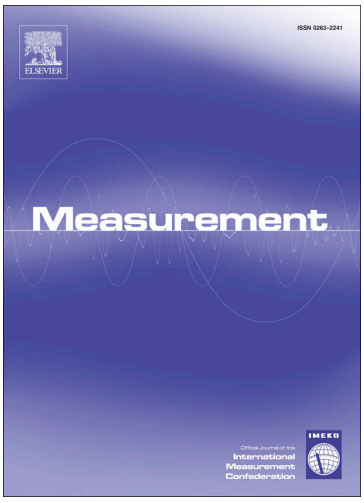
Journal Pre-proofs



Pneumonia Detection in chest X-ray images using Convolutional Neural Networks and Transfer Learning

Rachna Jain, Preeti Nagrath, Gaurav Kataria, V. Sirish Kaushik, D. Jude Hemanth

PII: S0263-2241(20)30584-4
DOI: https://doi.org/10.1016/j.measurement.2020.108046
Reference: MEASUR 108046

To appear in: *Measurement*

Received Date: 13 November 2019
Revised Date: 28 April 2020
Accepted Date: 21 May 2020

Please cite this article as: R. Jain, P. Nagrath, G. Kataria, V. Sirish Kaushik, D. Jude Hemanth, Pneumonia Detection in chest X-ray images using Convolutional Neural Networks and Transfer Learning, *Measurement* (2020), doi: https://doi.org/10.1016/j.measurement.2020.108046

Pneumonia Detection in chest X-ray images using Convolutional Neural Networks and Transfer Learning

Rachna Jain[1], Preeti Nagrath[1], Gaurav Kataria[1], V. Sirish Kaushik[1] and D. Jude Hemanth[2]

[1]Department of CSE, Bharati Vidyapeeth's College of Engineering, New Delhi, India
[2]Department of ECE, Karunya Institute of Technology and Sciences, Coimbatore, India
*Corresponding author email: judehemanth@karunya.edu

Abstract: A large number of children die due to pneumonia every year worldwide. An estimated 1.2 million episodes of pneumonia were reported in children up to 5 years of age, of which 880,000 died in 2016. Hence, pneumonia is a major cause of death amongst children, with high prevalence rate in South Asia and Sub-Saharan Africa. Even in a developed country like the United States, pneumonia is among the top 10 causes of deaths. Early detection and treatment of pneumonia can reduce mortality rates among children significantly in countries having a high prevalence. Hence, this paper presents Convolutional Neural Network models to detect pneumonia using x-ray images. Several Convolutional Neural Networks were trained to classify x-ray images into two classes viz., pneumonia and non-pneumonia, by changing various parameters, hyperparameters and number of convolutional layers. Six models have been mentioned in the paper. First and second models consist of two and three convolutional layers, respectively. The other four models are pre-trained models, which are VGG16, VGG19, ResNet50, and Inception-v3. The first and second models achieve a validation accuracy of 85.26% and 92.31% respectively. The accuracy of VGG16, VGG19, ResNet50 and Inception-v3 are 87.28%, 88.46%, 77.56% and 70.99% respectively.

Keywords: Convolutional Neural Networks, Pneumonia detection, VGG Net, ResNet, and Inception-v3.

1. Introduction

Convolutional Neural Networks (CNNs) are inspired from the visual cortex of the brain and are used to solve difficult image-driven pattern recognition tasks, recognizing linear and non-linear patterns (Cireşan, D. C. et al.). CNN is good for image classification as less number of parameters and connections are required in such networks. This makes the training of such neural networks (CNNs) far easier compared to other neural networks. Artificial Neural Networks, on the other hand, have difficulty in computing image data in view of a high degree of computational complexity involved (O'Shea, K. et al.). Six models are presented in this paper to detect pneumonia in x-ray images, which will help control this deadly infection in children and other age groups. Pneumonia is caused by bacteria, viruses, or fungi in the air we breathe. The patient with pneumonia, have serious inflammation in lung's air sacs, which get filled with fluid or pus, this makes it extremely difficult to breathe. Pneumonia can be mild or life-threatening. Hence, timely detection and treatment of pneumonia are critical for controlling high mortality rates among children due to this infectious disease in developing and developed countries (Ypsilantis, P. P. et al.). Six neural network models presented in this paper, diagnose pneumonia using x-ray images of the patients (Rajpurkar et al.).

Accuracy of the model is directly correlated with the size of the dataset that is, use of large datasets help improve the accuracy of the model, but there is no direct correlation between the depth of the model and the accuracy of the model.

As humans apply their previously gained knowledge to understand and solve new tasks, in a similar way, neural networks are trained and tested on different datasets. The knowledge acquired by the network can then be applied to train and test new datasets. This process is known as Transfer Learning. In transfer learning, the neural network utilizes previously gained knowledge to solve newer tasks (Wang, J. et al.). In neural networks, previously gained knowledge would be weights and features. These networks save weights and features from before and then use these to achieve high performance on the target task (Mao, W. et al.). Transfer learning models which are available on Keras are Xception, VGG16, VGG19, ResNet and its variants, Inception-v3 and its variants, MobileNet, DenseNet, and NASNet. Transfer learning models used in this paper are VGG16, VGG19, ResNet50, and Inception-v3 (Szegedy, C. et al.). All of these models have been trained and tested on the ImageNet dataset. This dataset contains about 15 million images belonging to 22,000 different categories and is the largest dataset for image classification.

VGG16, VGG19, ResNet50, and Inception-v3 are transfer learning models consisting of many layers (Simonyan, K. et al.). One of the biggest problems faced when developing deep networks is vanishing gradient. In vanishing gradient problem, during back propagation, the gradients become infinitesimally small, leading to loss of integral information. Due to this, the accuracy of the network saturates and then starts degrading. Different techniques have been employed by the models used in this paper to overcome the vanishing gradient problem. Training deep networks have certain restrictions viz., the dataset should be large, a large number of computational resources are used to achieve high performance, and the process of fine-tuning each parameter and hyper-parameter to achieve the optimum results is quite mundane.

The research paper is organized into six sections: Section 1 introduces the topic and explains its importance. Section 2 explores work related to our model. Section 3 explains the methodology of the paper, explaining the architecture of a basic CNN model and the specific models presented in this paper. This section also explains the dataset used to train and test the six models. Section 4 showcases the results achieved by each model, and Section 5 concludes the research paper. References are listed in Section 6.

2. Related work

The problem of classifying chest x-ray images into different classes has been significantly explored in the field of medical diagnosis. Many research papers have been published, tackling this problem. Rajpurkar et al. trained a deep learning model to detect pneumonia in chest x-ray images on the dataset ChestX-ray14. Using ChXNet, which is a 121 layer CNN they classified chest x-ray images at a level exceeding practicing radiologists. Apart from detecting pneumonia, their model also detected 14 other diseases. They compared the performance of their model with practicing academic radiologists. Their model provides a state of the art performance and hopes to improve the delivery of healthcare. Guan Q. et al. developed an AG-CNN model approach to detect thorax disease from chest x-ray images. This research has been conducted on Chest X-ray 14 dataset. The classification was done using two branch attention guided CNN. The two branches being global and local pick up global and local cues to predict thorax disease. Heat maps are also used to train the CNN model. They compared their model's performance with other models. Their approach outperformed various other models, having an average AUC of 0.871.

