

# CLASSIFICATION OF X-RAY IMAGES FOR DETECTING COVID-19 USING DEEP TRANSFER LEARNING

NIDHI BANSAL\*

*\*Department of Information Science and Technology, Anna University, CEG Guindy,  
Chennai, Tamilnadu 600025, India  
goelnidhi8@gmail.com*

S. SRIDHAR

*Department of Information Science and Technology, Anna University, CEG Guindy,  
Chennai, Tamilnadu 600025, India  
ssridhar2004@gmail.com*

**Abstract:** The coronavirus disease COVID-19 eruption is stated as a pandemic by the World Health Organization. It is affecting around 212 countries and territories across the globe. There is a need to constantly analyze and find patterns from lungs X-Ray images. Early diagnosis can constraint the exposure of person and aids to bound the feast of the virus. The manual diagnosis is quite tedious and time-consuming process. The main aim of this paper is to explore the transfer learning potential. A deep learning framework is proposed adopting the capability of pretrained Deep Convolutional Neural Network models with transfer learning. This assists in classification of the chest X-Ray Images with high level of accuracy. An analysis is done with utilization of six pretrained models – VGG16, VGG19, ResNet50V2, InceptionV3, Xception and NASNetLarge. The experiment results showed that the highest accuracy obtained was 97% using VGG16 and VGG19 with sensitivity and specificity of 100% and 94% respectively.

*Keywords:* Deep Learning; COVID-19; Chest X-Ray; Transfer Learning.

## 1. Introduction

The coronavirus COVID-19 has proven to be a Public Health Emergency with ever increasing cases but limited healthcare facilities and resources. The new coronavirus is human-to-human transmissible disease.<sup>1</sup> With the number of confirmed cases approaching 5 million over the globe and approx. 3 lakhs deaths worldwide,<sup>2</sup> the new virus is endangering the existence of human race. The situation is alarming as the new cases has increased to approx. five times in the past one month<sup>3</sup> and there is a sudden rise of 1 million number of cases in last two weeks.<sup>3</sup> The common symptoms of coronavirus are high fever, cough, sore throat and difficulty in breathing. It starts with a fever and can develop into a pneumonia. The complications may include shortness of breath, chest tightness, and other severe infections. It is found in a report<sup>4</sup> that 80.9% of cases are mild, 14% are severe and 5% are critical. The risk of death is more in elderly people and people with pre-existing ailments.<sup>4</sup> The higher risk patients are with cardio disease, asthma, diabetes and hypertension. Relatively few cases are seen among children.<sup>4</sup>

The global experts are working rigorously to fight the pandemic and find best solutions. The uncontrollable spread of the COVID-19 may be due to several reasons. Initially, a patient may have little or no symptoms at all. By the time the virus is diagnosed, a person might have transmitted to several other persons. Secondly, it is similar to a flu, so a person doesn't feel anything in earlier stage. Symptoms may take 2-14 days to appear.<sup>5</sup> Third, number of testing kits are not sufficiently available to test enough population. Moreover,

there is no vaccine developed so far to cure the 2019 coronavirus, so it takes time for treatment and its spread is difficult to control.

The detection and classification of diseases in medical images has been made automatic and more efficient with Deep Learning methods. Convolutional neural network (CNN), have extensive applications in medical image processing, such as classification of benign and malignant tumors, segmentation of skin lesion, pneumonia detection and many more. In the proposed model, pre-trained CNN models are used to classify chest X-Ray images and calculate the probability of infection with COVID-19. The findings might assist in the early screening of patients with urgent intervention.

## **2. Deep Transfer Learning**

Transfer learning refers to a process where a model trained on one problem is used to fine tune the other problem. It assists in decreasing the training time for a model and building a more generalized model.<sup>6</sup> The training of Deep Convolutional Neural Networks (CNN) from scratch is difficult as it demands huge amount of training data. Also, the CNN models demand extensive time, sometimes in days or even weeks. These limitations can be overcome by reusing the model weights from pre-trained models. These already trained models were developed for benchmark datasets, such as the ImageNet image recognition tasks.<sup>6</sup> Models are trained on more than one million images and capable to categorize images into 1000 classes. The models with high performance can be used directly, or for a new problem it can be blended.

