A computer-aided diagnosis system for the classification of COVID-19 and non-COVID-19 pneumonia on chest X-ray images by integrating CNN with sparse autoencoder and feed forward neural network

Gayathri J.L. [a], Bejoy Abraham [a,*], Sujarani M.S. [a], Madhu S. Nair [b]

[a] Department of Computer Science and Engineering, College of Engineering Perumon, Kollam, 691 601, Kerala, India
[b] Artificial Intelligence & Computer Vision Lab, Department of Computer Science, Cochin University of Science and Technology, Kochi, 682 022, Kerala, India

ABSTRACT

Several infectious diseases have affected the lives of many people and have caused great dilemmas all over the world. COVID-19 was declared a pandemic caused by a newly discovered virus named Severe Acute Respiratory Syndrome Coronavirus 2 (SARS-CoV-2) by the World Health Organisation in 2019. RT-PCR is considered the golden standard for COVID-19 detection. Due to the limited RT-PCR resources, early diagnosis of the disease has become a challenge. Radiographic images such as Ultrasound, CT scans, X-rays can be used for the detection of the deathly disease. Developing deep learning models using radiographic images for detecting COVID-19 can assist in countering the outbreak of the virus. This paper presents a computer-aided detection model utilizing chest X-ray images for combating the pandemic. Several pre-trained networks and their combinations have been used for developing the model. The method uses features extracted from pre-trained networks along with Sparse autoencoder for dimensionality reduction and a Feed Forward Neural Network (FFNN) for the detection of COVID-19. Two publicly available chest X-ray image datasets, consisting of 504 COVID-19 images and 542 non-COVID-19 images, have been combined to train the model. The method was able to achieve an accuracy of 0.9578 and an AUC of 0.9821, using the combination of InceptionResnetV2 and Xception. Experiments have proved that the accuracy of the model improves with the usage of sparse autoencoder as the dimensionality reduction technique.

1. Introduction

The novel coronavirus outbreak was reported by officials in December 2019 in Wuhan City, China. The virulence of COVID-19 has affected more than 200 million lives and killed more than four million people across the world [1]. The Reverse Transcription-Polymerase Chain Reaction (RT-PCR) test is treated as the golden standard for detection of the SARS-CoV-2 virus. The rapid increase in the number of patients, and the lack of sufficient RT-PCR test facilities in several parts of the world causes delay in testing and detection of the disease. Accessible, fast and affordable methods could play an important role in the diagnosis of the disease. Radiographic methods are easily available and affordable compared to RT-PCR test. A computer-aided diagnosis method using X-ray images could assist the medical practitioners in the detection of COVID-19 at an early stage.

Modalities for COVID-19 detection include Computed tomography

(CT), X-Ray and Ultrasound imaging. COVID-19 manifests common abnormal X-ray findings including ground glass and consolidative opacities in the peripheral lung regions, nodular opacities and bilateral patchy and confluent patterns. Some image findings include small amounts of pleural effusions, which are uncommon.Viral pneumonia exhibits patches in bilateral areas of consolidation, thickening of bronchial walls, bilateral consolidations, and ground glass opacities or centrilobular nodules poorly defined. Since COVID-19 and other pneumonia share some of these X-ray characteristics, it is not easy to differentiate the COVID-19 images from other pneumonia. Typical radiological COVID-19 pattern including bilateral peripheral or focal round ground glass opacities with or without consolidation differentiates COVID-19 from other pneumonia [2].

However, manual examination of the image modalities is time-consuming as the number of cases are increasing day by day. The application of machine learning in the biomedical field can assist

---

physicians in the computer-aided diagnosis of medical images efficiently and effectively. CAD systems aid radiologists to expound the medical images. Hence computer-aided detection could assist the radiologists in distinguishing COVID-19 infected radiographic images. CAD systems using Artificial Neural Networks (ANNs) and Deep Learning (DL) have shown tremendous success in the field of medical data analysis [3]. Deep learning technologies widely used in disease diagnosis include CNNs, autoencoder, Deep Belief Networks (DBN) and Generative Adversarial Network (GAN) [3]. Several works using ANNs and Deep Learning has been published in the application of disease diagnosis, including detection of interstitial lung disease [4], depression screening [5], schizophrenia [6], ECG arrhythmias classification [7], and ischemic heart disease [8].

Various works related to computer-aided detection of COVID-19 was published. Abraham et al. [9,10] developed models comprising an ensemble of CNNs to detect COVID-19. Ardakani et al. [11] discussed the usage of ten convolutional networks on COVID-19 detection. Horry et al. [12] highlight the use of different image modalities to help faster detection of the disease. Shaban et al. [13] adopted a new methodology for feature selection by integrating filter and wrapper methods and classifying using an ensemble learning technique. Phankokkruad et al. [14] developed a transfer learning technique that involves fine-tuning of the pre-trained network. Rekha Hanumanthu et al. [15] discussed different deep learning and transfer learning methods adopted for the early diagnosis of the disease. Wang et al. [16] adopted a method where the features are extracted using a UNet, and later, the classification was performed using a progressive classifier. Hassantabar et al. [17] used a convolutional neural network, where the Softmax layer helps detect SARS-COV2 infection. Rahimzadeh and Attar et al. [18] used transfer learning methodology, which involves fine-tuning a concatenation of Xception and Resnet50V2 for diagnosing COVID-19. Phankokkruad et al. [14] implemented a model involving transfer learning experimented on three different pre-trained networks such as VGG16, Xception and InceptionResnetV2. Ucar and Korkmax et al. [19] presented an Artificial Intelligence structure based on Squeezenet pre-trained network accompanied by Bayesian optimization. Li et al. [20] explored a multi-task contrastive learning for COVID-19 diagnosis. The contrastive learning task has been implemented using supervised neural networks. The method involves aggregation through contrastive loss. Pandit et al. [21] adopted fine-tuning of VGG-16 network for the diagnosis of the COVID-19 from chest radiographs. Chandra et al. [22] utilized an ensemble learning methodology for the detection of coronavirus. The methodology involves majority voting from different weak learners. Saufi et al. [23] used stacked sparse autoencoders to extract features from X-ray and CT scans for the detection of COVID-19. Lazrag et al. [24] explored wavelet analysis for feature extraction followed by autoencoder for feature modelling to detect COVID-19. Behura et al. [25] used XGBoost and sparse autoencoder for feature selection and classification. Ismael and Sengur et al. [26] developed a model involving classification using Support Vector Machine(SVM) prospecting the features extracted from X-ray using Resnet50 for the diagnosis of COVID-19. Toraman et al. [27] accomplished a methodology for discovering COVID-19 infections using capsule networks utilizing X-ray lung imaging.

Most of the existing works have used Convolutional Neural Networks for the detection of COVID-19. No methods have explored the combination of CNN with sparse autoencoder for the diagnosis of COVID-19. The proposed method chose to explore sparse autoencoder as a dimensionality reduction method. Sparse autoencoder has been found successful in the field of disease diagnosis, including Alzheimer's [28], Parkinson's disease [29], heart disease [30], identification of neonatal sleep state [31], glaucoma [32], cerebral microbleeds [33], etc to name a few. Sparse autoencoder enforces the sparsity constraint directing the single layer network for code learning resulting in error minimization while code reconstruction [34]. The sparsity penalty imposed on the hidden layers on top of the reconstruction error eliminates overfitting

[35]. Sparse representation of data has benefits in denoising robustness and improved classification performance in high dimensional latent spaces [36]. The proposed work has the following contributions.

● The method uses an ensemble of Xception and InceptionResnetV2 for feature extraction. The features are passed to a custom made sparse autoencoder for reducing dimensionality of feature vector, followed by a Feed Forward Neural Network (FFNN) for classification. No state-of-the-art methods have employed such a pipeline for the detection of COVID-19.
● The proposed method explores neural network techniques for all stages of computer-aided diagnosis of the disease, namely feature extraction, dimensionality reduction and classification. The technique utilizes neural networks, namely, CNN, Sparse autoencoder, and FFNN, for feature extraction, dimensionality reduction, and classification. Most of the existing methods have either performed transfer learning using CNN or used non-neural network methods for feature selection and classification combined with features extracted using CNN.

