

$$O_k = \frac{e^{I_k}}{\sum_{k=1}^n e^{I_k}}, \text{ where } I_k = \sum_{j=1}^m (W_{jk}O_j) \text{ and } k = 1, 2, 3, \dots, n \quad (6)$$

This  $O_k$  is then passed through the softmax activation function. For using this activation function, the output of a particular neuron in the output layer is assumed to be 1 if its value calculated by Equation (6) is the maximum. The outputs of all other neurons are considered to be 0.

The training of the PRN is accomplished via backpropagation learning algorithm [70–72]. The feature vectors of the CXR images and their corresponding attributes (i.e., COVID-19, normal, lung opacity, and viral pneumonia) are used as the known input pairs. Once the process of training is over, the trained PRN is tested for the feature vectors of unknown CXR images to determine their attributes. In this work, we have analyzed the classification performances of PRN for six different training algorithms as listed in Table 3.

**Table 3.** Training algorithms used to train the PRN classifier.

| Training Algorithm        | Matlab Function |
|---------------------------|-----------------|
| Gradient Descent (GD)     | "traingd"       |
| GD with Momentum          | "traingdm"      |
| Variable Learning Rate GD | "traingdx"      |
| Levenberg–Marquardt       | "trainlm"       |
| Resilient Backpropagation | "trainrp"       |
| Scaled Conjugate Gradient | "trainscg"      |

### 2.3.2. Classification of CXR Images Using SVM Classifier

SVM is a popular supervised learning algorithm widely used for data classification in machine learning [73–75]. The ultimate function of SVM in data classification applications is to create the best decision boundary, called hyperplane that facilitates the classification among different classes. The utilization of hyperplane performs well for linearly separable data [73,74]. In case of linearly inseparable data classification, SVM essentially utilizes a method called kernel trick by which the linearly inseparable data are transformed into linearly separable data to be classified by a linear classifier. In this study, Gaussian radial basis function (RBF) has been employed as the SVM kernel as it provides better performances in many machine-learning applications [75,76]. Such RBF is defined by [76]:

$$\kappa(x, x_c) = \exp\left(-\frac{\|x - x_c\|^2}{2\sigma^2}\right). \quad (7)$$

In Equation (7),  $\|x - x_c\|$  is the Euclidean distance of  $x$  from the center  $x_c$  of the function and the parameter  $\sigma$  controls the smoothness of the function. Since the value of  $\kappa$  decreases with the increase in Euclidean distance, the use of Equation (7) can approximate the local characteristics of a nonlinear function closed to  $x_c$ . The RBF kernel nonlinearly projects two-dimensional original features onto a three-dimensional space. As a result, the linearly

inseparable data can be separated by using an appropriate hyperplane. The data grouping approach of SVM via RBF kernel is depicted in Figure 6.

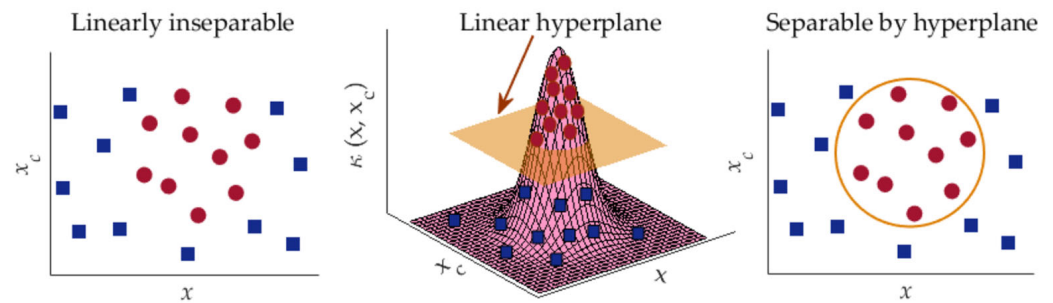


**Figure 6.** Data classification approach of SVM using kernel trick.

In SVM, the known input pairs are employed for the optimization of parameters of classification model. This optimized model is then applied to classify unknown samples. Since SVM can match various data groups acquired from training phase, it can identify images with same categories.

### 2.3.3. Classification of CXR Images Using DT Classifier

The DT classifier used in this study is based on classification and regression tree (CART) algorithm that employs binary tree structure [77–80]. In such CART-based DT classifier, the tree structure consists of nodes, which are linked via branches as depicted in Figure 7.

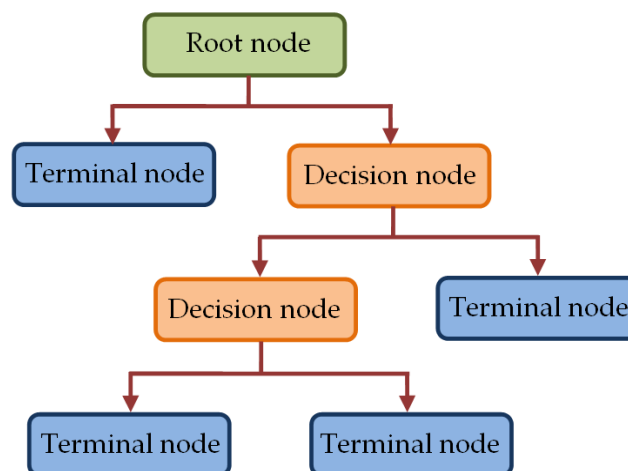


**Figure 7.** The tree structure in DT classifier.

The topmost node in Figure 7, called the root node, utilizes all samples to be classified based on various features to create sub-groups. A sub-group is further split in the decision node to create more sub-groups to be split by other decision nodes. Alternatively, such splitting results in final nodes (terminal nodes), which represent the label of the class. The grouping of all samples in the root node and that in the decision nodes is performed based on a predefined criterion. In our study, we have utilized Gini index criterion to construct the trees. For a group of samples  $D$  having  $c$  classes, such Gini index is calculated by [79].

$$G(D) = 1 - \sum_{i=1}^c (P_i)^2 \quad (8)$$

where  $P_i$  corresponds to the probability of class  $i = 1, 2, \dots, c$  in  $D$ .

### 2.3.4. Classification of CXR Images Using RF Classifier

Random forest classifiers utilize multiple decision trees to form forest-like structures by employing randomly selected subsets of features from the feature set of the samples [78,80]. In this ensemble learning procedure, multiple DT-based classifiers are fitted on different subsets of features. Each of the different trees in RF classifier provides the label of a class. The class labels provided by individual trees go through a voting process and the label obtained through the majority voting is considered the final class label. The working principle of a generalized RF classifier is illustrated in Figure 8.

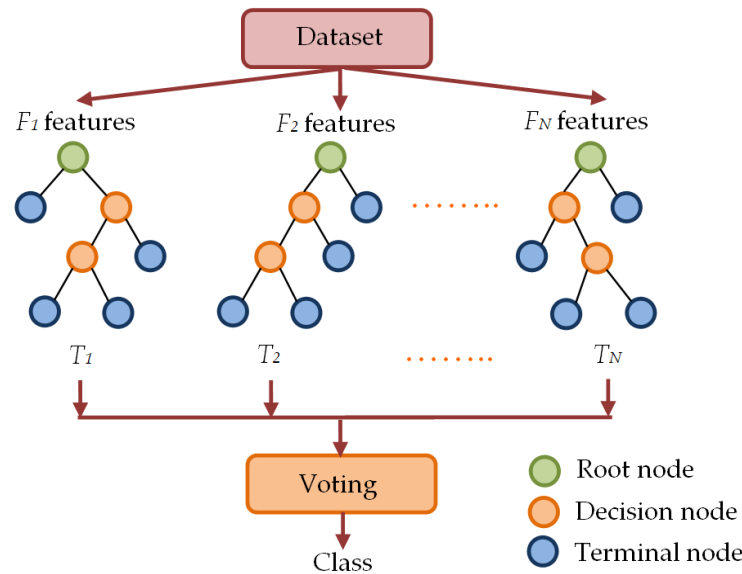


**Figure 8.** Multiple decision tree-based forest-like structure in RF classifier.

As shown in Figure 8, the RF classifier randomly picks subsets of features to construct random decision trees  $T_1, T_2, \dots, T_N$  with corresponding labels  $L_i$  of class  $i = 1, 2, \dots, c$ . Then, the process of majority voting is utilized to determine the final class label  $L$  of the RF classifier.