There are many ways to use the pre-trained models. First, it can be used to classify images from new dataset. Second way, the pre-trained model is used for image pre-processing and feature extraction. Third way, first few layers of the model trained are frozen during training and the last layers are retrained according to the one's dataset. The state-of-the-art architecture models used as transfer learning are Visual Geometry Group (VGG16),<sup>7</sup> (VGG19),<sup>7</sup> Residual Networks (ResNet),<sup>9</sup> Inception CNNs,<sup>11</sup> Xception,<sup>15</sup> NASNetLarge<sup>20</sup> etc.

### **2.1. Visual Geometry Group(VGG)**

VGG was introduced in 2014.<sup>7</sup> The convolutional(conv.) layer1 receives an image with a size 224 x 224 as the input. The image is passed through a stack of conv. layers, with 3x3 filter size. Max Pooling is done over a 2 x 2 window. There are total 21 layers which comprises 13 conv., 5 Max Pooling and 3 Dense layers. Conv. 1 has 64 filters, Conv. 2 has 128 filters, Conv. 3 has 256 filters while Conv. 4 and Conv. 5 has 512 filters. The final three dense layers has a channel size of 4096, 4096, and 1000 respectively.<sup>7</sup> The model size is 528MB and the network achieves 90.1% top-5 accuracy.<sup>8</sup> VGG-19 is a variant of VGG-16 with 19 weight layers (16 conv. And 3 Dense).<sup>7</sup> It outperforms AlexNet and GoogleNet by replacing large filters of size 11 and 5 with small size of 3x3. VGGNets are slow to learn, however, they are used in many image classification tasks due to good classification accuracy.

## **2.2. Residual Neural Network (ResNet)**

Residual Neural Network was introduced in 2015.<sup>9</sup> ResNet can train up to hundreds and thousands of layers and achieves significant performance with an image input size of 224 x 224. However, deep networks are hard to train because of the vanishing gradient problem<sup>10</sup> during backpropagation, i.e. when the weights are updated so slow, they stop updating after a while and the gradient becomes infinitively small. It results in performance degradation rapidly as the network starts going deeper.

The essence of ResNet resides in “residual blocks”, qualifies the network to preserve the information learnt previously by using an identity mapping weight function, without bothering about the vanishing gradient problem.<sup>10</sup> Therefore, the deeper model will not produce a high training error. A 152 layer ResNet is used on ImageNet challenge but the architecture is still less complex.<sup>9</sup> ResNet employs global average pooling instead of dense layers, so model size is comparatively smaller than VGG16 and VGG19. The reduced model size is 98MB for ResNet50 and ResNet50V2, 171 MB for ResNet101 and ResNet101V2, 232 MB for ResNet152 and ResNet152V2.<sup>8</sup> The top-5 accuracy of the ResNet50V2 is 93%, while 94.2% for ResNet152V2.<sup>8</sup>

## **2.3. Inception Network**

The Inception architecture was first introduced in 2014<sup>11</sup> and was a major breakthrough in the evolution of CNN models. The original architecture InceptionV1 was introduced by GoogLeNet with the input image size of 299 x 299. The main aim of the inception net is to compute 1×1, 3×3 and 5×5 conv. within the same module. Outputs of these convolutions are stacked and fed in the next network layer. The architecture was later enhanced by Ioffe with the addition of batch normalization in 2015 called InceptionV2.<sup>12</sup> The Inception V3<sup>13</sup> architecture proposed factorization to the inception module to promote the classification accuracy in ImageNet, in 2015 publication. The size of the model is 92 MB and top 5 error rate is only 4.2%.<sup>8</sup>

InceptionV4<sup>14</sup> has higher no. of inception modules than InceptionV3. Inception network with residual connections (InceptionResNet) outperforms expensive Inception network with no residual connections. InceptionResNet models were able to achieve higher accuracies than Inception in ILSVRC classification task.<sup>8</sup>

## **2.4. Xception Net**

Inception architecture is extended to replace the Inception modules. These modules are named depthwise separable convolutions called Xception.<sup>15</sup> The architecture of Xception is a linear stack of depthwise separable convolution layers with residual connections.<sup>15</sup> The network has an image input size of 299-by-299 and is 71 layers deep with top 5 accuracy of 94.5%.<sup>8</sup> Xception Net performed better than earlier models in many image classification challenges.