Xu Y. et al. trained a CNN model for classification and segmentation of brain tumor images of large dimensions. This model uses data augmentation, feature selection, and feature pooling techniques. The accuracy of segmentation and classification of this model are 84% and 97.5% respectively. They presented their approach in MICCAI 2014 Brain Tumor Digital Pathology Challenge. Rubin et al. presented a dual CNN which performs large scale automatic recognition of front and lateral images of chest x-ray on MIMIC-CXR dataset, which is the largest available dataset of chest X-rays till date. This neural network is used to detect common thorax disease. The dataset was divided into training data, testing data, and validation data. 70% of the data was used for training, 20% was used for testing, and 10% for validation. Their model has an average AUC of 0.721 and 0.668 for PA and AP, respectively. They aim to improve their model's performance by using data augmentation and pixel normalization techniques to provide aid to the workflow of the process to identify common thorax disease.

Lakhani P. et al. trained a deep CNN for automated classification of pulmonary tuberculosis from chest radiographs. AlexNet and GoogLeNet, which are dual CNNs, were used for classification purposes. The dataset was pre-processed before evaluation. Their model had an astounding AUC of 0.99. Their model had a specificity 100% and sensitivity of 97.3%. CNN is used to detect and classify abnormalities in frontal chest radiographs using deep convolutional neural networks was trained by Cicero M. et al. The input images were of the size 256X256 pixels. The AUC of the model is 0.964 with an average specificity and sensitivity of 91% showing that deep convolutional neural networks can be developed with high classification accuracy and can help in the diagnosis procedure.

Anthimopoulos M. et al. presented a CNN model to identify interstitial lung disease patterns. Their model consists of 5 convolutional layers, employing leaky ReLU activation function, average pooling layers, and three dense layers. The dataset on which it was trained contains seven classes, and the dataset has 14696 images. Their model had an accuracy of 85.5%. They hope to extend their model to classify 3D images to be a supportive tool for diagnostic purposes. Glozman T. et al. presented a transfer model, which is an extension to AlexNet to classify Alzheimer's disease on the ADNI database. Data augmentation technique was employed to enhance the performance of the deep neural network.

Cho Y. et al. presented an ISC method which is based on incremental learning. They used a dataset which comprised of cortical thickness data. Their model achieved a specificity of 93% on the classification of AD patients from HC subjects. Hemanth D. J. et al. dealt with the problem of the high convergence time period for ANNs. They presented two models, which are MCPN and MKNN, which classified MR images iteration free with high accuracy. They used sensitivity and specificity as performance measures for their models. Three new deep CNN models were presented by Szegedy C. et al. which are variants of the combination of Inception and ResNet models. Their model showed promising results. They achieved 3.08% top 5 error on the testing dataset of ImageNet classification challenge.

The ability of deep CNN models to achieve groundbreaking results on complex datasets was shown by Krizhevsky A. et al. achieving a top 5 error percent of 17%. The dataset used was the ImageNet dataset. Dropout increased the efficiency of the model considerably. Their network contains 60 million total number of parameters and has five convolutional layers and max-pooling layers. Three fully connected layers were used to provide optimum results. State of the art deep CNN model, which was submitted to ILSVRC 2014 developed by Simonyan K. et al. which was also used in this paper achieved a 92.7% top-5 test accuracy on the ImageNet dataset. Their model has multiple variants, being widely used for classification purposes in medical research. This model was the first model to introduce small kernel sized filters one after the other instead of using one large kernel sized filter. He K. et al. presented the approach of residual learning for classification purposes. This model introduces shortcut connections to improve

performance. The dataset used for training and testing was the ImageNet dataset. This model was submitted to ILSVRC 2015.

Jaiswal, A. et al. presented a Mask-RCNN based identification model for pixel-wise segmentation incorporating global and local features. They introduced critical alterations in the training process merging bounding boxes from multiple models. The performances evaluated on chest radiograph dataset which depict potential pneumonia causes. The quality of images is an imperative factor in diagnosis of disease. Elhoseny, M. et al proposed an optimal bilateral filter to remove noise from the medical images. A detailed review is presented by Chandra, T et al. analyzing the filters to reduce quantum noise in chest x-ray images.

3. Methodology

The paper involves multiple steps, starting with the dataset being imported from Kaggle. The dataset was pre-processed. Thereafter, the dataset was divided into train and test sets, which consisted of 5216 and 624 images, respectively. The six models were trained on the training dataset, each with different architectures (Szegedy, C. et al.). Each model was trained for 20 epochs, with training & testing batch sizes of 32 and one respectively. After training and testing, the validation accuracy of models 1, 2, VGG16, VGG19, ResNet50, and Inception-v3 were calculated. The following sub-headings further explain the above stages in depth. The different stages of the work is depicted in figure 1 with the help of a flow chart.

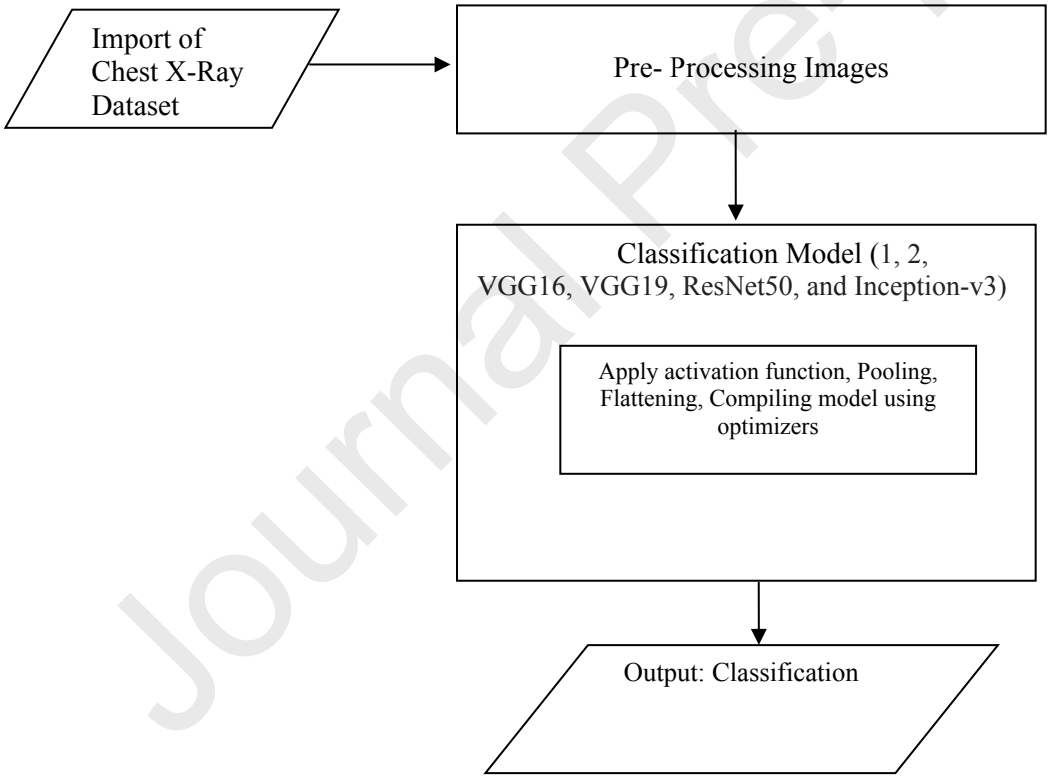


Figure 2 - Flowchart of stages of proposed work

Algorithm

Step 1: Pass 64x64 images through convolutional layer (32 feature maps, ReLU activation function)