The method has chosen the ensemble of CNNs, sparse autoencoder and FFNN empirically based on experimental analysis. The experiments we performed prove the effectiveness of the novel framework composed of CNN, sparse autoencoder, and FFNN in diagnosing COVID-19. The rest of the paper is organized as follows. Section 2.1 discusses datasets used for training the model. Section 2.2 describes the architecture of the proposed model. Section 3 is a result analysis phase utilizing single-CNN and different combinations of pre-trained networks. The section also gives an overview of the comparison of the proposed model with other classifiers and dimensionality reduction techniques. Section 4 presents the conclusion of our work based on the result analysis phase.

2. Materials and methods

2.1. Dataset

Two publicly available datasets have been used to train the model. The first dataset is a public dataset by Cohen et al. [37], available in Github, consisting of both CT and Chest X-ray images of COVID-19 observations, other types of pneumonia and healthy patients. From this dataset X-ray images are filtered for training the model. The dataset consists of 783 X-ray images, among which 504 are COVID-19 images and 279 are non-COVID-19 images. The second dataset is a public open data set from Kaggle created by Paul Mooney [38]. The dataset consists of 390 chest X-ray images of bacterial and other viral pneumonia. It was constructed before the COVID-19 outbreak.

A balanced dataset is essential for building an effective model [39]. To balance the dataset for achieving an effective model, the first 263 X-ray images of Pneumonia affected patients have been extracted from the second dataset. The combined dataset consists of 504 COVID-19 images and 542 non-COVID-19 images. The non-COVID-19 images consist of both normal and pneumonia images.

2.2. Proposed method

The proposed architecture is divided into three phases. The model consists of feature extraction, dimensionality reduction and the classification phase.

2.2.1. Feature extraction

Feature extraction phase reduces the dimension of the initial raw dataset into manageable groups for optimizing the processing. Many of the recent research studies have worked on models based on pre-trained networks as a feature extractor [9]. In the proposed model, pre-trained networks are used as a feature extractor. Pre-trained networks are trained on an Imagenet database [40] consisting of 1000 image classes.

Even though trained on non-biomedical images, pre-trained CNNs in combination with off-the-shelf classifiers were found successful in the detection of a wide range of diseases from X-ray images, including tuberculosis [41], breast cancer [42] and pneumonia [43]. The convolutional layers built on top of each other, learn more complex features for reliable classification tasks. Automated feature extraction by CNN makes these networks highly efficient for classification tasks.

In the proposed model, images are pre-processed according to the input size in the input layer of the chosen pre-trained model, and then the dataset is fed into the network. Both single-CNN and multi-CNN has been utilized for the analysis. InceptionResnetV2 [44], Xception [45], EfficientnetB0 [46], Darknet-53 [47], Resnet101 [48] are used for the experimentation. The input size of different pre-trained networks differs in size. Table 1 denotes some of the pre-trained networks, their depth and their input size used for the analysis. The images are pre-processed to the respective input sizes of the pre-trained models before the feature extraction phase. The dataset used for the proposed model consists of 1046 instances.

The method has used CNN as a feature extractor and not as a transfer learning method, where parameters of an end-to-end pre-trained CNN are fine-tuned to suit the data at hand. While using pre-trained CNN as a feature extractor, activations from any of the deep layers except Softmax layer can be used as features for classification using an off-the-shelf classifier like FFNN. The layer from which features are to be extracted is a design choice [49,50]. However, selecting features from the fully connected layer right before the Softmax classification layer is a good option [49,50]. The activations of the last fully connected layer represent global feature representation of the image [51,52]. Another function of the last fully connected layer is dimension reduction [53]. Softmax layer output the vector of probability values of an input image belonging to one among the 1000 classes and hence it cannot be used as a feature extractor.

In the study by Abidin et al. [54], features extracted from the last fully-connected layer outperformed features from the other layers. Several research works have used the last fully-connected layer to extract features for classification using an off-the-shelf classifier [51, 55–64]. Based on the above-mentioned reasons, we have chosen activations of the last fully-connected layer as the feature vector. The output set after the feature extraction phase is a feature set with dimension 1046×1000.

The CNN includes three basic layers: convolution layer, pooling layer and a Softmax layer. The center of the convolutional neural network is the convolution layer. Convolutional operation is performed in this layer which is the linear multiplication of the filter mask and the input array image to produce a feature map. Consider $f(x, y)$ as the input image and $h(x, y)$ be the filter mask. Convolution operation can be mathematically expressed as:

$$g(x, y) = f(x, y) * h(x, y) \qquad (1)$$

An activation function is applied to the output of the convolution layer. The activation function used is ReLU. The next block after the convolution layer is the pooling layer. The most commonly used pooling layer is max-pooling layer. The pooling layer reduces the number of parameters used for the computation of the network. The final layer of the convolutional neural network is the classification layer, where the instances are classified according to the respective classes. For multi-

CNN the features are extracted from two CNN models and then concatenated to produce a new feature set. The dimension of the feature set is $n \times 1000 \, m$, where n is the number of X-ray images and m is the number of pre-trained CNN models.

2.2.2. Dimensionality reduction

Dimensionality reduction process reduces the dimension of feature vector by eliminating the features that will less contribute to the predictor variable. The presence of these irrelevant features may result in a decrease in the overall performance of the model. The proposed model uses sparse autoencoder for dimensionality reduction. Autoencoder is a network that imposes a bottleneck architecture, representing the input image in a compressed knowledge representation form [65]. The network follows an unsupervised learning technique for the task of representation learning [66].

The basic idea of the autoencoder is that it encodes the input sensor data using its hidden layer and outputs the best feature expression. The concept of autoencoder lies in taking in an unlabelled set and framing it as a supervised problem to get an output $\hat{x}$, which is a reconstruction of the original input x. The amount of information that traverses in the whole network is constrained in the bottleneck that drives a learned compression of the input image. Only the variations in the input data are maintained by the model for avoiding the redundancies. An autoencoder is composed of an encoder and a decoder. An encoder maps the input to a latent space using encoder activation. Later the input is reconstructed by the decoder using a decoder activation function. The activation of a basic autoencoder is represented as:

$$T_{AE} = E_x[l(x, g_d(g_e(x)))] \qquad (2)$$

where E_x denotes the pre-activation values, l is the squared loss function, g_e denotes the encoder activation function, and g_d indicates the decoder function. The cost function of an autoencoder is represented as:

$$C_x = \frac{\Sigma((L(x, \hat{x}))}{n} + \frac{\rho}{n} \Sigma w_{ij}^2 \qquad (3)$$

where $L(x, \hat{x})$, n, ρ and w_{ij} denotes the reconstruction error loss, number of training samples (X-ray images), weight decay parameter and the weight at $(i,j)^{th}$ location, respectively.

Different kinds of activation autoencoders are available, among which our proposed model uses sparse autoencoder. Sparse autoencoder constructs a loss function, and the network is allowed to learn encoding and decoding, which relies only on a small number of neurons. Rather than regularizing the activations, sparse autoencoders regularizes the weights of the network. Activations of different nodes of the neural network are data-dependent, as different inputs will activate different neurons.

Sparse autoencoders learn patterns by imposing sparsity constraints on the hidden layers [66]. The difference between the sparse autoencoder and the basic autoencoders lies in the cost function. Global regularization is used to solve the main objective function, whereas the sparsity penalty solves the trivial identity mapping and overfitting. In the model, features extracted using multiple pre-trained networks of the dimension $1046 \times 1000 \, m$, where m denotes the number of pre-trained networks is reduced to most relevant features of the dimension 1046×1000 using the sparse encoder for improving the performance of the model.