### 2.3.5. Classification of CXR Images Using KNN Classifier

The KNN is a non-parametric machine learning classifier that utilizes the similarity between the available training samples and the new sample to be classified [80–82]. The class labels for KNN classifier are determined by calculating the closeness among each test sample and the training samples in  $n$ -dimensional space. The classification of a new test sample using KNN classifier is illustrated in Figure 9.

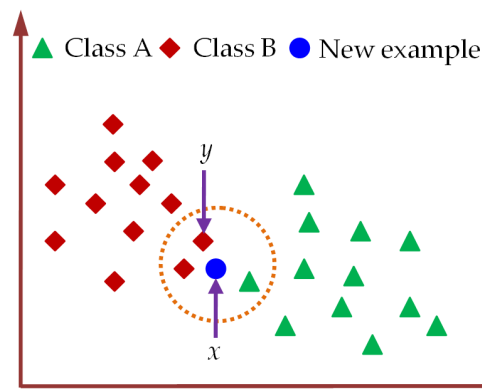


**Figure 9.** The assigning of class label to a new test sample using KNN classifier.

For the classification of the new test sample using KNN classifier, the number (k) of neighbors is fixed first. Then, the Euclidean distance between the new test sample and training samples is determined by Equation (9) to select k-nearest neighbors [82]. Next, the number of training samples belonging to a particular category is counted among these k-nearest neighbors. Finally, the label of the new test sample is assigned to the class for which the number of neighbors among these k-nearest neighbors is maximum. For k = 3, the new example sample in Figure 9 is categorized as class B by KNN classifiers. In practical applications of KNN classifiers, k is usually selected to be an odd number which can minimally be k = 1 [82].

$$d_E = \sum_{i=1}^n (x_i - y_i)^2 \quad (9)$$

### 2.3.6. Classification of CXR Images Using Ensemble-CNN Based SVM Classifier

Ensemble technique combines diverse models together to yield better performance than any of the constituent models. Although CNN based classifiers offer good classification performance, they can also be utilized in ensemble configuration to build robust and highly reliable classification model [83–86]. In this study, we have explored the performance of CNN-based SVMs in such ensemble configuration to further improve the classification accuracy in CXR image classification. To do so, we have identified 3 best CNNs among nineteen different pre-trained CNNs listed in Table 2 based on their classification performances. The CXR images have been applied directly to each of the 3 different CNNs for extracting feature vectors. Three SVMs are then utilized separately to classify the CXR images based on the extracted feature vectors. The mode statistics of SVMs derived CXR image class labels were considered as the ultimate image class as depicted in Figure 10.

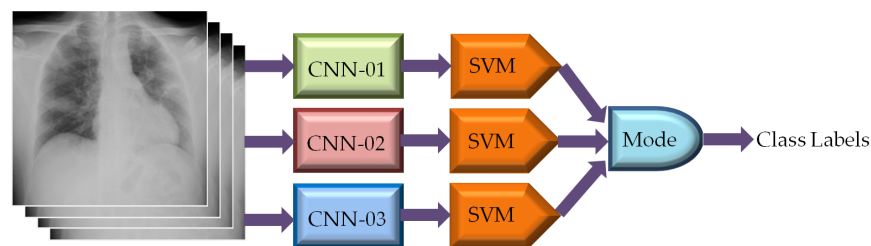


**Figure 10.** Classification of CXR images using ensemble-CNN-based SVM classifier.

### 2.4. Performance Evaluation of Classifiers

In this study, we have adopted five-fold cross-validation to generalize the performance of the classifiers. Each of the five folds contains different combinations of 4288 (i.e., 1072 from each class) CXR images for training and 1072 (i.e., 268 from each class) CXR images for testing the classifier as listed in Table 1. The processes of data splitting for five-fold cross-validation and performance evaluation of classifiers are illustrated in Figure 11.

|        | Section 1 | Section 2 | Section 3 | Section 4 | Section 5 | Fold score( $F_i$ )                                 |
|--------|-----------|-----------|-----------|-----------|-----------|-----------------------------------------------------|
| Fold 1 | Test      | Train     | Train     | Train     | Train     | $F_1$                                               |
| Fold 2 | Train     | Test      | Train     | Train     | Train     | $F_2$                                               |
| Fold 3 | Train     | Train     | Test      | Train     | Train     | $F_3$                                               |
| Fold 4 | Train     | Train     | Train     | Test      | Train     | $F_4$                                               |
| Fold 5 | Train     | Train     | Train     | Train     | Test      | $F_5$                                               |
|        |           |           |           |           |           | Model Score( $M$ ) = $\frac{1}{5} \sum_{i=1}^5 F_i$ |

**Figure 11.** Data splitting and performance evaluation processes in five-fold cross-validation.

To evaluate the performance of the classifiers, we first computed the performance score per-class in a particular fold. For this, four confusion matrix parameters  $TP$  (true positive),  $TN$  (true negative),  $FP$  (false positive), and  $FN$  (false negative) were estimated to calculate the four performance scores, namely, accuracy, precision, recall, and specificity by applying Equations (10)–(13) for each of the four different image classes of COVID-19, normal, lung opacity, and viral pneumonia in case of a particular fold.

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + FN + TN} \times 100(\%) \quad (10)$$

$$\text{Precision} = \frac{TP}{TP + FP} \times 100(\%) \quad (11)$$

$$\text{Recall} = \frac{TP}{TP + FN} \times 100(\%) \quad (12)$$

$$\text{Specificity} = \frac{TN}{TN + FP} \times 100(\%) \quad (13)$$

The per-fold performance scores stem from the average over the four-class performances. The ultimate performance score is the average obtained from the five-fold performances. In addition to four performance metrics (i.e., accuracy, precision, recall, and specificity), the ultimate performances of the classifiers have also been assessed in terms of F1 score as given by Equation (14), in which the precision and the recall scores used are the ultimate performance scores of the classifier.

$$\text{F1 score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \times 100(\%) \quad (14)$$

In this study, we have extracted the image features of the training images offline. The classifiers have also been trained on the extracted features offline during the training phase. However, the feature extraction and classification of different trained classifiers in the testing phase are performed online to facilitate comparative analysis of their runtime. The whole system was implemented by MATLAB R2021a on a workstation with Intel® Core™ i7-11700@2.50 GHz, 8 cores 16 logical processors, HP Ex900 M.2 500 GB PCIe NVMe Internal SSD, Gigabyte GeForce RTX 2060 OC 6 GB Graphics card, and 16 GB DDR4 RAM memory.

### 3. Results and Discussion

In our demonstration of four-class (i.e., COVID-19, normal, lung opacity, and viral pneumonia) classification, we have extracted the feature vectors of CXR images by using the LBP operator as well as pre-trained CNNs. Then, the classification has been performed by PRN, SVM, DT, RF, and KNN classifiers based on those extracted feature vectors. We have also evaluated the classification performances of LBP-based PRN, LBP-based SVM, LBP-based DT, LBP-based RF, LBP-based KNN, CNN-based SVM, and ensemble-CNN-based SVM by adopting the process of five-fold cross-validation as depicted in Figure 11. For instance, the confusion matrix for the testing images in a particular fold (i.e., Fold 1) is shown in Figure 12 for LBP-based SVM.

The confusion matrix in Figure 12 corroborates LBP-based classifiers' success and is used to compute the performances of LBP-based SVM for each of the four different classes in Fold 1 by using Equations (10)–(13) as shown in Table 4.