$$y = \max(0, x) \quad (1)$$

Step 2: Output of previous layer passed through a 2D max pooling layer of dimensions 2x2
Step 3: Input image size is set to 64x64 and passed through a convolutional layer (64 feature maps, ReLU activation function)
Step 4: Output of previous layer passed through a 2D max pooling layer of dimensions 2x2.
Step 5:

a. For model 1, output is directly flattened
b. For model 2, the input image size is set to 64x64 and passed through convolutional layer (128 feature maps,ReLU activation function) output of which is passed through a 2D max pooling layer of dimensions 2x2, and then flattened.

Step 6: Output of previous layer passed through a fully connected dense layer with 256 perceptrons (ReLU activation).
Step 7: Compile the model (Adam optimizer with learning rate of 0.001, Categorical cross entropy loss, softmax activation)
Cross entropy loss function for binary classification can be given as equation 2,

$$\textbf{Cross Entropy Loss} = -(y\log(p) + (1-y)\log(1-p)) \quad (2)$$

where y is the binary indicator (0 or 1) and p is the predicted probability.
Softmax function can be given as equation (3),

$$\sigma(z)_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}} \text{ for i = 1, ……., K and } z = (z_1, ……., z_K) \epsilon R^K \quad (3)$$

3.1 Dataset

The dataset used is available on Kaggle under the name "Chest X-Ray Images (Pneumonia)." This 1.16GB dataset contains 5216 images for training and 624 images for testing. Images in this dataset are grayscale with the dimension of 64 X 64. The dataset consists of three types of images - Normal, Bacterial Pneumonia, and Viral Pneumonia. The dataset is available on the following weblink: https://www.kaggle.com/paultimothymooney/chest-xray-pneumonia.

Figure 2- Shows the "Chest X-Ray dataset (pneumonia), comprising of three types of images which are Normal, Bacterial Pneumonia, and Viral Pneumonia.

Figure 2 consists of three images where a normal chest x-ray image shows no abnormal opacification in the lungs, lobar consolidation is exhibited in the x-ray images of the chest in case of Bacterial Pneumonia. More diffuse 'interstitial' pattern is observed in both the lungs of Viral Pneumonia patients. The chest x-ray images depicted above are of patients in the age group of one to five from Guangzhou Women and Children's Medical Centre, Guangzhou.

3.2 CNN Architecture

CNN is a feed-forward neural network shown in Figure 3. It consists of four layers of processing- the convolutional layer, the pooling layer, flattening layer, and the fully-connected layer (Albarqouni, S. et al.). The following sub-headings give a detailed description of every layer in the CNN architecture.

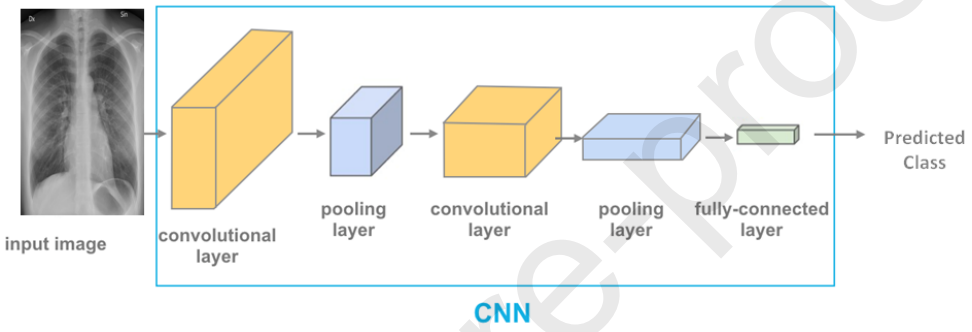


Figure-3 CNN Architecture

3.2.1 Convolutional Layer

The input image is converted into a matrix form. The convolution operation is applied between the input matrix and a feature detector/filter/kernel of dimension 3X3, and the result is a feature map (Xu, Y. et al.). This process reduces the dimensions of the image, which makes it easier to process the image. This also leads to loss of information, but the integral parts of the image are retained by the feature detector [18]. Multiple feature detectors are applied to the input matrix to obtain a layer of feature maps, which is our first convolutional layer. Further pooling and flattening are applied on this layer before these are fed into the fully-connected layer.

3.2.2 Activation Functions

Two different activation functions were used in all six models. These are ReLU and softmax activation functions. ReLU function is the most widely used activation function. The rectified linear function is a linear function which is applied to the convolutional layer, consisting of feature maps. The ReLU function outputs one if the input is positive. Otherwise, the output is zero (Krizhevsky, A. et al.). The neural network models that use ReLU function are easier to train and achieve better performance than models which use other activation functions such as sigmoid or hyperbolic tangent activation function as this function avoids and rectifies vanishing gradient problem. The ReLU function (Kingma, D. P. et al) is denoted by f(x) as given in Eq. 4 as follows:

$$f(x) = \max(0, x) \tag{4}$$

Softmax is another activation function which is widely used. Softmax function normalizes the inputs or logits into a probability distribution. Sum of all output probabilities in the distribution is equal to 1. Logits

are outputs of the logit layer or the last layer of the network. These are raw prediction values which range from minus infinity to infinity. The cost function generally used with softmax is categorical cross-entropy. Softmax activation function has been used in all six models.

3.2.3 Pooling layer

The purpose of pooling layer is to down-sample the input image further. In other words, to reduce the dimensions of the input image (Simonyan, K. et al.). The number of parameters of the image is reduced, hence reducing the computational complexity. The technique of sub-sampling used in the models is max-pooling and average-pooling. Max-pooling is a sample-based discretization process. The pooling layer of dimension 2X2 works over each feature map and scales its dimensionality using the 'MAX' function. Max-pooling selects the highest pixel value from the window of the image currently covered by the feature detector (Xu, Y. et al.). Max-pooling helps models recognize the salient features in the image

Average pooling is another sub-sampling technique that calculates the average value from the window of the image currently covered by the feature detector. Max-pooling is useful to recognize salient features of the image, but average pooling helps the neural network to identify the full extent of the image. Average pooling technique retains more amount of information in comparison to max-pooling.