2.2.3. Classification

The model utilizes Feed Forward Neural Network for classifying the predictor variables. Feed Forward Neural Networks were found successful in a wide range of medical applications including Alzheimer's disease [67], chronic kidney disease [68] and lung cancer detection [69] to name a few. The wide usage of Feed Forward Neural Network in pattern classification is due to their prediction capability regardless of the probability distribution information of distinct labels. These

Table 1
Pre-trained networks.

Network	Depth	Input Size
InceptionResnetV2	164	299×299
Resnet101	101	224×224
Xception	72	299×299
EfficientnetB0	82	224×224
Darknet53	53	256×256

networks gain efficiency from their parallel structure and their ability to improve their performance by experience. Hence, they can be used to efficiently classify the observations into different classes [67]. They can store the information in the network with less fault tolerance capacity. Advanced developments have proved these networks as the function approximators as they can approximate any arbitrary functions by fine-tuning the number of hidden layers and their parameters [70].

The network includes connections, with each of the links designated to different weights. The information flow in the network exists only in one direction. The output of the previous layer serves as the input for its successive layer. The output of a neuron is represented as the weighted sum of its inputs. Equation (4) depicts the weighted sum of inputs of a layer.

$$\rho_p^k = \sum_{q=1}^{m_{k-1}} w_{qp}^k r_q^{k-1} \tag{4}$$

where ρ_p^k represents the weighted sum of inputs, m_{k-1} represents the number of nodes in $(k-1)^{th}$ layer, r_q^{k-1} represents the output of the pth node in $(k-1)^{th}$ layer and w_{pq} denotes the weight of the link. Equation (5) denotes the output of a layer.

$$r_p^k = h(\rho_p^k) \tag{5}$$

where h represents the output function and p ranges from 1 to m_k. Error loss is computed from the predicted and actual output by reforming the connection weights. Equation (6) depicts the error loss.

$$\delta = \frac{1}{2N} \sum_{p=1}^{N} (r_p - t_p)^2 \tag{6}$$

where δ represents the error loss, r_p and t_p are the predicted and actual outputs, respectively. Accounting the error loss, the connection weights are updated. The weights are updated such that the δ values are minimized. In each epoch of training, the weight reformation process happens for minimized error loss. Weight reformation takes place from the last layers proceeding towards the lower layers.

Fig. 1 shows the diagrammatic representation of the model. The set $f_1, f_2, ..., f_n$ denotes the feature set extracted by the pre-trained networks, $f_1', f_2', ..., f_n'$ denotes the features after reducing dimensionality and $h_1, h_2, h_3, h_4, ..., h_{10}$ denotes the 10 nodes in the hidden layer of FFNN. These features are fed as input to the Feed Forward Neural Network.

3. Results and discussions

3.1. Experimental setup

The experimentation was performed on Intel core i5 processor with GPU support of 4 GB and 8 GB RAM. The model has been implemented using MATLAB. Neural network toolbox by Jingwei Too [71] is used for implementing FFNN. Accuracy, F1-Score, Precision, Specificity, Sensitivity, Area Under Curve(AUC) and Matthews Correlation Coefficient (MCC) have been computed to evaluate the model.

3.2. Classification results

The experiments were performed on both single pre-trained networks and concatenation of multiple pre-trained networks. For single CNN, the features were extracted and passed to Feed Forward Neural Network. Ten-fold cross-validation has been performed for analysing the model. Random partitioning of data into 10 equal folds with 9 folds of data treated as training and the remaining 1 fold serves as the testing dataset at each iteration. A single iteration takes 90% of data for training and 10% for testing. Analysing the model using a single pre-trained network as the feature extractor and a Feed Forward Neural Network as an off-the-shelf classifier exhibits good performance. The pre-trained networks used for the analysis phase are EfficientnetB0, Resnet101, Darknet-53, InceptionResnetV2 and Xception. The last fully connected layers of the pre-trained networks have been used as the feature extractor, which outputs a feature set of dimension 1046 × 1000.

EfficientnetB0, Resnet101, Darknet-53, InceptionResnetV2 and Xception was able to achieve an accuracy of 0.9321, 0.9216, 0.86043, 0.9149 and 0.9081, respectively. It was noticed that EfficientnetB0 achieved the highest performance among the five networks used for training. The specificity, sensitivity, precision, F1-Score and AUC of EfficientnetB0 was 0.9452, 0.9188, 0.9305, 0.9425 and 0.9756, respectively. Out of 504 COVID radiographic images, 475 instances were correctly classified by the FFNN. Among 542 non-COVID-19 images, 500 instances were correctly classified.

Fig. 2 represents the graphical analysis of the model performance using single-CNN without using the dimensionality reduction module. Table 2 indicates the performance of the model using the Feed Forward Neural Network as the off-the-shelf classifier and without performing dimensionality reduction using sparse autoencoder. Using multi-CNN, the greatest performance was achieved using the combination of Xception and EfficientnetB0 with an accuracy of 0.9301. Out of 542 instances, 461 instances were predicted correctly. Fig. 3 depicts the
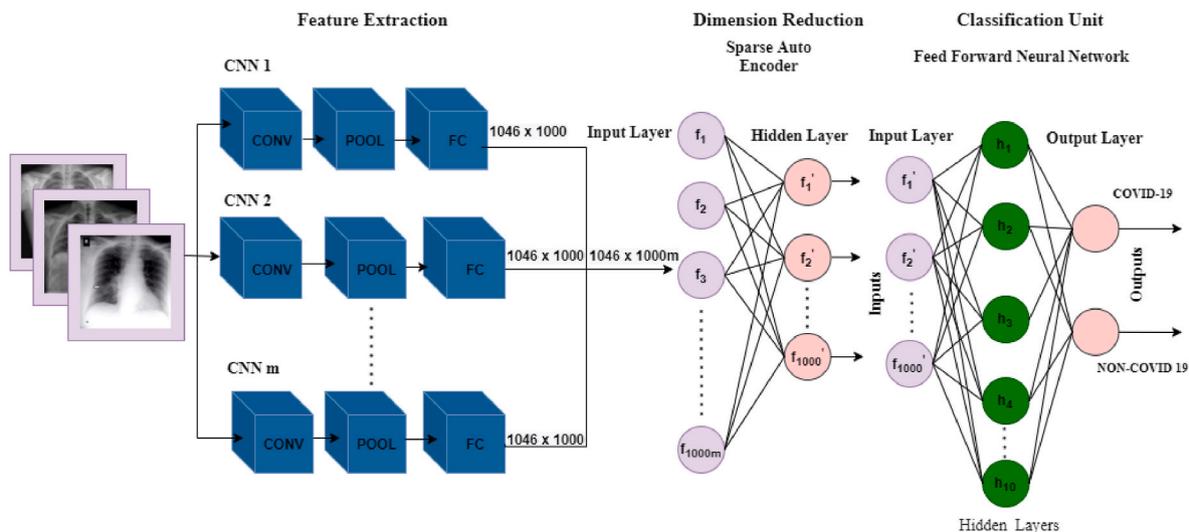
Fig. 1. Architecture of the proposed method.
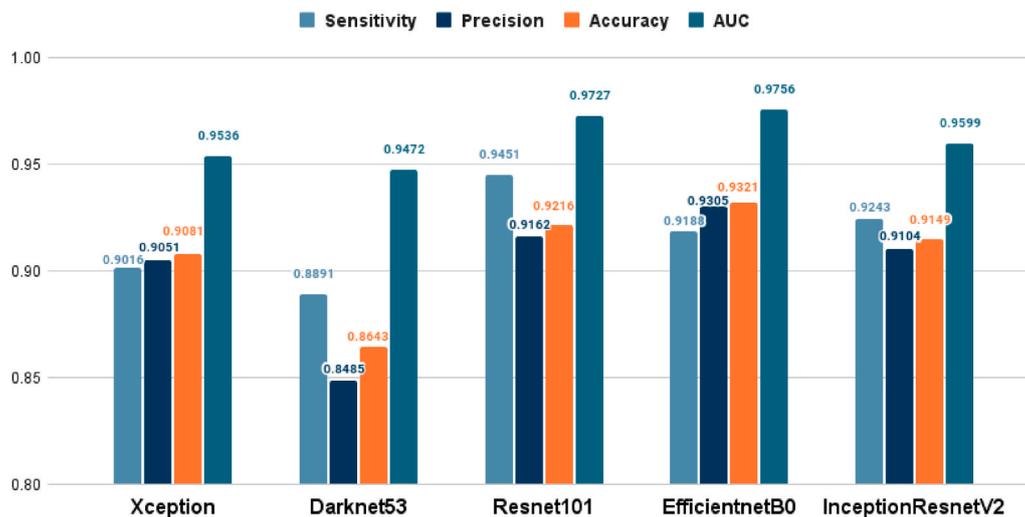
Fig. 2. Graphical analysis of model performance using Single-CNN without using sparse autoencoder.