It is observed in Table 4 that per-class classification accuracy for the test CXR images of each of the four different classes is around 90%. Now, we have averaged these per-class accuracies obtained for this fold to compute fold accuracy. For this Fold 1, LBP-based SVM yields 90.81% accuracy. We have also calculated the fold precision (81.62%), fold recall (81.68%), and fold specificity (93.90%) by averaging per-class precision, recall, and specificity, respectively, for Fold 1. In a similar fashion, we have calculated the per-fold

accuracy, precision, recall, and specificity for the other four folds in the case of LBP-based SVM. The results for each of the five folds are listed in Table 5.

|              |                 |              |        |              |                 |
|--------------|-----------------|--------------|--------|--------------|-----------------|
| Output Class | COVID-19        | 214          | 7      | 25           | 22              |
|              | Normal          | 3            | 227    | 24           | 14              |
|              | Lung opacity    | 21           | 2      | 232          | 13              |
|              | Viral pneumonia | 23           | 30     | 13           | 202             |
|              |                 | COVID-19     | Normal | Lung opacity | Viral pneumonia |
|              |                 | Target Class |        |              |                 |

**Figure 12.** Confusion matrix for the test CXR images in first fold (Fold 1) using LBP-based SVM.

**Table 4.** Classification performance for the test CXR images in Fold 1 using LBP-based SVM.

| Class           | Accuracy (%) | Precision (%) | Recall (%)   | Specificity (%) |
|-----------------|--------------|---------------|--------------|-----------------|
| COVID-19        | 90.58        | 79.85         | 81.99        | 93.34           |
| Normal          | 92.54        | 84.70         | 85.34        | 94.91           |
| Lung opacity    | 90.86        | 86.57         | 78.91        | 95.37           |
| Viral pneumonia | 89.27        | 75.37         | 80.48        | 91.96           |
| <b>Average</b>  | <b>90.81</b> | <b>81.62</b>  | <b>81.68</b> | <b>93.90</b>    |

**Table 5.** Classification performance per-fold for the test CXR images using LBP-based SVM.

| Fold           | Accuracy (%) | Precision (%) | Recall (%)   | Specificity (%) |
|----------------|--------------|---------------|--------------|-----------------|
| 1              | 90.81        | 81.62         | 81.68        | 93.90           |
| 2              | 90.30        | 80.60         | 80.77        | 93.53           |
| 3              | 91.09        | 82.18         | 82.16        | 94.06           |
| 4              | 90.86        | 81.72         | 81.82        | 93.91           |
| 5              | 81.25        | 62.50         | 72.59        | 87.50           |
| <b>Average</b> | <b>88.86</b> | <b>77.72</b>  | <b>79.80</b> | <b>92.58</b>    |

The overall performance of this classification model is computed by averaging five per-fold performances as listed in Table 5. Consequently, the overall accuracy, precision, recall, and specificity for using LBP-based SVM have been found to be 88.86%, 77.72%, 79.80%, and 92.58%, respectively. The overall F1 score for LBP-based SVM classification turned out to be 78.75% following Equation (14).

Next, we have applied a pattern recognition network (PRN) to classify four classes of CXR images utilizing the image feature vectors obtained from the LBP operator. In this classification process, we have also employed five-fold cross-validation to generalize the overall performance of LBP-based PRN. In this study, we have also explored the effects of applying six different training algorithms to train the PRN as listed in Table 3. The overall classification performances of LBP-based PRN for adopting each of the six training algorithms are shown in Figure 13.

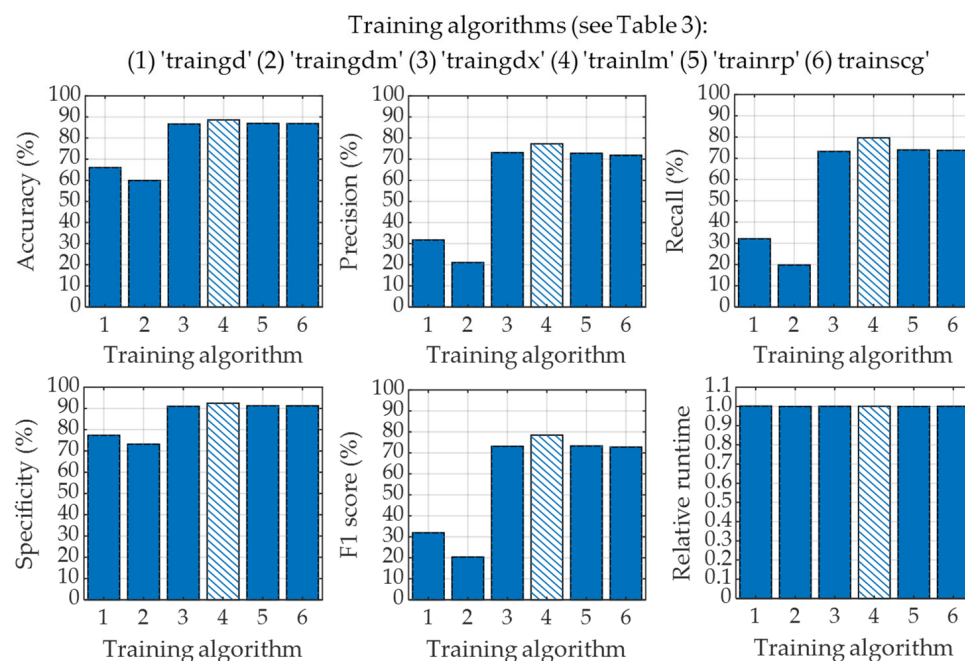


**Figure 13.** Classification performance for using different training algorithms in LBP-based PRN.

The performances of LBP-based PRN vary with training algorithms used to train the PRN as seen in Figure 13. It is evident that gradient descent (“traingd”) and gradient descent with momentum (“traingdm”) training algorithms failed to perform well when used with LBP-based PRN. It can also be observed in Figure 13 that the performances of LBP-based PRN are comparable if such PRN is trained with variable learning rate gradient descent (“traingdx”), Levenberg–Marquardt (“trainlm”), resilient backpropagation (“trainrp”) and scaled conjugate gradient (“trainscg”) learning algorithms. However, the training of LBP-based PRN by adopting the Levenberg–Marquardt algorithm provides the best performances with accuracy, precision, recall, specificity, and F1 score of 88.61%, 77.28%, 79.60%, 92.44%, and 78.42%, respectively. It is worth mentioning that all six different algorithms mentioned in Figure 13 have very similar runtime in the testing phase.

In a similar fashion, we have determined the classification performances of LBP-based DT, LBP-based RF, and LBP-based KNN. For instance, the overall accuracies provided by LBP-based DT, LBP-based RF, and LBP-based KNN are computed to be 83.77%, 87.43%, and 84.58%, respectively. The SVM yields the best accuracy in all cases of LBP-based machine learning classifiers used in this study. Consequently, we have only considered the SVM classifier for classifying CXR images in the next stage.

In this stage, we have extracted feature vectors from the CXR images by applying a deep learning algorithm where features are taken from the fully connected layer at the end of the CNN. To accomplish this feature extraction using CNN, we have employed a total of nineteen different pre-trained CNNs as listed in Table 2. After extracting the feature vectors from the CXR images with CNNs, we have applied the SVM classifier in this stage for the four-class classification of CXR images as described previously. In this stage, we have also applied five-fold cross-validation to generalize the performance of CNN-based SVM classifiers. The overall classification performances of CNN-based SVM for adopting each of the nineteen different pre-trained CNNs along with the LBP-based different machine learning classifiers are shown in Figure 14.