3.2.4 Flattening layer & fully-connected layers

The pooled feature map is straightened out into a column so that it can be fed into the neural network (Cireşan, D. C. et al.). This enables the neural network to easily process the feature maps that are generated. After the input image is fed through the convolutional and pooling layer and the flattening layer, it is fed into the fully connected layer. Input forward propagates calculating weights. The network makes a prediction. Depending on the prediction, we calculate a cost function, which in this case is categorical cross-entropy. The cost function tells one how well a network is performing. The calculation of the cost function is followed by back propagation, tweaking weights, and feature maps to optimize the network. This process of forward and back propagation takes place until the network is fully optimized. Adam optimizer has been used in all six models (Shin, H. C. et al.). Adam is an optimization algorithm. Adam optimizer is used to iteratively update the weights of the network based on the training data. Adam optimizer is useful for networks, training on large datasets or parameters, being easy to implement, computationally efficient, and requiring a small amount of memory.

3.2.5 Reducing Overfitting

Dropout was employed in models 2, VGG16, VGG19, and ResNet to reduce overfitting. Dropout technique sets the output of each hidden neuron to zero with 0.5 probability. The neurons that are initialized to zero do not participate in forward and backward propagation (Baldi, P. et al.). This results in a reduction of complex co-adaptations of neurons, as each neuron needs to do something useful, without relying on other neurons in the same layer. Therefore, neurons are compelled to learn several salient features that are useful in conjunction. Data augmentation is another way of reducing overfitting. Learning rate was also changed in the models to reduce overfitting. Learning rate is a hyper-parameter which controls the extent of adjustment of the weights of the network concerning the loss gradient. This hyper-parameter affects the speed at which the network can converge to some local minima.

3.3 Model Architecture

Six models in total were trained and tested on the "Chest X-Ray Images(Pneumonia)" dataset. A detailed description of each model presented in the paper is given below:

3.2.1 Model 1

The first model consists of 2 convolutional layers, the first convolutional layer has 32 feature maps employing ReLU function, and the second convolutional layer has 64 feature maps employing ReLU function. Max-pooling layers of 2X2 dimensions are used after each convolutional layer. A flattening layer is placed behind these layers. 2 dense layers are used, the first dense layer has 256 output perceptrons employing ReLU and second dense layer with two output perceptrons using softmax function. Learning rate is reduced to 0.001. Adam optimizer has been used with categorical cross-entropy as the cost function.

3.3.2 Model 2

The second model consists of 3 convolutional layers; first convolutional layer has 32 feature maps employing ReLU, the second convolutional layer has 64 feature maps employing ReLU and third convolutional layer has 128 feature maps employing ReLU. Max-pooling layers of 2X2 dimensions are used after each convolutional layer. 2 dense layers have been used, the first dense layer has 256 output perceptrons employing ReLU and second dense layer with two output perceptron using softmax function. Dropout layer is also added. Learning rate of the model is reduced to 0.0001. Adam optimizer has been used with categorical cross-entropy as the cost function.

3.3.3 VGG16 and VGG19

VGG16 is a CNN model which was developed by K. Simonyan and A. Zisserman. It was one of the most notable models submitted to the ILSVRC 2014 competition. This model achieves a 92.7% top-5 test accuracy on the ImageNet dataset. In total, the network has 16 layers (Simonyan, K. et al.). VGG16 introduced multiple 3X3 kernel-sized filters one after the other replacing large kernel sized filters which were used in earlier models. Multiple layers of kernels result in increased depth of the neural network. This enables the neural network to understand and recognize more complex features and patterns. Vgg16 contains convolutional layers of 3x3 dimensions, average-pooling layers of 2x2 dimensions, and fully connected layers. The initial width of the neural network is 64. The width of the neural network doubles after each pooling layer. The first two fully connected layers, each have 256 channels, and the third layer has two channels. The first two hidden layers employ ReLU activation function, and the final layer employs a softmax activation function. Dropout was applied after each 256 channel dense layer. Learning rate of the network is 0.0001. Adam optimizer has been used with categorical cross-entropy as the cost function. The representational depth of VGG16 is beneficial for classification accuracy.

Vgg19 which is a variant of VGG16 is a 19-layer convolutional neural network which is used mainly for image classification. Its basic architecture is similar to that of VGG16 (Simonyan, K. et al.). The only difference in VGG19 is the use of 2 dense layers having 256 and two channels, and the learning rate being reduced to 0.00001.

3.3.4 ResNet50

ResNet stands for residual network and is primarily used for image classification. Microsoft's ResNet achieved a 3.57% top 5 error on the ImageNet dataset and won the ILSVRC classification contest in 2015 (He, K. et al.). The network's convolutional layers have 3X3 filters, and downsampling is done directly by the convolutional layers having a stride of 2. The last layer of the network is a fully-connected layer with 256 and two channels employing ReLU and softmax activation functions, respectively. Learning rate of the network is 0.000001. Adam optimizer has been used with categorical cross-entropy as the cost function.

Shortcut connections are used in ResNet to rectify the problems of degrading accuracy and vanishing gradient, which occur in deep neural networks. These connections allow the network to skip through layers which it feels are irrelevant for training. This reduces the training error and helps the network to converge faster in comparison to other networks. Figure 4 depicts the working of shortcut connections in ResNet50 model.

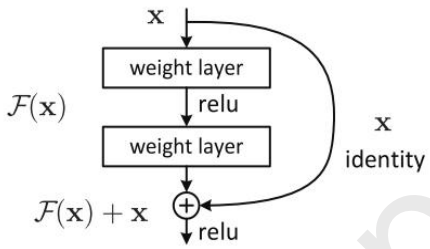


Figure 4 - shows the graphical representation of shortcut connections.

ResNet has multiple variants such as ResNet50, ResNeXt, ResNet34, ResNetV2 etc. ResNet50 was used on chest x-ray dataset to classify x-ray images into two classes. ResNet50 is a residual network consisting of 50 layers.

3.3.5 Inception-v3

Inception v3 depicted in figure 5 is a convolutional neural network used for image classification. Inception v3 is a CNN with 42 layers. It has multiple variants such as inceptionv1/google net, inception v2, and inception v4. Inception v1 was the first runner up at the ILSVRC 2015 competition (Szegedy, C. et al.). GoogleNet/ inception v1 was introduced in 2015, later with each new version; some new features were introduced.

Auxiliary classifiers were introduced in Inception v1. Auxiliary classifiers were added to avoid or prevent the activation of each layer to converge to zero. Batch normalization was introduced in Inception v2. This is a technique which rectifies the problem of vanishing gradients and zero activations by reducing the internal covariate shift. Additional factorization was first used in Inception v3, to reduce the number of connections/parameters of the network without decreasing the network efficiency. Learning rate of the network is 0.000001. Adam optimizer has been used with categorical cross-entropy as the cost function. Figure 4 shows the basic architecture of the inception-v3 network.