Table 2

Performance of the model using CNN and FFNN, without using sparse autoencoder.

Pre-trained model	Specificity	Sensitivity	F1-Score	Precision	Accuracy	AUC	MCC
InceptionResnetV2+Xception	0.9206	0.9350	0.9237	0.9127	0.9273	0.9750	0.8546
InceptionResnetV2+Resnet101	0.9206	0.9350	0.9237	0.9127	0.9273	0.9750	0.8546
Xception + Resnet101	0.9271	0.9100	0.9163	0.9226	0.9187	0.9733	0.8374
Darknet53+Resnet101	0.9117	0.9458	0.9228	0.9008	0.9272	0.9564	0.8552
Xception + EfficientnetB0	0.9225	0.9389	0.9147	0.9266	0.9301	0.9787	0.8604
Xception	0.9145	0.9016	0.9087	0.9051	0.9081	0.9536	0.8163
Darknet53	0.8379	0.8891	0.8115	0.8485	0.8604	0.9472	0.7222
Resnet101	0.9021	0.9451	0.8889	0.9162	0.9216	0.9727	0.8441
EfficientnetB0	0.9452	0.9188	0.9425	0.9305	0.9321	0.9756	0.8645
InceptionResnetV2	0.9066	0.9243	0.8968	0.9104	0.9149	0.9599	0.8298
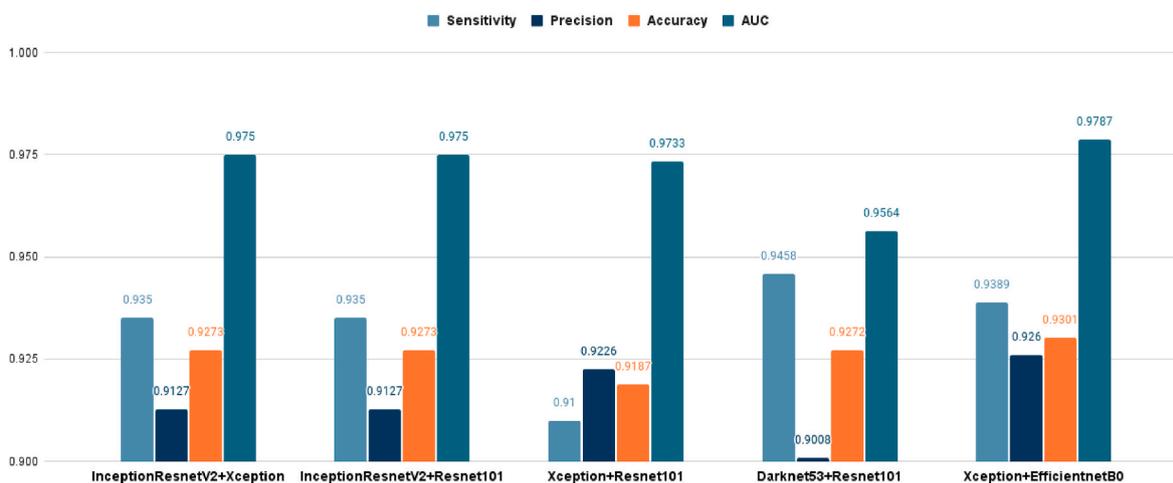
Fig. 3. Graphical analysis of model performance using Multi-CNN without using sparse autoencoder.

Table 3

Performance of the proposed model using Sparse Autoencoder and FFNN.

Pre-trained model	Specificity	Sensitivity	F1-Score	Precision	Accuracy	AUC	MCC
InceptionResnetV2+Xception	0.9594	0.9563	0.9563	0.9563	0.9578	0.9821	0.9158
Darknet53+Resnet101	0.9423	0.9613	0.9487	0.9365	0.9511	0.9800	0.9025
InceptionResnetV2+Resnet101	0.9227	0.9644	0.9378	0.9504	0.9417	0.9805	0.8842
Xception + Resnet101	0.9454	0.9537	0.9471	0.940 5	0.9493	0.9893	0.8986
Xception + EfficientnetB0	0.9436	0.9536	0.9460	0.9385	0.9483	0.9792	0.8967

graphical analysis of the multi-CNN model without using the dimensionality reduction module.

Dimensionality reduction using sparse autoencoder was performed in the second phase of analysis, which shows an improved result. Table 3 indicates the model's performance while using Sparse autoencoder and Feed Forward Neural Network. The best performance with an accuracy of 0.9578, F1-Score of 0.9563 and precision of 0.9563 was achieved with the combination of InceptionResnetV2 and Xception. The computation time taken by the model for testing 10% of the data is 276.87 s. The computation time taken by Darknet53+Resnet101, InceptionResnetV2+Resnet101, Xception+Resnet101, and Xception + EfficientnetB0 were 550.938 s, 397.66 s, 367.03 s and 312.18 s, respectively.

To ensure that the results are statistically significant, *p-value* based on chi-square has been computed for the best performing model of InceptionResnetV2 and Xception. A *p-value* less than 0.00001 is achieved that show that the results are statistically significant at p less than 0.05. Statistical model evaluation test, Matthews Correlation Coefficient (MCC) measure has also been analyzed for estimating model performance. MCC values close to one account for a strong correlation between predicted and the actual class. The highest MCC value of 0.9158 exhibited by the combination of Xception and InceptionResnetV2 denotes that the model is worth for distinguishing COVID-19 and non-COVID-19.

Xception networks denote extreme inception, with the inception architectures as the backbone of these networks. The convolutions in the original inceptions modules are restored with depthwise separable convolutions in Xception networks. This correlation scanning of 2D followed by 1D mapping is easier and more effective than full 3D mapping [72]. InceptionResnetV2, on the other hand, is an updated inceptionv3 network capable of better performance achievement than other convolutional networks. A combination of the inception block followed by the residual block in the architecture and the shortcut connections adds to the performance enhancement of the model. Thus the concatenation of these two efficient networks results in improved feature set generation and improved results.

The combination of InceptionResnetV2 and Resnet 101 achieved the highest sensitivity of 0.9644 and the combination of Xception and Resnet101 achieved the highest AUC of 0.9893. Even though the above-mentioned combinations achieved better sensitivity and AUC, the combination of InceptionResnetV2 and Xception outperformed the other combinations in all other performance measures. The experiments were repeated multiple times to ensure the stability of the results. As seed values were used to generate random weights and cross-validation folds, we could reproduce the same results each time.

The results show that the accuracy of the model has been improved with sparse encoder dimensionality reduction technique. However, no pre-trained CNNs were able to achieve 100% accuracy. Few X-ray images were misclassified in all the methods. The major cause of false positives and false negatives are the similarities in X-ray images of COVID-19 and pneumonia images, which makes the accurate prediction difficult. Among 504 COVID-19 instances, 482 images were correctly classified and the remaining 22 images were misclassified using the best performing combination of Xception and InceptionResnetV2 with sparse autoencoder and FFNN. Out of 542 non-COVID-19 images, 520 images were correctly classified, and 22 images were misclassified.