### 3. Methodology

Transfer Learning is an efficient approach to train CNN when there is a dearth of adequate training data and computational resources.<sup>16</sup> The parameters learnt on large data such as ImageNet are used for weight initialization. Transfer learning can be implemented on CPU's since training and classification takes less time to complete, with no special requirements for GPU's. The features of the previous layers of a pre-trained network usually contain low-level (edge, color) information. The later layers contain high-level attributes, particularly the categorical details. In this paper, an approach grounded upon transfer learning based technique is proposed which employs pre-trained weights for initial layers of the network and last few layers are fine-tuned according to the dataset.<sup>6</sup> This assists in classifying the chest X-Ray Images and identify the COVID-19 images from the dataset. The proposed network architecture is shown in Fig.1.

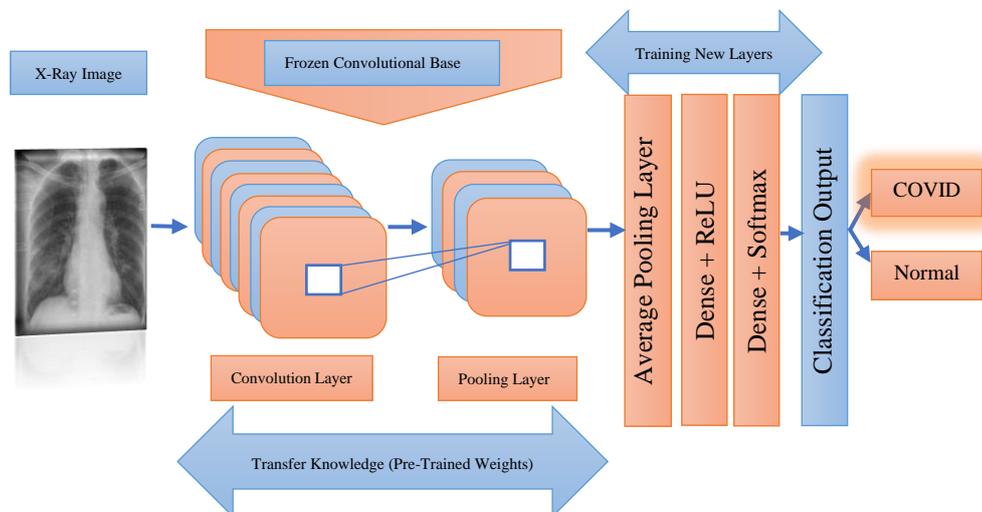


Fig. 1. The architecture of proposed model

#### 3.1. Dataset Description

The dataset is taken from Kaggle dataset repository,<sup>17</sup> it is an open access database of chest X-Ray images. It consists of X-Ray images of COVID-19, bacterial and viral pneumonia and normal people. The two image classes are considered for this work: COVID-19 and normal images. There are 70 COVID-19 and 930 normal images. The dataset is skewed towards normal chest images. Therefore, for the proposed work 70 COVID-19 and 80 normal images are considered. Each image has a different size. These images are pre-processed to resize them to 224x224 pixels, to be fed to network. The dataset includes both training and testing images. 20% of the data is used for validation. Fig 2 and Fig. 3 shows some COVID-19 and normal X-Ray images from dataset.

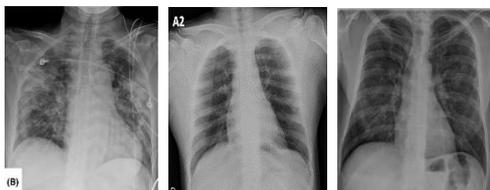


Fig. 2. COVID-19 Chest X-Ray Images

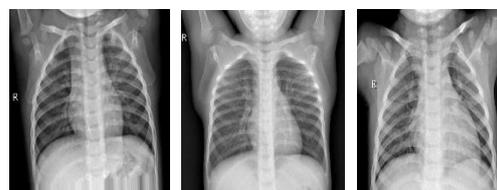


Fig. 3. Normal Chest X-Ray Images

### 3.2. Training and Algorithm

The explanation of the proposed the model is as follows. Some layers of a pre-trained model are used as a feature extraction component. The last few layers need to be finetuned.<sup>6</sup> The approach for fine-tuning is to load the model, then simply add new layers. This can be done by specifying the include top argument to False.