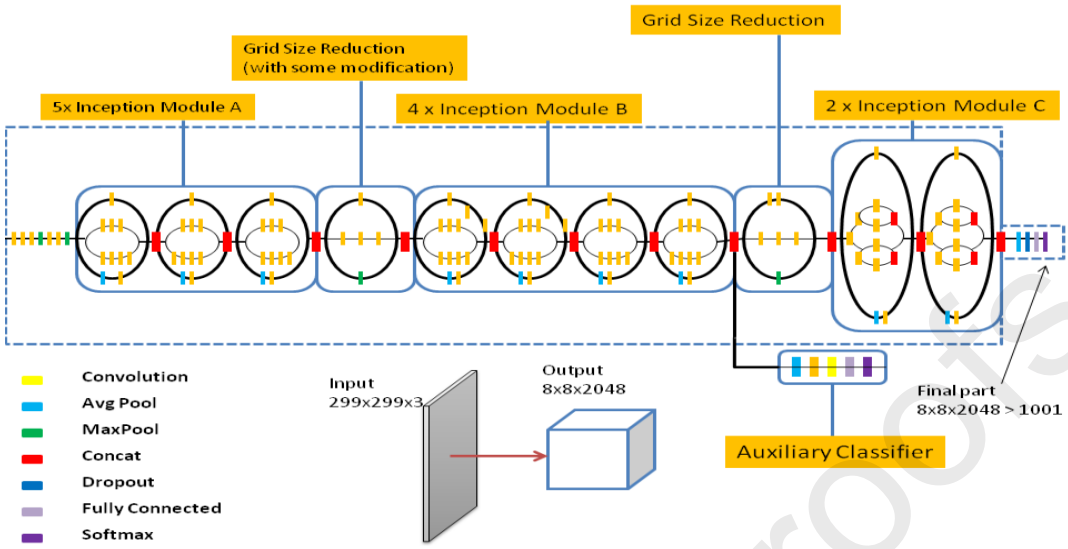


Figure 5 - Architecture of Inception-v3.

4. Experimental Results and Discussions

Six models were trained and tested on Chest X-Ray Images (Pneumonia) dataset consisting of 5216 images for training and 624 images for testing the models. The same technique of data pre-processing has been used for all six models. Performance measures used to analyze and identify the best performing models are Accuracy, Recall, and F1. The main significance of selecting an appropriate performance measure for classification task is an important challenge. In this paper, we have considered Accuracy, Recall and F1 score as evaluation measures. Accuracy measure is validation accuracy or classification accuracy of the model. The recall is used as performance evaluation measure in detection of bacterial pneumonia and viral pneumonia infected patients. If an actual positive patient is predicted as negative, then consequence can be very bad for the patient's health. Whereas precision is a good measure to evaluate the scenarios where false positive cost are high. A false positive means that the Chest X-Ray Images that are not having bacterial pneumonia and viral pneumonia infection, have been marked as bacterial pneumonia and viral pneumonia by the model. Precision is that how many of observations are really positive and how exact the model is. If precision is not high for the proposed model then, we may obtain a wrong diagnosis. F1 score performance measure is better than precision and recall as it will create balance between them for the uneven Normal, Bacterial Pneumonia, and Viral Pneumonia class distribution with large number of actual negatives.
The Accuracy (Hemanth, D. J. et al.) is given by Eq. 5 as follows:

$$Accuracy = \frac{t\_pos + t\_neg}{t\_pos + t\_neg + f\_pos + f\_neg} \quad (5)$$

Recall and F1 score (Hemanth, D. J. et al.) are given by Eq. 6 and Eq. 7 respectively which are as follows:

$$Recall = \frac{t\_pos}{t\_pos + f\_neg} \quad (6)$$

$$F = 2 \cdot \frac{precision \cdot recall}{precision + recall} \quad (7)$$

t_pos, t_neg, f_pos and f_neg mentioned in the above formula are True Positive, True Negative, False Positive and False Negative, respectively. The recall is the measure of a total number of true, relevant results that are returned. Recall of a model is crucial when the cost of false negatives is high. The recall is also known as sensitivity. Generally speaking, the F1 Score is the harmonic mean of precision and recall. If the F1 Score of a model is high, this means that the number of false positives and false negatives are less. It is a weighted average of recall and precision. The following sub-headings analyze the performance of the CNN models and the transfer learning models presented in this paper.

4.1 Analysis of CNN Models

For the experimental result evaluations out of the three classes normal patients, bacterial pneumonia and viral pneumonia for the sake of simplicity, the bacterial and viral pneumonia classes have been merged into one class as infected. The results have thus been evaluated as pneumonia predicted and normal.

Confusion matrix provides an insight about the error being made by the classifier. It is used to describe the performance of classification model on the test images for true values are known. It summarizes the production results. Confusion matrices of CNN models are given below:

Tables 1 and 2 - Confusion matrices of Model 1 and Model 2

	Predicted Label	
True Label	165	69
	23	367

	Predicted Label	
True Label	192	42
	6	384

Recall and F1 Score of CNN models are calculated from the above confusion matrices. Comparative analysis of performance measures of two CNN models has been presented below based on the results achieved while training and testing on the dataset.

Table 3 - shows the values of performance measures achieved by model 1 and model 2.

Model Name	Accuracy	Recall	F1 Score
Model 1	85.26%	94%	89%
Model 2	92.31%	98%	94%

Model 1 had training accuracy and training loss of 92.52% and 19.33% respectively. The validation accuracy achieved by model 1 is 85.26 %, whereas the validation loss is 38.36%. Similarly, for Model 2, the training accuracy and training losses are 96.30% and 9.98% respectively. The validation accuracy and validation loss attained by Model 2 are 92.31% and 25.23% respectively. Hence, it can be concluded that Model 2 has outperformed Model 1 as it has achieved higher value against each performance measure. Model 2 is not only a better performing model; it is a consistent and efficient model having scored above 90% in all the three performance measures and has an exceptionally high recall of 98%. Model 1 shows more over-fitting than Model 2. Model accuracy and model loss graph of each model is depicted in figure 6 and figure 7.

Figure 6 - shows the accuracy and loss graph of Model 1

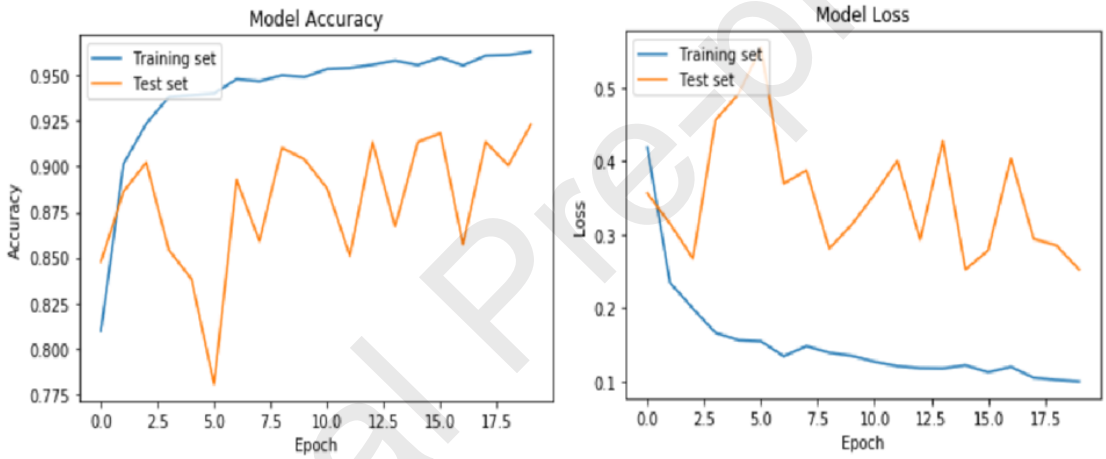


Figure 7 - Shows the accuracy and loss graph of Model 2

4.2 Analysis of Transfer Learning Models

Confusion matrices of Transfer Learning models are given below:

Tables 4, 5, 6 and 7 show Confusion matrices of VGG16, VGG19, ResNet50, and Inception-v3 resp.