The statistical MCC test has proven that the model has improved its efficiency by incorporating sparse autoencoder technique. Fig. 4 represents the graphical analysis of the model after performing dimensionality reduction using sparse autoencoder. Fig. 5 presents the graphical analysis of the accuracy comparison, integrating dimension reduction and without dimension reduction methodology.

To compare the performance of deep pre-trained CNN with shallow network, we further performed feature extraction using a single hidden layer sparse autoencoder and classification using FFNN. Feature extraction using the shallow sparse autoencoder and classification using FFNN achieved accuracy, specificity, sensitivity, f1-score, precision and AUC of 0.8528, 0.8345, 0.8755, 0.8412, 0.8095 and 0.9217, respectively. The proposed method, which used deep pre-trained CNNs for feature extraction significantly outperformed feature extraction using shallow sparse autoencoder.

3.3. Parameter setting

For training the model using sparse autoencoder and FFNN, some of the parameters have been assigned. The parameters for the model construction has been empirically initialized using trial and error method. Table 4 denotes the parameter setting used for training the model. The hidden size parameter for sparse autoencoder specifies the number of features to be extracted. The model is trained for an epoch of 100.

3.4. Comparison of the results with other feature selection techniques

Sparse autoencoder has been empirically chosen for dimensionality reduction after performing experimental analysis with two major feature selection techniques, namely, Principal Component Analysis (PCA) and Correlation Feature Selection (CFS). Attribute selection using CFS and PCA has been performed on Weka 3.6. The selected features were then passed to feed forward neural network. Table 5 presents the results obtained while passing the feature extracted from InceptionResnetV2 and Xception to different feature selection techniques. CFS
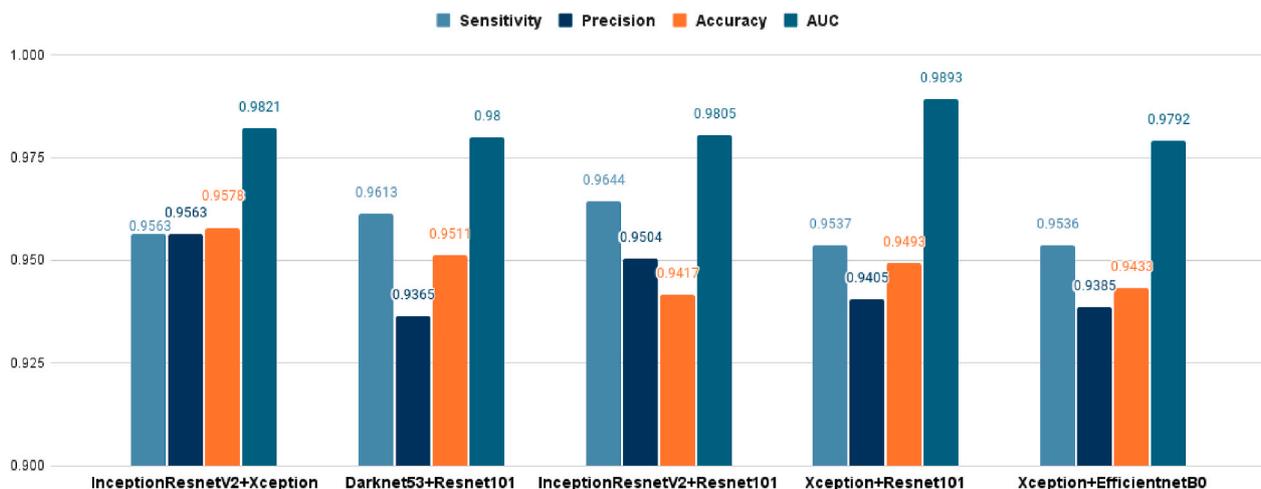
Fig. 4. Graphical analysis of Multi-CNN model performance after performing dimension reduction using sparse autoencoder.
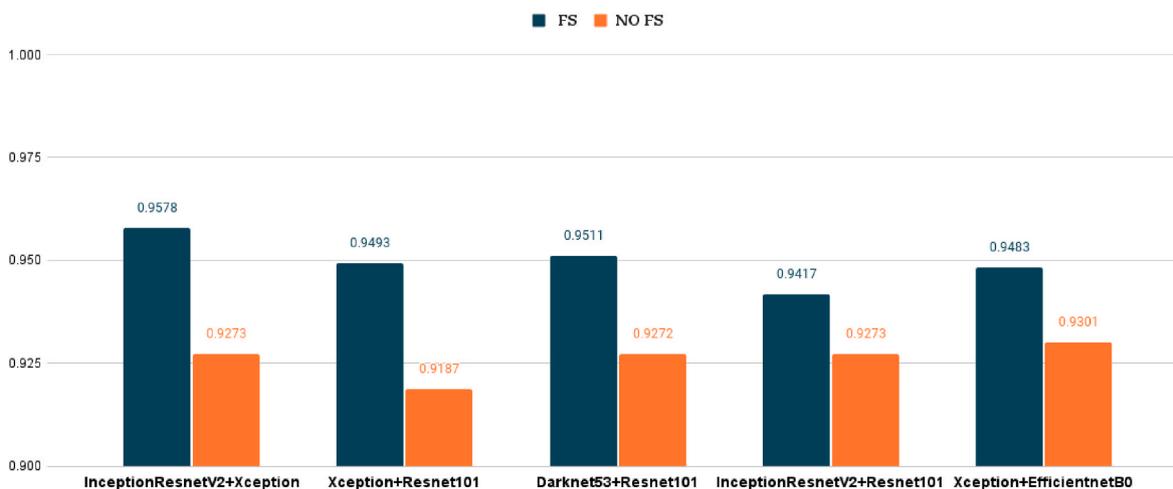
Fig. 5. Comparison of accuracy before and after employing sparse autoencoder as the dimensionality reduction technique.

Table 4
Parameter setting for Sparse autoencoder & FFNN.

Parameters(Sparse autoencoder)	Value
Hidden Size	1000
Random Seed	2
L2WeightRegularization	0.001
Sparsity Regularization	4.000
Sparsity Proportion	0.0500
Decoder Transfer Function	purelin
Parameters(FFNN)	Value
Hidden Layers	1
Random Seed	1
Maxepochs	100
K-Fold	10
trainFcn	trainlm
Net	feed forward net

and PCA were able to achieve an accuracy of 0.8556 and 0.8899, respectively. Among PCA and CFS, the proposed Sparse Autoencoder dimensionality reduction technique has proven its effectiveness with an accuracy of 0.9578. The Sparse Autoencoder has achieved an AUC of 0.9821, while CFS and PCA have acquired an AUC of 0.9326 and 0.9469, respectively. The time consumption for the sparse autoencoder, CFS and PCA were 190.308 s, 26 s and 260 s, respectively.

The superior performance of sparse autoencoder is attributed in the following. Sparse autoencoders learn the data projections more efficiently with the dimension and the sparsity constraints rather than the

other feature selection techniques. Autoencoder networks learn nonlinear transformations and are also more constructive in terms of model parameters with various layers than PCA with a single transformation [34].

3.5. Comparison of the results with other classifiers

FFNN has been empirically chosen as off-the-shelf classifier from experimental analysis with various other classifiers. Table 6 presents the performance of various classifiers with the best performing multi-CNN and sparse autoencoder. It is evident that only FFNN achieved an accuracy above 90%. Accuracy of Bayesnet, SVM, KNN, Random Forest and Adaboost falls in the range of 0.70–0.80. Among the different classifiers used for training, FFNN has achieved an accuracy of 0.9578 using InceptionResnetV2 and Xception as the backbone of the model and Sparse Autoencoder as the dimensionality reduction technique. The superior performance of FFNN owes to its ability to arrest more complex patterns. The time taken for the classification phase by the FFNN, Bayesnet, SVM, KNN, Random Forest and Adaboost were 0.128 s, 0.81 s, 2.17 s, 0.87 s, 2.1 s and 3.35 s, respectively.