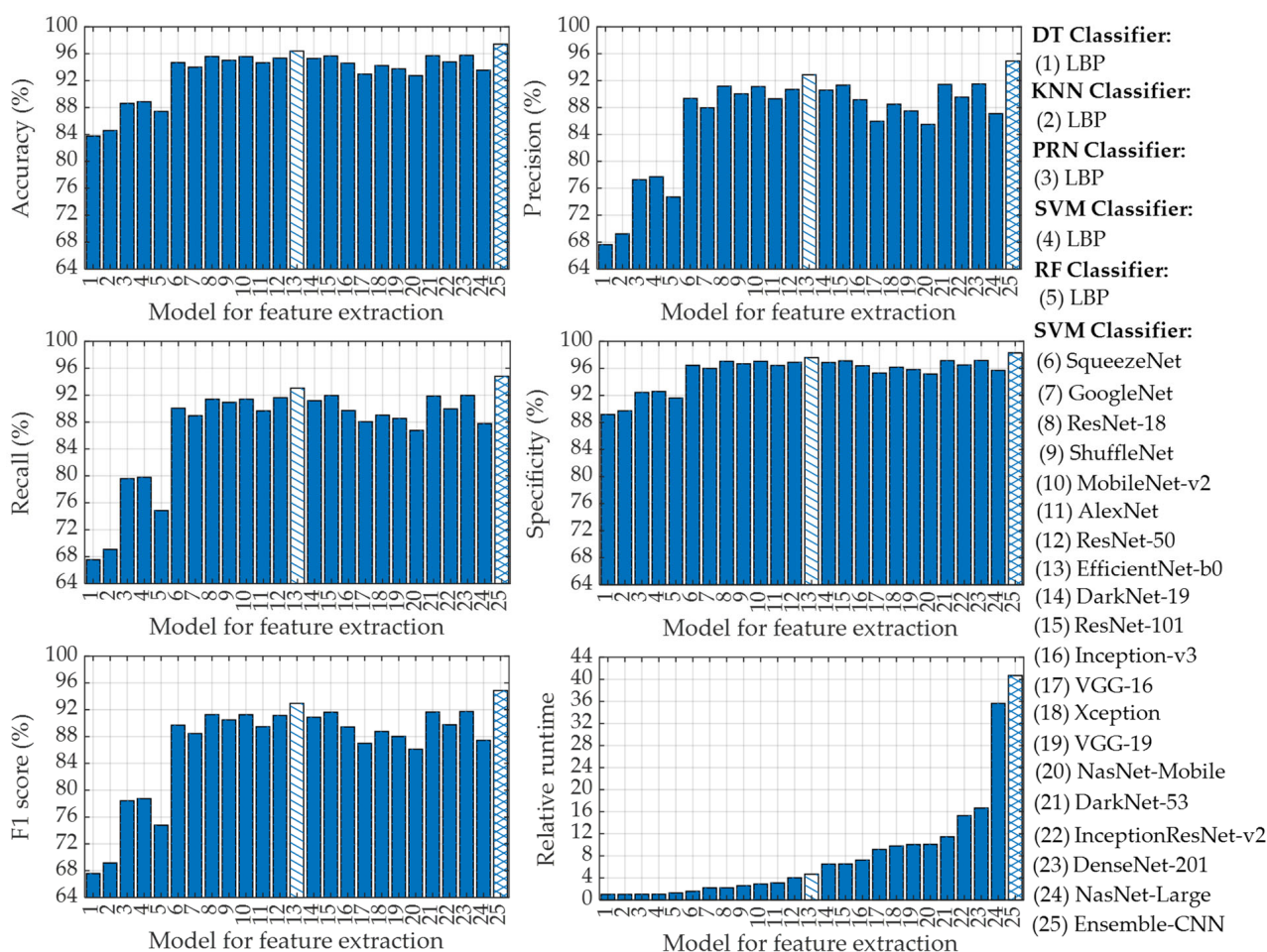


**Figure 14.** Classification performances for using LBP-based DT, LBP-based KNN, LBP-based PRN, LBP-based SVM, LBP-based RF, CNN-based SVM, and ensemble-CNN-based SVM classifiers.

It is easily observed in Figure 14 that each of the nineteen CNN-based SVM classifiers outperforms LBP-based different classifiers (i.e., DT, KNN, PRN, SVM, and RF). This is due to the fact that CNNs utilize a deep learning algorithm that is extremely powerful to extract feature vectors from the CXR images [60–63]. For instance, the lowest accuracy in Figure 14 provided by the pre-trained CNN model of the NasNet-Mobile-based SVM is 92.74%, which is even better than that of LBP-based DT (83.77%), LBP-based KNN (84.58%), LBP-based PRN (88.61%), LBP-based SVM (88.86%), and LBP-based RF (87.43%) classifiers. However, the classification performances of SVM utilizing image feature vectors extracted with EfficientNet-b0 (model 13 in Figure 14) are the best among the nineteen pre-trained CNN architectures as can be seen from Figure 14. Such “EfficientNet-b0” pre-trained CNN-based SVM can achieve overall accuracy, precision, recall, specificity, and F1 score of 96.39%, 92.86%, 93.04%, 97.59%, and 92.95%, respectively.

To further improve the classification performance of CNN-based SVM, we have finally utilized ensemble-CNN-based SVM as described in Section 2.3.6. To effectively utilize such ensemble configuration, we have selected the best three pre-trained CNNs (i.e., EfficientNet-b0, DenseNet-201, and DarkNet-53) among the nineteen different CNN architectures used in this study based on their classification metrics. The classification performances of ensemble-CNN-based SVM have also been plotted in Figure 14 (model 25) for the purpose of comparison. The topmost classification performances provided by different feature extraction-based classifiers are listed in Table 6.

**Table 6.** Topmost classification performance provided by different classifiers.

| Classifier                            | Accuracy (%) | Precision (%) | Recall (%)   | Specificity (%) | F1 Score (%) |
|---------------------------------------|--------------|---------------|--------------|-----------------|--------------|
| LBP + DT                              | 83.77        | 67.62         | 67.54        | 89.18           | 67.58        |
| LBP + KNN                             | 84.58        | 69.22         | 69.08        | 89.72           | 69.15        |
| LBP + RF                              | 87.43        | 74.70         | 74.85        | 91.62           | 74.77        |
| LBP + PRN (“trainlm”)                 | 88.61        | 77.28         | 79.60        | 92.44           | 78.42        |
| LBP + SVM                             | 88.86        | 77.72         | 79.80        | 92.58           | 78.75        |
| DarkNet-53 + SVM                      | 95.72        | 91.44         | 91.88        | 97.15           | 91.66        |
| DenseNet-201 + SVM                    | 95.76        | 91.51         | 91.96        | 97.17           | 91.74        |
| EfficientNet-b0 + SVM                 | 96.39        | 92.86         | 93.04        | 97.59           | 92.95        |
| <b>Ensemble-CNN + SVM<sup>1</sup></b> | <b>97.41</b> | <b>94.91</b>  | <b>94.81</b> | <b>98.27</b>    | <b>94.86</b> |

<sup>1</sup> Ensemble configuration of EfficientNet-b0, DenseNet-201, and DarkNet-53 pre-trained CNNs

It is seen in Table 6 that the classification performances attained for using ensemble-CNN-based SVM are the highest among the 25 different classifiers adopted in this study. For instance, the ensemble-CNN-based SVM can improve the classification accuracy by ~1% as compared to the best CNN-based SVM (i.e., EfficientNet-b0-based SVM). These overall performances of EfficientNet-b0-based SVM and ensemble-CNN-based SVM are more promising compared to some recently published results as listed in Table 7.