True Label	Predicted Label	
	168	66
	14	376

True Label	Predicted Label	
	182	52
	20	370

True Label	Predicted Label	
	104	130
	10	380

True Label	Predicted Label	
	116	118
	63	327

Confusion matrices depicts the error made by the classifier models and from the analysis drawn from Table 1, 2, 4, 5, 6 and 7 it is observed that Model 2 has 6.7% error (minimum among all) while the error observed in all the other models is more than 10% and ResNet50 being the worst among these models with an error rate of 21%. Comparative analysis of performance measures of four Transfer Learning models (VGG16, VGG19, ResNet50, and Inception-v3) has been presented below based on the results achieved during training and testing on the dataset.

Table 8- shows the values of performance measures achieved by each model

Model Name	Accuracy	Recall	F1 Score
VGG16	87.18%	96%	90%
VGG19	88.46%	95%	91%
ResNet50	77.56%	97%	84%
Inception-v3	70.99%	84%	78%

Table 9 - shows the values accuracy and loss achieved by each model during training and validation.

Model Name	Training Accuracy	Training Loss	Validation Accuracy	Validation Loss
VGG16	95.61%	12.03%	87.17%	37.94%
VGG19	92.85%	18.01%	88.46%	34.29%
ResNet50	94.29%	14.32%	77.56%	68.36%
Inception-v3	88.96%	28.20%	70.99%	97.56%

ResNet50 and Inception-v3 show substantial overfitting as the difference between training, and validation accuracy is quite large. These two models have large validation loss, and their validation accuracy or classification accuracy is also low. Hence, these two models show poor performance. The graphical representation of model accuracy and model loss of ResNet50 and Inception-v3 are shown in figure 8 and figure 9, these show the variation in training and validation accuracies and loss with increase in epochs.

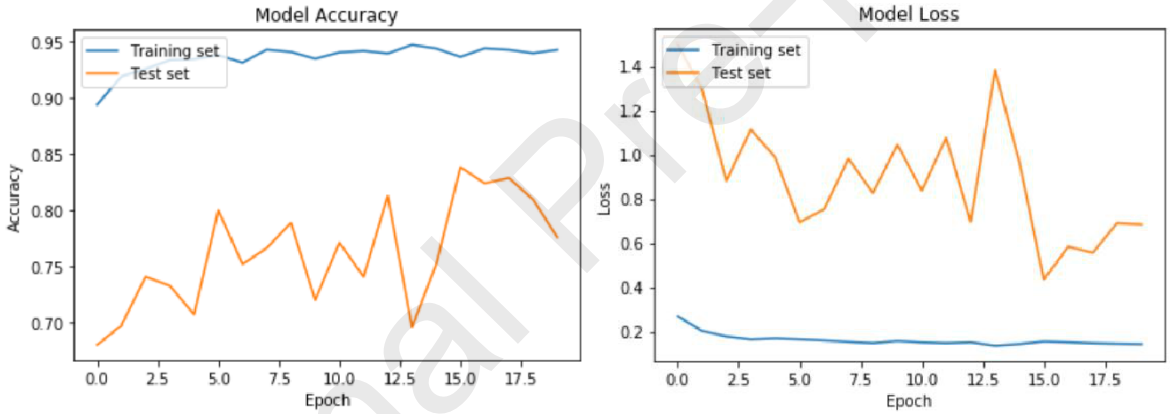


Figure 8 - shows the accuracy and loss graph of ResNet50.

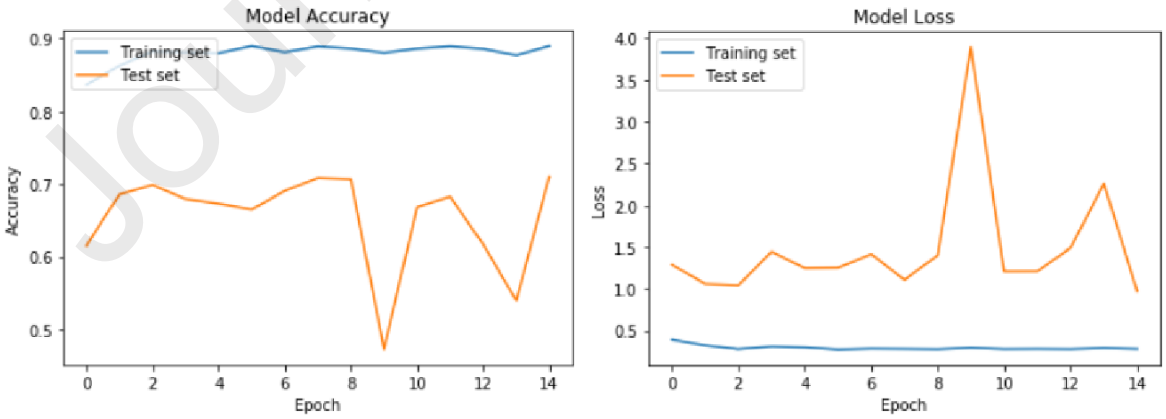


Figure 9 - shows the accuracy and loss graph of Inception-v3.

On the other hand, VGG16 and VGG19 show less overfitting. Their validation accuracy is also high. The conclusion that can be drawn from the above comparative analysis is that VGG19 outperforms every other Transfer Learning model as it has achieved the highest values for classification accuracy and F1 Score. Its recall is lesser than VGG16's, but VGG19 has a better overall performance. These four models are deep neural networks having a large number of layers. Their validation accuracy is lesser than the CNN models (shallow networks) presented above, given the smaller size of dataset used for training and testing. If larger datasets are used, these deep neural networks are likely to outperform the CNN models (shallow networks). The graphical representation of model accuracy and model loss of VGG16 and VGG19 are shown in figure 10 and figure 11 which show the variation in training and validation accuracies and loss with an increase in epochs.