3.6. Comparison with other state-of-the-art methods

Different state-of-the-art methods were analyzed for proving the effectiveness of the proposed model. Table 7 presents a consolidation of results achieved by other state-of-the-art methods and the proposed method. The comparison has considered only the methods using X-ray

Table 5
Performance achieved using various dimensionality reduction/feature selection techniques, in combination with the proposed pre-trained model and FFNN.

Pre-trained model	Method	Specificity	Sensitivity	F1-Score	Precision	Accuracy	AUC
InceptionResnetV2+Xception	Proposed	0.9594	0.9563	0.9563	0.9563	0.9578	0.9821
InceptionResnetV2+Xception	CFS	0.9031	0.8146	0.8582	0.9067	0.8556	0.9326
InceptionResnetV2+Xception	PCA	0.9228	0.8595	0.8900	0.9226	0.8899	0.9469

Table 6
Performance achieved using various classifiers, in combination with the proposed pre-trained model and sparse autoencoder.

Pre-trained model	Classifier	Specificity	Sensitivity	F1-Score	Precision	Accuracy	AUC
InceptionResnetV2+Xception	Bayesnet	0.8299	0.7038	0.7713	0.8532	0.7562	0.7970
InceptionResnetV2+Xception	SVM	0.7949	0.7899	0.7828	0.7758	0.7925	0.7920
InceptionResnetV2+Xception	KNN	0.7851	0.7930	0.7761	0.7599	0.7887	0.7850
InceptionResnetV2+Xception	Random Forest	0.8667	0.7450	0.8073	0.8810	0.7973	0.8790
InceptionResnetV2+Xception	Adaboost	0.8017	0.7079	0.7587	0.8175	0.7495	0.8300
InceptionResnetV2+Xception	Proposed	0.9594	0.9563	0.9563	0.9563	0.9578	0.9821

Table 7

Comparison of the results achieved by the proposed method with other state-of-the-art techniques.

Method	Number of images	Specificity	Sensitivity	F1-Score	Precision	Accuracy	AUC
Proposed	504 COVID-19 vs. 542 non-COVID-19	0.9594	0.9563	0.9563	0.9563	0.9578	0.9821
Abraham et al. [9]	453 COVID-19 vs. 497 non-COVID-19	-	0.9850	0.9140	0.8530	0.9115	0.9630
Li et al. [20]	231 COVID-19 vs. 1583 Normal	0.9191	0.9297	-	-	0.9723	0.9213
Pandit et al. [21]	224 COVID-19 vs. 1204 non-COVID-19	0.9727	0.9264	-	-	0.9600	-
Hassantabar et al. [17]	315 COVID-19 vs. 367 non-COVID-19	-	0.9610	-	-	0.9320	-
Chandra et al. [22]	696 COVID-19 vs. 696 non-COVID-19	-	-	-	-	0.9132	0.8310
Sethy et al. [73]	25 COVID-19 vs. 25 non-COVID-19	-	-	0.9141	-	0.9538	-
Waheed et al. [74]	403 COVID-19 vs. 721 Normal	0.9700	0.9000	-	0.9560	0.9500	-
Panwar et al. [75]	142 COVID-19 vs. 142 Normal	0.9700	0.9000	-	0.9560	0.8810	0.8809
Hemdan et al. [76]	25 COVID-19 vs. 25 non-COVID-19	-	1.0000	0.9100	0.8300	-	-
Zhang et al. [77]	100 COVID-19 vs. 1431 non-COVID-19	-	0.9600	-	-	-	0.9500
Ismael and Sengur et al. [26]	180 COVID-19 vs. 200 non-COVID-19	-	-	-	-	0.9470	-
Toraman et al. [27]	231 COVID-19 vs. 500 non-COVID-19	0.8095	0.9600	0.9375	0.9160	0.9124	-

images for the classification of COVID-19 and non-COVID-19 images. The methods employing CT scans and ultrasound for COVID-19 detection have not been considered, as they are entirely different modalities.

The number of instances used for training the model is different for multiple methods analyzed. The methods by Pandit et al. [21], Panwar et al. [75], Sethy et al. [73], Ismael and Sengur et al. [26] and Hemdan et al. [76] used a held-out validation set for the evaluation, whereas the other methods have used cross-validation. The results are on par with the state-of-the-art methods. Even though all the methods have achieved significant results, the proposed method achieved a better AUC, F1-score and precision than the other methods. Fair comparison between the different results analyzed are not possible since each method has utilized different number of images for the study.

3.7. Limitations and future research directions

Even though the method achieved significant results, some of the constraints are worth noting. The method is developed for binary classification of COVID-19 and non-COVID-19 images. The model has not been explored in a multi-class scenario in classifying normal, COVID-19 and pneumonia images. Also, a wide range of sparse autoencoder parameter values were not experimented in developing the model, which can be customized using the grid search method for relatively higher model performance. The method developed is specific to the diagnosis of COVID-19 from X-ray images. However, after empirical studies, it can be extended to diagnose other diseases. The method seems to have prospects in diagnosing other lung diseases and diseases that can be detected using X-rays. The strategy can also be applied to other imaging modalities, after customization. As a future research study, we propose applying the method for diagnosing COVID-19 from other imaging modalities such as CT and Ultrasound.

4. Conclusion

The proposed model implements a computer-aided model for COVID-19 detection utilizing chest X-ray images using Sparse Autoencoder and Feed Forward Neural Network. The concatenation of multiple pre-trained networks for feature extraction has been implemented which outperforms single-CNN. The usage of Sparse Autoencoder has greatly contributed in improving the accuracy of the model. It is worth noting that the performance of the model has considerably increased with the usage of the dimensionality reduction phase rather than using the Feed Forward Neural Network alone. From the analysis phase it is observed that combination of Xception and InceptionResnetV2 achieved greatest accuracy in combination with the custom-made sparse autoencoder and FFNN.

Declaration of conflict of interest

The authors declare that there are no conflicts of interest in this work.

References

[1] MAP COVID, Coronavirus Cases, Deaths, Vaccinations by Country, BBC News. Disponível em, 2021. https://www.bbc.com/news/world-51235105. (Accessed 21 September 2021).

[2] Liqa A. Rousan, Eyhab Elobeid, Musaab Karrar, Yousef Khader, Chest X-ray findings and temporal lung changes in patients with covid-19 pneumonia, BMC Pulm. Med. 20 (1) (2020) 1–9.

[3] Ryad Zemouri, Noureddine Zerhouni, Daniel Racoceanu, Deep learning in the biomedical applications: recent and future status, Appl. Sci. 9 (8) (2019) 1526.

[4] A. Dudhane, G. Shingadkar, P. Sanghavi, B. Jankharia, S. Talbar, Interstitial lung disease classification using feed forward neural networks, Adv. Intell. Syst. Res. ICCASP 137 (2017) 515–521.

[5] U Rajendra Acharya, Shu Lih Oh, Yuki Hagiwara, Jen Hong Tan, Hojjat Adeli, D Puthankattil Subha, Automated eeg-based screening of depression using deep convolutional neural network, Comput. Methods Progr. Biomed. 161 (2018) 103–113.

[6] Madiha J. Jafri, Vince D. Calhoun, Functional classification of schizophrenia using feed forward neural networks, in: 2006 International Conference of the IEEE Engineering in Medicine and Biology Society, IEEE, 2006, pp. 6631–6634.

[7] Giovanna Sannino, Giuseppe De Pietro, A deep learning approach for ecg-based heartbeat classification for arrhythmia detection, Future Generat. Comput. Syst. 86 (2018) 446–455.

[8] K. Rajeswari, V. Vaithiyanathan, T.R. Neelakantan, Feature selection in ischemic heart disease identification using feed forward neural networks, Procedia Eng. 41 (2012) 1818–1823.

[9] Bejoy Abraham, Madhu S. Nair, Computer-aided detection of covid-19 from x-ray images using multi-cnn and bayesnet classifier, Biocybern. Biomed. Eng. 40 (4) (2020) 1436–1445.