**Table 7.** Comparison of recently published studies on COVID-19 detection from CXR images<sup>1</sup>.

| Authors             | Methods                      | Total Images | Classes  | A (%)        | P (%)        | R (%)        | S (%)        | F1 (%)       |
|---------------------|------------------------------|--------------|----------|--------------|--------------|--------------|--------------|--------------|
| Vinod et al. [45]   | CNN + DT                     | 300          | 2        | -            | 88.00        | 83.00        | -            | 85.00        |
| Ozturk et al. [40]  | DarkCovidNet                 | 125          | 2        | 98.08        | 98.03        | 95.13        | 95.3         | 96.51        |
| Wang et al. [22]    | COVID-Net                    | 13,962       | 3        | 83.50        | -            | 91.00        | -            | -            |
| Heidari et al. [56] | CNN + CAD                    | 8474         | 3        | 94.50        | 98.40        | -            | 98.00        | 94.00        |
| Vinod et al. [58]   | Covix-Net                    | 9000         | 3        | 96.80        | 98.00        | 99.00        | -            | -            |
| Khan et al. [60]    | CoroNet                      | 1251         | 4        | 89.60        | 93.00        | 98.20        | 96.40        | 95.60        |
| Farooq et al. [64]  | COVID-ResNet                 | 2862         | 4        | 96.23        | -            | -            | -            | -            |
| <b>This paper</b>   | <b>EfficientNet-b0 + SVM</b> | <b>5360</b>  | <b>4</b> | <b>96.39</b> | <b>92.86</b> | <b>93.04</b> | <b>97.59</b> | <b>92.95</b> |
| <b>This Paper</b>   | <b>Ensemble-CNN + SVM</b>    | <b>5360</b>  | <b>4</b> | <b>97.41</b> | <b>94.91</b> | <b>94.81</b> | <b>98.27</b> | <b>94.86</b> |

<sup>1</sup> A: Accuracy, P: Precision, R: Recall, S: Specificity, F1: F1 score.

As observed in Table 7, the performances attained by applying an Efficient-b0-based SVM classifier are much better than that achieved in Ref. [60] for four-class classification using 1251 CXR images. The overall accuracy of the Efficient-b0-based SVM classifier is also comparable to that achieved in Ref. [64]. It is to be noted that the dataset used in Ref. [64] is imbalanced as there is a big difference in the number of images in each of the four classes (with only 68 CXR images in the COVID-19 class). However, the ensemble-CNN-based SVM classifier used in this study for four-class classification can provide much-improved classification performances as compared to other methods listed in Table 7 with overall accuracy, precision, recall, specificity, and F1 score of 97.41%, 94.91%, 94.81%, 98.27%, and 94.86%, respectively. To the best of our knowledge, these classification performances rank the best among all other reported values for four-class classification of COVID-19, normal, lung opacity, and viral pneumonia CXR images in the existing literature.

Now we focus on relative runtime comparison analysis among different classifiers that use different feature extraction algorithms as shown in Figure 14. It is evident that LBP-based DT requires the lowest runtime. Thus, the relative runtime of a particular technique is normalized with respect to the runtime taken by LBP-based DT. The relative runtimes of different LBP-based machine learning classifiers are nearly uniform with LBP-based RF being the slowest. However, the relative runtimes of CNN-based SVMs vary in accordance with the depth of the layered architecture of the pre-trained CNNs. Among them, the SqueezeNet-based SVM yields the lowest runtime while the NasNet-Large-based SVM requires the highest relative runtime as can be seen in Figure 14. However, the

relative runtime of the single CNN (i.e., EfficientNet-b0)-based SVM which provides the best classification performances is moderately low as compared to other CNN architectures used in this study. To be specific, the “EfficientNet-b0”-based SVM is 4.63 times slower as compared to LBP-based DT. However, such “EfficientNet-b0”-based SVM can provide significantly improved classification performances compared to that of LBP-based machine learning classifiers as shown in Figure 14. It is also observed in Figure 14 and Table 6 that the ensemble-CNNs-based SVM provides the highest classification performance (e.g., 97.41% accuracy) among all the classifiers used in this study. However, to achieve such high performance, this classifier requires relative runtime of 40.68 (i.e., 40.68 times higher than LBP-based DT), which is ~8.72 times larger than that of “EfficientNet-b0”-based SVM.

#### 4. Conclusions

This paper presents a rigorous study on the identification of COVID-19 infection from CXR images based on machine learning approaches. The feature vectors of CXR images have been extracted successfully by utilizing LBP operator and pre-trained CNNs of nineteen different architectures. Then, PRN, SVM, DT, RF, and KNN classifiers have been applied to classify four-class CXR images comprising COVID-19, normal, lung opacity, and viral pneumonia by utilizing the extracted feature vectors of the CXR images. The performances of LBP-based PRN, LBP-based SVM, LBP-based DT, LBP-based RF, LBP-based KNN, CNN-based SVM, and ensemble-CNN-based SVM classifiers have been investigated in detail on the four-class test images and their performances are analyzed in terms of accuracy, precision, recall, specificity, F1 score, and relative runtime. The effects of using six different learning algorithms used to train the LBP-based PRN are analyzed in detail and the results indicate that the Levenberg–Marquardt learning algorithm provides the best classification performance for using LBP-based PRNs in this study. The results also show that the classification performances of LBP-based classifiers are not up to the mark and are significantly lower than that of CNN-based SVM. Among nineteen different single pre-trained CNN-based SVM classifiers, the use of EfficientNet-b0 CNN architecture performs best in our study. The use of such CNN architecture can achieve overall classification performances of 96.39% accuracy, 92.86% precision, 93.04% recall, 97.59% specificity, and 92.95% F1 score with moderately low relative runtime. To further improve the classification performance of CNN-based SVM, we have also utilized ensemble-CNN-based SVM. Such an ensemble configuration consisting of three pre-trained CNNs (i.e., EfficientNet-b0, DenseNet-201, and DarkNet-53) has provided improved classification performances with 97.41% accuracy, 94.91% precision, 94.81% recall, 98.27% specificity, and 94.86% F1 score but required highest runtime to classify CXR images. We believe that the strategy suggested in this paper will provide doctors and physicians with a complementary tool for the diagnosis and prognosis of COVID-19-infected patients. Moreover, the framework so proposed can be integrated into a decision support system that can diagnose COVID-19 based on CXR images, thus considerably minimizing both human and machine error.

**Author Contributions:** Conceptualization, A.K.A. and M.U.A.; data curation, M.-A.-A.; formal analysis, A.K.A., M.-A.-A., I.A. and M.U.A.; investigation, A.K.A. and M.-A.-A.; methodology, A.K.A., M.-A.-A. and M.U.A.; software, A.K.A.; supervision, M.U.A.; validation, A.K.A. and M.-A.-A.; visualization, A.K.A. and I.A.; writing—original draft, A.K.A. and I.A.; writing—review and editing, A.K.A., M.-A.-A., I.A. and M.U.A. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding. The APC was funded by the University of Dhaka, Dhaka-1000, Bangladesh.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The dataset used in this study is available at the following link: <https://www.kaggle.com/datasets/tawsifurrahman/covid19-radiography-database> (accessed on 12 May 2022).