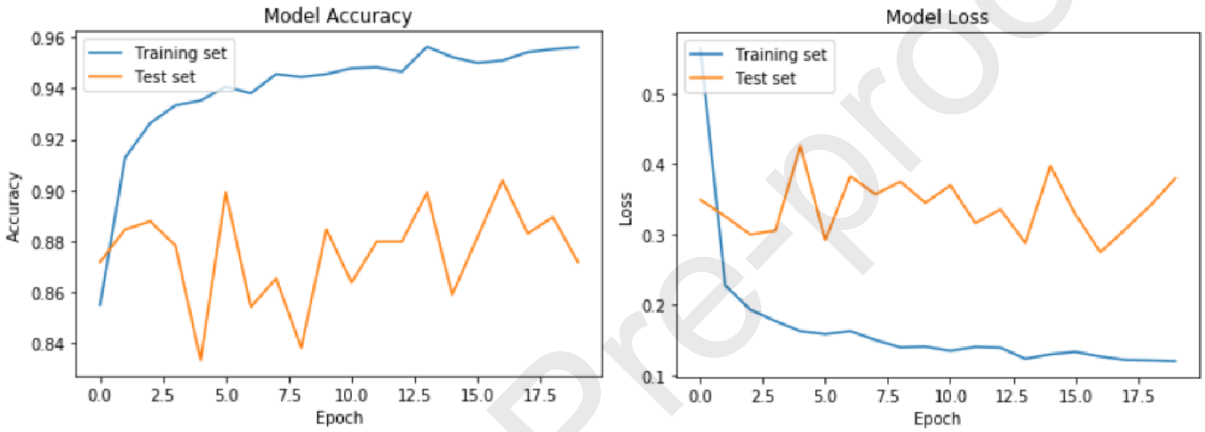


Figure 10 - shows the accuracy and loss graph of VGG16 model.

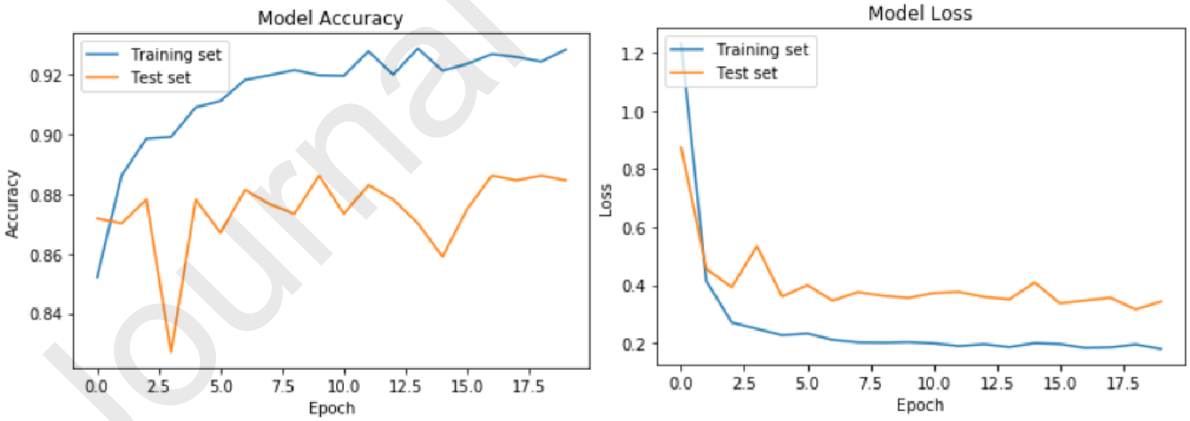


Figure 11 - shows the accuracy and loss graph of VGG19 model.

Table 10: Comparison with previous work

PAPER NUMBER	TOPIC	METHODOLOGY	ACCURACY	OUR OUTCOME
[38]	Pneumonia Detection Using CNN based Feature Extraction	CNN Models along with DenseNet-169 and SVM	80.02% Using DenseNet-169	85.28% using VGG16
[39]	A transfer learning method with deep residual network for pediatric pneumonia diagnosis	VGG16 and CNN	74.2% using VGG16	85.28% using VGG16
[40]	Deep neural network ensemble for pneumonia localization from a large-scale chest x-ray database	RetinaNet+ Mask RCNN	75.8%	77.56% using ResNet50
[41]	Diagnosis of Pneumonia from Chest X-Ray Images using Deep Learning	VGG16 and Xception	87% using VGG16 and 82% using Xception	87.28% using VGG16 and 88.46% using VGG19
[42]	Chest X-ray Image Classification Using Faster R-CNN	Fully connected RCNN	62%	92.31% using three CNN model

5. Conclusion and Future Work

This research paper presents two high performing neural networks for real-time applications. Both models are highly accurate and consistent. The recall is an important performance evaluator in this work as it is necessary to minimize the number of false negatives in the case of medical imaging. Recall of Model 2 is

as high as 98%, and VGG19 also attains a high recall of 95%. Model 2 and VGG19 networks obtained high f1 scores of 94% and 91% respectively.

In the view of an impressive performance against all performance measures, Model 2 and VGG19 models can be effectively used by medical officers for diagnostic purposes for early detection of pneumonia in children as well as adults. A large number of x-ray images can be processed very quickly to provide highly precise diagnostic results, thus helping healthcare systems provide efficient patient care services and reduce mortality rates.

In future work, authors of this paper aim to improve the classification accuracy of all the models by fine-tuning every parameter and hyper-parameter. Rajpurkar et al. presented ChexNet model, which is an efficient and accurate model that can be used for real-time applications. Models presented in this paper can be extended to classify other diseases as CheXNet did with high accuracy. Overall performance of the models can be improved with the use of larger datasets.

References

1. Cireşan, D. C., Meier, U., Masci, J., Gambardella, L. M., & Schmidhuber, J. (2011). High-performance neural networks for visual object classification. *arXiv preprint arXiv:1102.0183*.

2. O'Shea, K., & Nash, R. (2015). An introduction to convolutional neural networks. *arXiv preprint arXiv:1511.08458*.

3. Ypsilantis, P. P., & Montana, G. (2017). Learning what to look in chest X-rays with a recurrent visual attention model. *arXiv preprint arXiv:1701.06452*.

4. Rajpurkar, P., Irvin, J., Zhu, K., Yang, B., Mehta, H., Duan, T., & Lungren, M. P. (2017). Chexnet: Radiologist-level pneumonia detection on chest x-rays with deep learning. *arXiv preprint arXiv:1711.05225*.

5. Szegedy, C., Ioffe, S., Vanhoucke, V., & Alemi, A. A. (2017, February). Inception-v4, inception-resnet and the impact of residual connections on learning. In *Thirty-First AAAI Conference on Artificial Intelligence*.

6. Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

7. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2016). Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 2818-2826).

8. Albarqouni, S., Baur, C., Achilles, F., Belagiannis, V., Demirci, S., & Navab, N. (2016). Aggnet: deep learning from crowds for mitosis detection in breast cancer histology images. *IEEE transactions on medical imaging*, *35*(5), 1313-1321.

9. Xu, Y., Jia, Z., Ai, Y., Zhang, F., Lai, M., Eric, I., & Chang, C. (2015, April). Deep convolutional activation features for large scale brain tumor histopathology image classification and segmentation. In *2015 international conference on acoustics, speech and signal processing (ICASSP)* (pp. 947-951).

10. Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems* (pp. 1097-1105).

11. Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

12. Shin, H. C., Roberts, K., Lu, L., Demner-Fushman, D., Yao, J., & Summers, R. M. (2016). Learning to read chest x-rays: Recurrent neural cascade model for automated image annotation. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 2497-2506).

13. Baldi, P., & Sadowski, P. J. (2013). Understanding dropout. In *Advances in neural information processing systems* (pp. 2814-2822).