[10] Bejoy Abraham, Madhu S. Nair, Computer-aided detection of covid-19 from ct scans using an ensemble of cnns and ksvm classifier, Signal, Image Video Process. (2021) 1–8.

[11] Ali Abbasian Ardakani, Alireza Rajabzadeh Kanafi, U Rajendra Acharya, Nazanin Khadem, Afshin Mohammadi, Application of deep learning technique to manage covid-19 in routine clinical practice using CT images: results of 10 convolutional neural networks, Comput. Biol. Med. 121 (2020) 103795.

[12] Michael J. Horry, Subrata Chakraborty, Manoranjan Paul, Anwaar Ulhaq, Biswajeet Pradhan, Manas Saha, Nagesh Shukla, Covid-19 detection through transfer learning using multimodal imaging data, IEEE Access 8 (2020) 149808–149824.

[13] M Shaban Warda, Asmaa H. Rabie, Ahmed I. Saleh, M.A. Abo-Elsoud, A new covid-19 patients detection strategy (cpds) based on hybrid feature selection and enhanced knn classifier, Knowl. Base Syst. 205 (2020) 106270.

[14] Manop Phankokkruad, Covid-19 pneumonia detection in chest x-ray images using transfer learning of convolutional neural networks, in: Proceedings of the 3rd International Conference on Data Science and Information Technology, 2020, pp. 147–152.

[15] Swapna Rekha Hanumanthu, Role of intelligent computing in Covid-19 prognosis: a state-of-the-art review, Chaos, Solit. Fractals (2020), 109947.

[16] Xinggang Wang, Xianbo Deng, Qing Fu, Qiang Zhou, Jiapei Feng, Hui Ma, Wenyu Liu, Chuansheng Zheng, A weakly-supervised framework for covid-19 classification and lesion localization from chest ct, IEEE Trans. Med. Imag. 39 (8) (2020) 2615–2625.

[17] Shayan Hassantabar, Mohsen Ahmadi, Abbas Sharifi, Diagnosis and detection of infected tissue of covid-19 patients based on lung x-ray image using convolutional neural network approaches, Chaos, Solit. Fractals 140 (2020) 110170.

[18] Mohammad Rahimzadeh, Abolfazl Attar, 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, Inform. Med. Unlocked 19 (2020) 100360.

[19] Ferhat Ucar, Deniz Korkmaz, Covid diagnosis-net: deep bayes-squeezenet based diagnosis of the coronavirus disease 2019 (covid-19) from x-ray images, Med. Hypotheses 140 (2020) 109761.

[20] Jinpeng Li, Gangming Zhao, Yaling Tao, Penghua Zhai, Hao Chen, Huiguang He, Ting Cai, Multi-task contrastive learning for automatic CT and X-ray diagnosis of COVID-19, Pattern Recogn. 114 (2021) 107848.

[21] Mohammad Khalid Pandit, Shoaib Amin Banday, Roohie Naaz, Mohammad Ahsan Chishti, Automatic detection of covid-19 from chest radiographs using deep learning, Radiography 27 (2) (2021) 483–489.

[22] Tej Bahadur Chandra, Kesari Verma, Bikesh Kumar Singh, Deepak Jain, Satyabhuwan Singh Netam, Coronavirus disease (covid-19) detection in chest x-ray images using majority voting based classifier ensemble, Expert Syst. Appl. 165 (2021) 113909.

[23] Syahril Ramadhan Saufi, Muhd Danial Abu Hasan, Zair Asrar Ahmad, Mohd Salman Leong, Lim Meng Hee, Detection of covid-19 from chest x-ray and ct scan images using improved stacked sparse autoencoder, Pertanika J. Sci. Technol. 29 (3) (2021).

[24] Houda Lazrag, Ramzi Ben Ali, Ridha Ejbali, Sparse wavelet auto-encoder for covid-19 cases identification, in: International Conference on Hybrid Intelligent Systems, Springer, 2020, pp. 22–29.

[25] Aradhana Behura, A deep learning application for prediction of covid-19, in: Impact of AI and Data Science in Response to Coronavirus Pandemic, Springer, 2021, pp. 127–148.

[26] Aras M. Ismael, Abdulkadir Şengür, Deep learning approaches for covid-19 detection based on chest x-ray images, Expert Syst. Appl. 164 (2021) 114054.

[27] Suat Toraman, Talha Burak Alakus, Ibrahim Turkoglu, Convolutional capsnet: a novel artificial neural network approach to detect covid-19 disease from x-ray images using capsule networks, Chaos, Solit. Fractals 140 (2020) 110122.

[28] Pushkar Bhatkoti, Manoranjan Paul, Early diagnosis of alzheimer's disease: a multi-class deep learning framework with modified k-sparse autoencoder classification, in: 2016 International Conference on Image and Vision Computing New Zealand (IVCNZ), IEEE, 2016, pp. 1–5.

[29] Shiqi Li, Haijun Lei, Feng Zhou, Jamal Gardezi, Baiying Lei, Longitudinal and multi-modal data learning for Parkinson's disease diagnosis via stacked sparse auto-encoder, in: 2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019), IEEE, 2019, pp. 384–387.

[30] Ibomoiye Domor Mienye, Yanxia Sun, Zenghui Wang, Improved sparse autoencoder based artificial neural network approach for prediction of heart disease, Inform. Med. Unlocked 18 (2020) 100307.

[31] Luay Fraiwan, Khaldon Lweesy, Neonatal sleep state identification using deep learning autoencoders, in: 2017 IEEE 13th International Colloquium on Signal Processing & its Applications (CSPA), IEEE, 2017, pp. 228–231.

[32] U. Raghavendra, Anjan Gudigar, Sulatha V. Bhandary, Tejaswi N. Rao, Edward J. Ciaccio, U Rajendra Acharya, A two layer sparse autoencoder for glaucoma identification with fundus images, J. Med. Syst. 43 (9) (2019) 1–9.

[33] Yu-Dong Zhang, Yin Zhang, Xiao-Xia Hou, Hong Chen, Shui-Hua Wang, Seven-layer deep neural network based on sparse autoencoder for voxelwise detection of cerebral microbleed, Multimed. Tool. Appl. 77 (9) (2018) 10521–10538.

[34] V. Jonnalagadda, Sparse, Stacked and Variational Autoencoder, Medium, 2018.

[35] Abhinav Prakash, Different Types of Autoencoders, https://iq.opengenus.org/types-of-autoencoder/ (Retrieved on 11 November 2021).

[36] Rui Shi, Jian Ji, Chunhui Zhang, Qiguang Miao, Boosting sparsity-induced autoencoder: a novel sparse feature ensemble learning for image classification, Int. J. Adv. Rob. Syst. 16 (3) (2019), 1729881419853471.

[37] Joseph Paul Cohen, Paul Morrison, Dao Lan, Covid-19 Image Data Collection, 2020 arXiv 2003.11597, https://github.com/IEEE8023/covid-chestxray-dataset. (Accessed 14 June 2021).

[38] P. Mooney, Pneumonia X Rays, 2018. https://www.kaggle.com/paultimothymooney/chest-xray-pneumonia. (Accessed 14 June 2021).

[39] M Mostafizur Rahman, Darryl N. Davis, Addressing the class imbalance problem in medical datasets, Int. J. Mach. Learn. Comput. 3 (2) (2013) 224.

[40] Deng Jia, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, Li Fei-Fei, Imagenet: a large-scale hierarchical image database, in: 2009 IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2009, pp. 248–255.

[41] UK Lopes, João Francisco Valiati, Pre-trained convolutional neural networks as feature extractors for tuberculosis detection, Comput. Biol. Med. 89 (2017) 135–143.

[42] Aditi V. Vedalankar, Shankar S. Gupta, Ramchandra R. Manthalkar, Addressing architectural distortion in mammogram using alexnet and support vector machine, Inform. Med. Unlocked 23 (2021) 100551.