**Acknowledgments:** We are thankful to Mahfuzur R Khan for providing us with MATLAB 2021a software.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Cui, J.; Li, F.; Shi, Z.-L. Origin and evolution of pathogenic coronaviruses. *Nat. Rev. Microbiol.* **2019**, *17*, 181–192. [CrossRef] [PubMed]
2. Hu, B.; Guo, H.; Zhou, P.; Shi, Z.-L. Characteristics of SARS-CoV-2 and COVID-19. *Nat. Rev. Microbiol.* **2021**, *19*, 141–154. [CrossRef] [PubMed]
3. Wu, J.T.; Leung, K.; Leung, G.M. Nowcasting and forecasting the potential domestic and international spread of the 2019-ncov outbreak originating in wuhan, china: A modelling study. *Lancet* **2020**, *395*, 689–697. [CrossRef] [PubMed]
4. Lai, C.-C.; Shih, T.-P.; Ko, W.-C.; Tang, H.-J.; Hsueh, P.-R. 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*, 105924. [CrossRef]
5. Schoeman, D.; Fielding, B.C. Coronavirus envelope protein: Current knowledge. *Virol. J.* **2019**, *16*, 1–22. [CrossRef] [PubMed]
6. Kumar, A.; Singh, R.; Kaur, J.; Pandey, S.; Sharma, V.; Thakur, L.; Sati, S.; Mani, S.; Asthana, S.; Sharma, T.K.; et al. Wuhan to world: The COVID-19 pandemic. *Front. Cell. Infect. Microbiol.* **2021**, *11*, 596201. [CrossRef]
7. Singhal, T. A review of coronavirus disease-2019 (COVID-19). *Indian J. Pediatr.* **2020**, *87*, 281–286. [CrossRef]
8. Chen, N.; Zhou, M.; Dong, X.; Qu, J.; Gong, F.; Han, Y.; Qiu, Y.; Wang, J.; Liu, Y.; Wei, Y.; et al. Epidemiological and clinical characteristics of 99 cases of 2019 novel coronavirus pneumonia in wuhan, china: A descriptive study. *Lancet* **2020**, *395*, 507–513. [CrossRef]
9. Huang, C.; Wang, Y.; Li, X.; Ren, L.; Zhao, J.; Hu, Y.; Zhang, L.; Fan, G.; Xu, J.; Gu, X.; et al. Clinical features of patients infected with 2019 novel coronavirus in wuhan, china. *Lancet* **2020**, *395*, 497–506. [CrossRef]
10. Wang, D.; Hu, B.; Hu, C.; Zhu, F.; Liu, X.; Zhang, J.; Wang, B.; Xiang, H.; Cheng, Z.; Xiong, Y.; et al. Clinical characteristics of 138 hospitalized patients with 2019 novel coronavirus-infected pneumonia in wuhan, china. *JAMA* **2020**, *323*, 1061–1069. [CrossRef]
11. Guan, W.-J.; Ni, Z.; Hu, Y.; Liang, W.; Ou, C.; He, J.; Liu, L.; Shan, H.; Lei, C.; Hui, D.S.C.; et al. Clinical characteristics of coronavirus disease 2019 in china. *N. Engl. J. Med.* **2020**, *382*, 1708–1720. [CrossRef]
12. Sanders, J.M.; Monogue, M.L.; Jodlowski, T.Z.; Cutrell, J.B. Pharmacologic treatments for coronavirus disease 2019 (COVID-19): A review. *JAMA* **2020**, *323*, 1824–1836. [CrossRef] [PubMed]
13. Wiersinga, W.J.; Rhodes, A.; Cheng, A.C.; Peacock, S.J.; Prescott, H.C. Pathophysiology, transmission, diagnosis, and treatment of coronavirus disease 2019 (COVID-19): A review. *JAMA* **2020**, *324*, 782–793. [CrossRef] [PubMed]
14. Wang, W.; Xu, Y.; Gao, R.; Lu, R.; Han, K.; Wu, G.; Tan, W. Detection of SARS-CoV-2 in different types of clinical specimens. *JAMA* **2020**, *323*, 1843–1844. [CrossRef] [PubMed]
15. Sethuraman, N.; Jeremiah, S.S.; Ryo, A. Interpreting diagnostic tests for SARS-CoV-2. *JAMA* **2020**, *323*, 2249–2251. [CrossRef] [PubMed]
16. Kucirka, L.M.; Lauer, S.A.; Laeyendecker, O.; Boon, D.; Lessler, J. Variation in false-negative rate of reverse transcriptase polymerase chain reaction-based SARS-CoV-2 tests by time since exposure. *Ann. Intern. Med.* **2020**, *173*, 262–267. [CrossRef] [PubMed]
17. Li, Y.; Yao, L.; Li, J.; Chen, L.; Song, Y.; Cai, Z.; Yang, C. Stability issues of rt-pcr testing of SARS-CoV-2 for hospitalized patients clinically diagnosed with COVID-19. *J. Med. Virol.* **2020**, *92*, 903–908. [CrossRef] [PubMed]
18. Bai, H.X.; Hsieh, B.; Xiong, Z.; Halsey, K.; Choi, J.W.; Tran, T.M.L.; Pan, I.; Shi, L.-B.; Wang, D.-C.; Mei, J.; et al. Performance of radiologists in differentiating COVID-19 from non-COVID-19 viral pneumonia at chest ct. *Radiology* **2020**, *296*, E46–E54. [CrossRef]
19. Ai, T.; Yang, Z.; Hou, H.; Zhan, C.; Chen, C.; Lv, W.; Tao, Q.; Sun, Z.; Xia, L. Correlation of chest ct and rt-pcr testing in coronavirus disease 2019 (COVID-19) in china: A report of 1014 cases. *Radiology* **2020**, *296*, E32–E40. [CrossRef]
20. Shatri, J.; Tafilaj, L.; Turkaj, A.; Dedushi, D.; Shatri, M.; Bexheti, S.; Mucaj, S.K. The role of chest computed tomography in asymptomatic patients of positive coronavirus disease 2019: A case and literature review. *J. Clin. Imaging Sci.* **2020**, *10*, 1–4. [CrossRef]
21. Zhao, W.; Jiang, W.; Qiu, X. Deep learning for COVID-19 detection based on ct images. *Sci. Rep.* **2021**, *11*, 1–12. [CrossRef]
22. Wang, L.; Lin, Z.Q.; Wong, A. Covid-net: A tailored deep convolutional neural network design for detection of COVID-19 cases from chest X-ray images. *Sci. Rep.* **2020**, *10*, 1–12. [CrossRef] [PubMed]
23. Dong, D.; Tang, Z.; Wang, S.; Hui, H.; Gong, L.; Lu, Y.; Xue, Z.; Liao, H.; Chen, F.; Yang, F.; et al. The role of imaging in the detection and management of COVID-19: A review. *IEEE Rev. Biomed. Eng.* **2020**, *14*, 16–29. [CrossRef] [PubMed]
24. Syrjala, H.; Broas, M.; Ohtonen, P.; Jartti, A.; Pääkkö, E. Chest magnetic resonance imaging for pneumonia diagnosis in outpatients with lower respiratory tract infection. *Eur. Respir. J.* **2017**, *49*, 1–7. [CrossRef]