14. He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770-778).

15. https://en.wikipedia.org/wiki/Precision_and_recall

16.https://www.lung.org/lung-health-and-diseases/lung-disease-lookup/pneumonia/what-causes-pneumonia.html

17. Ciresan, D. C., Meier, U., Masci, J., Gambardella, L. M., & Schmidhuber, J. (2011, June). Flexible, high performance convolutional neural networks for image classification. In *Twenty-Second International Joint Conference on Artificial Intelligence*.

18. Rubin, J., Sanghavi, D., Zhao, C., Lee, K., Qadir, A., & Xu-Wilson, M. (2018). Large scale automated reading of frontal and lateral chest x-rays using dual convolutional neural networks. *arXiv preprint arXiv:1804.07839*.

19. Lakhani, P., & Sundaram, B. (2017). Deep learning at chest radiography: automated classification of pulmonary tuberculosis by using convolutional neural networks. *Radiology*, *284*(2), 574-582.

20. https://www.who.int/news-room/fact-sheets/detail/pneumonia.

21. https://www.kaggle.com/paultimothymooney/chest-xray-pneumonia.

22. Srivastava, N. (2013). Improving neural networks with dropout. *University of Toronto*, *182*, 566.

23. Kalchbrenner, N., Grefenstette, E., & Blunsom, P. (2014). A convolutional neural network for modelling sentences. *arXiv preprint arXiv:1404.2188*.

24. Cireşan, D., Meier, U., & Schmidhuber, J. (2012). Multi-column deep neural networks for image classification. *arXiv preprint arXiv:1202.2745*.

25. Zeiler, M. D., & Fergus, R. (2014, September). Visualizing and understanding convolutional networks. In *European conference on computer vision* (pp. 818-833). Springer, Cham.

26. Guan, Q., Huang, Y., Zhong, Z., Zheng, Z., Zheng, L., & Yang, Y. (2018). Diagnose like a radiologist: Attention guided convolutional neural network for thorax disease classification. *arXiv preprint arXiv:1801.09927*.

27. LeCun, Y., Huang, F. J., & Bottou, L. (2004, June). Learning methods for generic object recognition with invariance to pose and lighting. In *CVPR (2)* (pp. 97-104).

28. Hou, L., Samaras, D., Kurc, T. M., Gao, Y., Davis, J. E., & Saltz, J. H. (2016). Patch-based convolutional neural network for whole slide tissue image classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 2424-2433).

29. Oquab, M., Bottou, L., Laptev, I., & Sivic, J. (2014, June). Weakly supervised object recognition with convolutional neural networks. In *Proc. of NIPS*.

30. Anthimopoulos, M., Christodoulidis, S., Ebner, L., Christe, A., & Mougiakakou, S. (2016). Lung pattern classification for interstitial lung diseases using a deep convolutional neural network. *IEEE transactions on medical imaging*, *35*(5), 1207-1216.

31. https://www.datacamp.com/community/tutorials/convolutional-neural-networks-python .

32. https://en.wikipedia.org/wiki/Rectifier_(neural_networks) .

33. Pesce, E., Ypsilantis, P. P., Withey, S., Bakewell, R., Goh, V., & Montana, G. (2017). Learning to detect chest radiographs containing lung nodules using visual attention networks. *arXiv preprint arXiv:1712.00996*.

34. Cicero, M., Bilbily, A., Colak, E., Dowdell, T., Gray, B., Perampaladas, K., & Barfett, J. (2017). Training and validating a deep convolutional neural network for computer-aided detection and classification of abnormalities on frontal chest radiographs. *Investigative radiology*, *52*(5), 281-287.

35. Glozman, T., & Liba, O. (2016). Hidden Cues: Deep Learning for Alzheimer's Disease Classification CS331B project final report.

36. Cho, Y., Seong, J. K., Jeong, Y., Shin, S. Y., & Alzheimer's Disease Neuroimaging Initiative. (2012). Individual subject classification for Alzheimer's disease based on incremental learning using a spatial frequency representation of cortical thickness data. *Neuroimage*, *59*(3), 2217-2230.

37. Hemanth, D. J., Vijila, C. K. S., Selvakumar, A. I., & Anitha, J. (2014). Performance improved iteration-free artificial neural networks for abnormal magnetic resonance brain image classification. *Neurocomputing*, *130*, 98-107.

38. Varshni, D., Thakral, K., Agarwal, L., Nijhawan, R., & Mittal, A. (2019, February). Pneumonia Detection Using CNN based Feature Extraction. In *2019 IEEE International Conference on Electrical, Computer and Communication Technologies (ICECCT)* (pp. 1-7). IEEE.

39. Liang, G., & Zheng, L. (2019). A transfer learning method with deep residual network for pediatric pneumonia diagnosis. *Computer methods and programs in biomedicine*, 104964.

40. Sirazitdinov, I., Kholiavchenko, M., Mustafaev, T., Yixuan, Y., Kuleev, R., & Ibragimov, B. (2019). Deep neural network ensemble for pneumonia localization from a large-scale chest x-ray database. *Computers & Electrical Engineering*, *78*, 388-399.

41. Ayan, E., & Ünver, H. M. (2019, April). Diagnosis of Pneumonia from Chest X-Ray Images Using Deep Learning. In *2019 Scientific Meeting on Electrical-Electronics & Biomedical Engineering and Computer Science (EBBT)* (pp. 1-5). IEEE.

42. Ismail, A., Rahmat, T., & Aliman, S. (2019). CHEST X-RAY IMAGE CLASSIFICATION USING FASTER R-CNN. *MALAYSIAN JOURNAL OF COMPUTING*, *4*(1), 225-236.

43. Jaiswal, A., Tiwari, P., Kumar, S., Gupta, D., Khanna, A., & Rodrigues, J. (2019). Identifying pneumonia in chest X-rays: A deep learning approach. *Measurement*, *145*, 511-518.

44. Elhoseny, M., & Shankar, K. (2019). Optimal bilateral filter and Convolutional Neural Network based denoising method of medical image measurements. *Measurement*, *143*, 125-135.

45. Chandra, T., & Verma, K. (2020). Analysis of quantum noise-reducing filters on chest X-ray images: A review. *Measurement*, *153*, 107426.

46. Wang, J., Mo, Z., Zhang, H., & Miao, Q. (2020). Ensemble diagnosis method based on transfer learning and incremental learning towards mechanical big data. *Measurement*, *155*, 107517.

47. Mao, W., Ding, L., Tian, S., & Liang, X. (2020). Online detection for bearing incipient fault based on deep transfer learning. *Measurement*, *152*, 107278.

HIGHLIGHTS

- Deep learning-based pneumonia detection in x-ray images is done in this work
- Different models of deep learning and transfer learning are analysed in this work for the image classification application.
- An extensive analysis is carried out in this work with several experimental results

Credit author statement

Rachna Jain - Conceptualization
Preeti Nagrath – Formal analysis
Gaurav Kataria – Investigation & methodology
V. Sirish Kaushik – Investigation & methodology
D. Jude Hemanth - Validation