[43] Ali Mohammad Alqudah, Shoroq Qazan, Ihssan S. Masad, Artificial intelligence framework for efficient detection and classification of pneumonia using chest radiography images, J. Med. Biol. Eng. (2021) 1–11.

[44] Christian Szegedy, Sergey Ioffe, Vanhoucke Vincent, Alex Alemi, Inception-v4, Inception-Resnet and the Impact of Residual Connections on Learning, 2016 arXiv preprint arXiv:1602.07261.

[45] François Chollet, Xception: deep learning with depthwise separable convolutions, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 1251–1258.

[46] Mingxing Tan, V Le Quoc, Efficient net: Rethinking Model Scaling for Convolutional Neural Networks, 2019 arXiv preprint arXiv:1905.11946.

[47] Redmon Joseph, Farhadi Ali, Yolov3: an Incremental Improvement, 2018 arXiv preprint arXiv:1804.02767.

[48] Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.

[49] Image category classification using deep learning. https://in.mathworks.com/help/vision/ug/image-category-classification-using-deep-learning.html, 2020, retrieved on 11.11.2021.

[50] C. Perez, Big Data and Deep Learning. Examples with Matlab, Lulu Press, Inc, 2020.

[51] Samet Akcay, Mikolaj E. Kundegorski, Chris G. Willcocks, Toby P. Breckon, Using deep convolutional neural network architectures for object classification and detection within x-ray baggage security imagery, IEEE Trans. Inf. Forensics Secur. 13 (9) (2018) 2203–2215.

[52] Hu Fan, Gui-Song Xia, Jingwen Hu, Liangpei Zhang, Transferring deep convolutional neural networks for the scene classification of high-resolution remote sensing imagery, Rem. Sens. 7 (11) (2015) 14680–14707.

[53] Arun Kumar Sangaiah, Deep Learning and Parallel Computing Environment for Bioengineering Systems, Academic Press, 2019.

[54] Anas Z. Abidin, Botao Deng, Adora M. DSouza, Mahesh B. Nagarajan, Paola Coan, Axel Wismüller, Deep transfer learning for characterizing chondrocyte patterns in phase contrast x-ray computed tomography images of the human patellar cartilage, Comput. Biol. Med. 95 (2018) 24–33.

[55] Muhammad Awais, Xi Long, Bin Yin, Chen Chen, Saeed Akbarzadeh, Saadullah Farooq Abbasi, Muhammad Irfan, Chunmei Lu, Xinhua Wang, Laishuan Wang, et al., Can pre-trained convolutional neural networks be directly used as a feature extractor for video-based neonatal sleep and wake classification? BMC Res. Notes 13 (1) (2020) 1–6.

[56] Amirreza Mahbod, Gerald Schaefer, Chunliang Wang, Rupert Ecker, Isabella Ellinge, Skin lesion classification using hybrid deep neural networks, in: ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2019, pp. 1229–1233.

[57] Sheng Guo, Weilin Huang, Limin Wang, Yu Qiao, Locally supervised deep hybrid model for scene recognition, IEEE Trans. Image Process. 26 (2) (2016) 808–820.

[58] B Hegde Roopa, Keerthana Prasad, Harishchandra Hebbar, Brij Mohan Kumar Singh, Feature extraction using traditional image processing and convolutional neural network methods to classify white blood cells: a study, Australas. Phys. Eng. Sci. Med. 42 (2) (2019) 627–638.

[59] M. Toğaçar, B. Ergen, Z. Cömert, F. Özyurt, A deep feature learning model for pneumonia detection applying a combination of mrmr feature selection and machine learning models, Irbm 41 (4) (2020) 212–222.

[60] Mesut Toğaçar, Burhan Ergen, Zafer Cömert, Waste classification using autoencoder network with integrated feature selection method in convolutional neural network models, Measurement 153 (2020), 107459.

[61] Zhe Zhu, Ehab Albadawy, Ashirbani Saha, Jun Zhang, Michael R. Harowicz, Maciej A. Mazurowski, Deep learning for identifying radiogenomic associations in breast cancer, Comput. Biol. Med. 109 (2019) 85–90.

[62] Na Liu, Lihong Wan, Yu Zhang, Tao Zhou, Hong Huo, Tao Fang, Exploiting convolutional neural networks with deeply local description for remote sensing image classification, IEEE access 6 (2018) 11215–11228.

[63] Sivaramakrishnan Rajaraman, Sameer K. Antani, Mahdieh Poostchi, Kamolrat Silamut, Md A. Hossain, Richard J. Maude, Stefan Jaeger, George R. Thoma, Pre-trained convolutional neural networks as feature extractors toward improved malaria parasite detection in thin blood smear images, PeerJ 6 (2018), e4568.

[64] Fatih Demir, Daban Abdulsalam Abdullah, Abdulkadir Sengur, A new deep cnn model for environmental sound classification, IEEE Access 8 (2020) 66529–66537.

[65] Andrew Ng, et al., Sparse autoencoder, CS294A Lect. Notes 72 (2011) 1–19, 2011.

[66] Bruno A. Olshausen, David J. Field, Sparse coding with an over complete basis set: a strategy employed by v1? Vis. Res. 37 (23) (1997) 3311–3325.

[67] Debesh Jha, Ji-In Kim, Goo-Rak Kwon, Diagnosis of alzheimer's disease using dual-tree complex wavelet transform, pca, and feed-forward neural network, J. Healthcare eng, Hindawi 2017 (2017).

[68] Njoud Abdullah Almansour, Hajra Fahim Syed, Nuha Radwan Khayat, Rawan Kanaan Altheeb, Renad Emad Juri, Jamal Alhiyafi, Saleh Alrashed, Sunday O. Olatunji, Neural network and support vector machine for the prediction of chronic kidney disease: a comparative study, Comput. Biol. Med. 109 (2019) 101–111.

[69] R. Arulmurugan, H. Anandakumar, Early detection of lung cancer using wavelet feature descriptor and feed forward back propagation neural networks classifier, in: Computational Vision and Bio Inspired Computing, Springer, 2018, pp. 103–110.

[70] Foued Saâdaoui, A seasonal feedforward neural network to forecast electricity prices, Neural Comput. Appl. 28 (4) (2017) 835–847.

[71] Jingwei Too, Neural Network Toolbox, 2020. https://in.mathworks.com/matlabcentral/fileexchange/71468-neural-network-toolbox?s_tid=prof_contriblnk. (Accessed 14 June 2021).

[72] Joyce Xu, An Intuitive Guide to Deep Network Architectures, Medium, 2017.

[73] Prabira Kumar Sethy, Santi Kumari Behera, Pradyumna Kumar Ratha, Preesat Biswas, Detection of Coronavirus Disease (COVID-19) Based on Deep Features and Support Vector Machine, 2020 preprints202003.0300.v1.

[74] Abdul Waheed, Muskan Goyal, Deepak Gupta, Ashish Khanna, Fadi Al-Turjman, Plácido Rogerio Pinheiro, Covid gan: data augmentation using auxiliary classifier GAN for improved covid-19 detection, IEEE Access 8 (2020) 91916–91923.

[75] Harsh Panwar, P.K. Gupta, Mohammad Khubeb Siddiqui, Ruben Morales-Menendez, Vaishnavi Singh, Application of deep learning for fast detection of covid-19 in X-rays using ncovnet, Chaos, Solit. Fractals 138 (2020) 109944.

[76] Ezz El-Din Hemdan, Marwa A. Shouman, Mohamed Esmail Karar, Covidx-net: A Framework of Deep Learning Classifiers to Diagnose Covid-19 in X-Ray Images, 2020 arXiv preprint arXiv:2003.11055.

[77] Jianpeng Zhang, Yutong Xie, Yi Li, Chunhua Shen, Yong Xia, Covid-19 Screening on Chest X-Ray Images Using Deep Learning Based Anomaly Detection vol. 27, 2020 arXiv preprint arXiv:2003.12338.