25. Bernheim, A.; Mei, X.; Huang, M.; Yang, Y.; Fayad, Z.A.; Zhang, N.; Diao, K.; Lin, B.; Zhu, X.; Li, K.; et al. Chest ct findings in coronavirus disease-19 (COVID-19): Relationship to duration of infection. *Radiology* **2020**, *295*, 685–691. [\[CrossRef\]](#)
26. Fang, Y.; Zhang, H.; Xie, J.; Lin, M.; Ying, L.; Pang, P.; Ji, W. Sensitivity of chest ct for COVID-19: Comparison to rt-pcr. *Radiology* **2020**, *296*, E115–E117. [\[CrossRef\]](#)
27. Xie, X.; Zhong, Z.; Zhao, W.; Zheng, C.; Wang, F.; Liu, J. Chest ct for typical 2019-ncov pneumonia: Relationship to negative rt-pcr testing. *Radiology* **2020**, *296*, E41–E45. [\[CrossRef\]](#) [\[PubMed\]](#)
28. Ng, M.-N.; Lee, E.Y.P.; Yang, J.; Yang, F.; Li, X.; Wang, H.; Lui, M.M.; Lo, C.S.-Y.; Leung, B.; Khong, P.-L.; et al. Imaging profile of the COVID-19 infection: Radiologic findings and literature review. *Radiol. Cardiothorac. Imaging* **2020**, *2*, e200034. [\[CrossRef\]](#) [\[PubMed\]](#)
29. Kallianos, K.; Mongan, J.; Antani, S.; Henry, T.; Taylor, A.; Abuya, J.; Kohli, M. How far have we come? artificial intelligence for chest radiograph interpretation. *Clin. Radiol.* **2019**, *74*, 338–345. [\[CrossRef\]](#)
30. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. *Commun. ACM.* **2017**, *60*, 84–90. [\[CrossRef\]](#)
31. Roberts, M.; Driggs, D.; Thorpe, M.; Gilbey, J.; Yeung, M.; Ursprung, S.; Aviles-Rivero, A.I.; Etmann, C.; McCague, C.; Beer, L.-C.; et al. Common pitfalls and recommendations for using machine learning to detect and prognosticate for COVID-19 using chest radiographs and ct scans. *Nat. Mach. Intell* **2021**, *3*, 199–217. [\[CrossRef\]](#)
32. Chowdhury, M.E.H.; Rahman, T.; Khandakar, A.; Mazhar, R.; Kadir, M.A.; Mahbub, Z.B.; Islam, K.R.; Khan, M.S.; Iqbal, A.; Emadi, N.A.; et al. Can ai help in screening viral and COVID-19 pneumonia? *IEEE Access* **2020**, *8*, 132665–132676. [\[CrossRef\]](#)
33. Rahman, T.; Khandakar, A.; Qiblawey, Y.; Tahir, A.; Kiranyaz, S.; Kashem, S.B.A.; Islam, M.T.; Maadeed, S.A.; Zughaier, S.M.; Khan, M.S.; et al. Exploring the Effect of Image Enhancement Techniques on COVID-19 Detection using Chest X-ray Images. *Comput. Biol. Med.* **2021**, *132*, 104319. [\[CrossRef\]](#)
34. Tuan, D.; Pham, T.D. Classification of COVID-19 chest X-rays with deep learning: New models or fine tuning? *Health Inf. Sci. Syst.* **2021**, *9*, 1–11. [\[CrossRef\]](#)
35. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention 2015*; Springer: Cham, Switzerland, 2015; pp. 234–241. [\[CrossRef\]](#)
36. Rajpurkar, P.; Irvin, J.; Ball, R.L.; Zhu, K.; Yang, B.; Mehta, H.; Duan, T.; Ding, D.; Bagul, A.; Langlotz, C.P.; et al. Deep Learn. Chest Radiogr. Diagn: A retrospective comparison of the CheXNeXt algorithm to practicing radiologists. *PLoS Med.* **2018**, *15*, e1002686. [\[CrossRef\]](#)
37. Lakhani, P.; Sundaram, B. Deep learning at chest radiography: Automated classification of pulmonary tuberculosis by using convolutional neural networks. *Radiology* **2017**, *284*, 574–582. [\[CrossRef\]](#)
38. Ho, T.K.K.; Gwak, J. Multiple feature integration for classification of thoracic disease in chest radiography. *Appl. Sci.* **2019**, *9*, 4130. [\[CrossRef\]](#)
39. Chouhan, V.; Singh, S.K.; Khamparia, A.; Gupta, D.; Tiwari, P.; Moreira, C.; Damaševičius, R.; Albuquerque, V.H.C.D. A novel transfer learning based approach for pneumonia detection in chest X-ray images. *Appl. Sci.* **2020**, *10*, 559. [\[CrossRef\]](#)
40. Ozturk, T.; Talo, M.; Yildirim, E.A.; Baloglu, U.B.; Yildirim, O.; Acharya, U.R. Automated detection of COVID-19 cases using deep neural networks with X-ray images. *Comput. Biol. Med.* **2020**, *121*, 103792. [\[CrossRef\]](#) [\[PubMed\]](#)
41. Ismael, A.M.; Şengür, A. Deep learning approaches for COVID-19 detection based on chest X-ray images. *Expert Syst. Appl.* **2021**, *164*, 114054. [\[CrossRef\]](#)
42. Afshar, P.; Heidarian, S.; Naderkhani, F.; Oikonomou, A.; Plataniotis, K.N.; Mohammadi, A. Covid-caps: A capsule network-based framework for identification of COVID-19 cases from X-ray images. *Pattern Recognit. Lett.* **2020**, *138*, 638–643. [\[CrossRef\]](#) [\[PubMed\]](#)
43. Frid-Adar, M.; Rula Amer, R.; Gozes, O.; Nassar, J.; Greenspan, H. COVID-19 in cxr: From detection and severity scoring to patient disease monitoring. *IEEE J. Biomed. Health Inform.* **2021**, *25*, 1892–1903. [\[CrossRef\]](#) [\[PubMed\]](#)
44. Zhang, R.; Guo, Z.; Sun, Y.; Lu, Q.; Xu, Z.; Yao, Z.; Duan, M.; Liu, S.; Ren, Y.; Huang, L.; et al. COVID-19xraynet: A two-step transfer learning model for the COVID-19 detecting problem based on a limited number of chest X-ray images. *Interdiscip. Sci. Comput. Life Sci.* **2020**, *12*, 555–565. [\[CrossRef\]](#) [\[PubMed\]](#)
45. Vinod, D.N.; Prabakaran, S.R.S. Data science and the role of Artificial Intelligence in achieving the fast diagnosis of COVID-19. *Chaos Solitons Fractals* **2020**, *140*, 110182. [\[CrossRef\]](#) [\[PubMed\]](#)
46. Ezzat, D.; Hassanien, A.E.; Ella, H.A. An optimized deep learning architecture for the diagnosis of COVID-19 disease based on gravitational search optimization. *Appl. Soft Comput.* **2021**, *98*, 106742. [\[CrossRef\]](#)
47. Bukhari, S.U.K.; Bukhari, S.S.K.; Syed, A.; Shah, S.H.S. The diagnostic evaluation of convolutional neural network (cnn) for the assessment of chest X-ray of patients infected with COVID-19. *MedRxiv* **2020**. [\[CrossRef\]](#)
48. Tartaglione, E.; Barbano, C.A.; Berzovini, C.; Calandri, M.; Grangetto, M. Unveiling COVID-19 from chest X-ray with deep learning: A hurdles race with small data. *Int. J. Environ. Res. Public Health.* **2020**, *17*, 6933. [\[CrossRef\]](#)
49. Wang, Z.; Xiao, Y.; Li, Y.; Zhang, J.; Lu, F.; Hou, M.; Liu, X. Automatically discriminating and localizing COVID-19 from community-acquired pneumonia on chest X-rays. *Pattern Recognit.* **2021**, *110*, 107613. [\[CrossRef\]](#)
50. Kana, E.B.G.; Kana, M.G.Z.; Kana, A.F.D.; Kenfack, R.H.A. A web-based diagnostic tool for COVID-19 using machine learning on chest radiographs (cxr). *MedRxiv* **2020**. [\[CrossRef\]](#)

51. Bassi, P.R.A.S.; Attux, R. A deep convolutional neural network for COVID-19 detection using chest X-rays. *Res. Biomed. Eng.* **2022**, *38*, 139–148. [\[CrossRef\]](#)
52. Rahaman, M.M.; Li, C.; Yao, Y.; Kulwa, F.; Rahman, M.S.; Wang, Q.; Qi, S.; Kong, F.; Zhu, X.; Zhao, X. Identification of COVID-19 samples from chest X-ray images using deep learning: A comparison of transfer learning approaches. *J. X-ray Sci. Technol.* **2020**, *28*, 821–839. [\[CrossRef\]](#) [\[PubMed\]](#)
53. Luz, E.; Silva, P.; Silva, R.; Silva, L.; Guimarães, J.; Miozzo, G.; Moreira, G.; Menotti, D. Towards an effective and efficient deep learning model for COVID-19 patterns detection in X-ray images. *Res. Biomed. Eng.* **2022**, *38*, 149–162. [\[CrossRef\]](#)
54. Abbas, A.; Abdelsamea, M.M.; Gaber, M.M. Classification of COVID-19 in chest X-ray images using detrac deep convolutional neural network. *Appl. Intell.* **2021**, *51*, 854–864. [\[CrossRef\]](#) [\[PubMed\]](#)
55. Ucar, F.; Korkmaz, D. Covidiagnosis-net: Deep bayes-squeezenet based diagnosis of the coronavirus disease 2019 (COVID-19) from X-ray images. *Med. Hypotheses* **2020**, *140*, 109761. [\[CrossRef\]](#) [\[PubMed\]](#)
56. Heidari, M.; Mirniaharikandehi, S.; Khuzani, A.Z.; Danala, G.; Qiu, Y.; Zheng, B. Improving the performance of cnn to predict the likelihood of COVID-19 using chest X-ray images with preprocessing algorithms. *Int. J. Med. Inform.* **2020**, *144*, 104284. [\[CrossRef\]](#) [\[PubMed\]](#)
57. Zokaenikoo, M.; Kazemian, P.; Mitra, P.; Kumara, S. Aidcov: An interpretable artificial intelligence model for detection of COVID-19 from chest radiography images. *ACM Trans. Manag. Inf. Syst.* **2021**, *12*, 1–20. [\[CrossRef\]](#)
58. Vinod, D.N.; Jeyavadhanam, B.R.; Zungeru, A.M.; Prabakaran, S.R.S. Fully automated unified prognosis of COVID-19 chest X-ray/CT scan images using Deep Covix-Net model. *Comput. Biol. Med.* **2021**, *136*, 104729. [\[CrossRef\]](#)
59. Hussain, L.; Nguyen, T.; Li, H.; Abbasi, A.A.; Lone, K.J.; Zhao, Z.; Zaib, M.; Chen, A.; Duong, T.Q. Machine-learning classification of texture features of portable chest X-ray accurately classifies COVID-19 lung infection. *Biomed. Eng. Online* **2020**, *19*, 1–18. [\[CrossRef\]](#)
60. Khan, A.I.; Shah, J.L.; Bhat, M.M. Coronet: A deep neural network for detection and diagnosis of COVID-19 from chest X-ray images. *Comput. Methods Programs Biomed.* **2020**, *196*, 105581. [\[CrossRef\]](#)
61. Panahi, A.; Moghadam, R.A.; Akrami, M.; Madani, K. Deep residual neural network for COVID-19 detection from chest X-ray images. *SN Comput. Sci.* **2022**, *3*, 1–10. [\[CrossRef\]](#)
62. Attaullah, M.; Ali, M.; Almufareh, M.F.; Ahmad, M.; Hussain, L.; Jhanjhi, N.; Humayun, M. Initial stage COVID-19 detection system based on patients' symptoms and chest X-ray images. *Appl. Artif. Intell.* **2022**, *36*, 1–20. [\[CrossRef\]](#)
63. Ghoshal, B.; Tucker, A. Estimating uncertainty and interpretability in deep learning for coronavirus (COVID-19) detection. *arXiv* **2020**, arXiv:2003.10769. [\[CrossRef\]](#)
64. Farooq, M.; Hafeez, A. Covid-resnet: A deep learning framework for screening of COVID-19 from radiographs. *arXiv* **2020**, arXiv:2003.14395. [\[CrossRef\]](#)
65. COVID-19 Radiography Database. Available online: <https://www.kaggle.com/datasets/tawsifurrahman/COVID-19-radiography-database> (accessed on 12 May 2022).
66. Ojala, T.; Pietikäinen, M.; Mäenpää, T. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Trans. Pattern Anal. Mach. Intell.* **2002**, *24*, 971–987. [\[CrossRef\]](#)
67. Lahdenoja, O.; Poikonen, J.; Laiho, M. Towards Understanding the Formation of Uniform Local Binary Patterns. *ISRN Mach. Vis.* **2013**, *2013*, 429347. [\[CrossRef\]](#)
68. Liu, L.; Zhao, L.; Long, Y.; Kuang, G.; Fieguth, P. Extended local binary patterns for texture classification. *Image Vis. Comput.* **2012**, *30*, 86–99. [\[CrossRef\]](#)
69. Wang, Y.; Zhao, Y.; Chen, Y. Texture classification using rotation invariant models on integrated local binary pattern and Zernike moments. *EURASIP J. Adv. Signal Process.* **2014**, *2014*, 182. [\[CrossRef\]](#)
70. Rumelhart, D.E.; Hinton, G.E.; Williams, R.J. Learning representations by back-propagating errors. *Nature* **1986**, *323*, 533–536. [\[CrossRef\]](#)
71. Shyla, N.S.J.; Emmanuel, W.R.S. Glaucoma detection and classification using modified level set segmentation and pattern classification neural network. *Multimed Tools Appl.* **2022**, 1–19. [\[CrossRef\]](#)
72. Gündoğdu, S. Improving breast cancer prediction using a pattern recognition network with optimal feature subsets. *Croat Med J.* **2021**, *62*, 480–487. [\[CrossRef\]](#)
73. Almansour, N.A.; Syed, H.F.; Khayat, N.R.; Altheeb, R.K.; Juri, R.E.; Alhiyafi, J.; Alrashed, S.; Olatunji, S.O. Neural network and support vector machine for the prediction of chronic kidney disease: A comparative study. *Comput. Biol. Med.* **2019**, *109*, 101–111. [\[CrossRef\]](#) [\[PubMed\]](#)
74. Guhathakurata, S.; Kundu, S.; Chakraborty, A.; Banerjee, J.S. A novel approach to predict COVID-19 using support vector machine. *Data Sci. COVID-19* **2021**, *18*, 351–364. [\[CrossRef\]](#)
75. Babaei, M.; Moeini, R.; Ehsanzadeh, E. Artificial Neural Network and Support Vector Machine Models for Inflow Prediction of Dam Reservoir (Case Study: Zayandehroud Dam Reservoir). *Water Resour. Manag.* **2019**, *33*, 2203–2218. [\[CrossRef\]](#)
76. Tharwat, A. Parameter investigation of support vector machine classifier with kernel functions. *Knowl. Inf. Syst.* **2019**, *61*, 1269–1302. [\[CrossRef\]](#)
77. Jijo, B.T.; Abdulazeez, A.M. Classification Based on Decision Tree Algorithm for Machine Learning. *J. App. Sci. Technol. Trend.* **2021**, *2*, 20–28. [\[CrossRef\]](#)

78. Dhivyaa, C.R.; Sangeetha, K.; Balamurugan, M.; Amaran, S.; Vetriselvi, T.; Johnpaul, P. Skin lesion classification using decision trees and random forest algorithms. *J. Ambient. Intell. Humaniz. Comput.* **2020**, 1–13. [CrossRef]
79. Daniya, T.; Geetha, M.; Kumar, K.S. Classification and regression trees with gini index. *Adv. Math. Sci. J.* **2020**, 9, 1857–8438. [CrossRef]
80. Imad, M.; Khan, N.; Ullah, F.; Hassan, M.A.; Hussain, A.; Faiza. COVID-19 Classification based on Chest X-ray Images Using Machine Learning Techniques. *J. Comput. Sci. Technol. Studi.* **2020**, 2, 1–11. Available online: <https://al-kindipublisher.com/index.php/jcsts/article/view/531> (accessed on 7 September 2022).
81. Sharmila, A.; Geethanjali, P. DWT Based Detection of Epileptic Seizure from EEG Signals Using Naive Bayes and k-NN Classifiers. *IEEE Access* **2016**, 4, 7716–7727. [CrossRef]
82. Hu, L.Y.; Huang, M.W.; Ke, S.W.; Tsai, C.F. The distance function effect on k-nearest neighbor classification for medical datasets. *SpringerPlus* **2016**, 5, 1304. [CrossRef]
83. Zhou, T.; Lu, H.; Yang, Z.; Qiu, S.; Huo, B.; Dong, Y. The ensemble deep learning model for novel COVID-19 on CT images. *Appl. Soft Comput.* **2021**, 98, 106885. [CrossRef] [PubMed]
84. Mouhafid, M.; Salah, M.; Yue, C.; Xia, K. Deep Ensemble Learning-Based Models for Diagnosis of COVID-19 from Chest CT Images. *Healthcare* **2022**, 10, 166. [CrossRef] [PubMed]
85. Biswas, S.; Chatterjee, S.; Majee, A.; Sen, S.; Schwenker, F.; Sarkar, R. Prediction of COVID-19 from Chest CT Images Using an Ensemble of Deep Learning Models. *Appl. Sci.* **2021**, 11, 7004. [CrossRef]
86. Zhang, B.; Qi, S.; Monkam, P.; Li, C.; Yang, F.; Yao, Y.-D.; Qian, W. Ensemble Learners of Multiple Deep CNNs for Pulmonary Nodules Classification Using CT Images. *IEEE Access* **2019**, 7, 110358–110371. [CrossRef]

**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.