The final three layers of the model are modified conforming to the new classification task as follows:

- (i) “Average Pooling” Layer,
- (ii) “Dense (fully-connected)” layer with ReLU activation function,<sup>18</sup>
- (iii) “Dense (fully-connected)” layer with Softmax function<sup>18</sup> with a classification output.

The pooling layer is added to extend the feature extraction potential of model and, the new fully connected layers are added to perform classification by learning features of a new dataset. ReLU (Rectified Linear Unit) is an activation function that is most commonly used in CNNs. The function introduces non-linearity to the structure and is preferred as it converges faster and overcomes the vanishing gradient problem.<sup>10</sup> Mathematically, it is defined as  $y = \max(0, x)$ .<sup>18</sup> The size of the final dense layer is set to the two which indicates the no. of classes in the given dataset i.e., COVID-19 & normal. The complete methodology for chest X-Ray images classification using the transfer learning is given in Algorithm 1.

#### Algorithm 1:

Step 1: The dataset is taken from Kaggle Dataset repository<sup>17</sup>, it contains two folders COVID-19 and normal chest X-Ray images. Some images are shown in Fig. 2 and Fig. 3.

Step 2: Resize the input images so that they are consistent with the size of the input layer of pre-trained network.

Step 3: Images are augmented to handle varying rotations during training, by setting rotation to 15 degrees.

Step 4: Partition the data into training and test sets; 80% of images are used for training and 20% as a test dataset to test the network.

Step 5: Modify the Network Architecture by swapping the final layers of the pre-trained network as: “average pooling”, “fully-connected layer”, “softmax” with a “classification output”, to identify the probability of COVID-19 and normal class.

Step 6: Train the Network.

Step 7: Test the new classifier on the testing dataset.

Step 8: Plot the accuracy and loss during training and validation phase to analyze the performance of the model.

### 3.3. Performance Measures

It is crucial step to validate the effectiveness of a model. The performance of the proposed model is evaluated on the parameters of Sensitivity, Specificity, and Accuracy. They can be mathematically calculated using equations Eq. (1)-(3).<sup>19</sup> In the equations,  $P$  and  $N$  indicate the number of positive and negative samples respectively while  $TP$  is True positive,  $FN$  is False Negative,  $TN$  is True Negative and  $FP$  is False Positive.

*Accuracy* is the ratio of the correctly classified samples to the total number of samples.

$$Accuracy = \frac{TP+TN}{TP+FP+TN+FN} \quad (1)$$

*Sensitivity* is the *True positive rate (TPR)*, it represents the ratio of the positive correctly classified samples to the total number of positive samples.

$$Sensitivity = \frac{TP}{TP+FN} \quad (2)$$

*Specificity* is the *True negative rate (TNR)*, it represents the ratio of the correctly classified negative samples to the total number of negative samples.

$$Specificity = \frac{TN}{TN+FP} \quad (3)$$

## 4. Experimental Results and Discussion

The proposed network model has been evaluated using six different transfer learning models namely VGG16,<sup>7</sup> VGG19,<sup>7</sup> ResNet50V2,<sup>9</sup> InceptionV3,<sup>11</sup> Xception<sup>15</sup> and NASNetLarge.<sup>20</sup> In this paper, the weights from pre-trained models are used and the results obtained are compared. The performance is analyzed using the measures of accuracy, sensitivity and specificity<sup>19</sup> to detect COVID-19 from X-Ray images. Table 1 shows a comparison of classification accuracy of the proposed network model using different transfer learning models. The highest accuracy to detect COVID-19 is 97% using VGG16 and VGG19. The accuracy is quite good with a sensitivity of 100% on VGG model, means a person with “COVID-19 positive” is identified as positive accurately. But the specificity using VGG (94%) is a bit concerning, a patient should not be classified as “COVID negative” when the same is positive. As the coronavirus spreads human-to-human, if it is wrongly identified as negative, it can transmit to a big community.

Though ResNet50V2 also gives 97% accuracy with a sensitivity of 93%, but specificity is 100%. Xception Net, NasNetLarge and InceptionV3 provide accuracy of 94%, 94% and 91% respectively and giving 100% specificity, means we could accurately identify them as “COVID-19 negative”. The low sensitivity is also dangerous and should be avoided. If a patient is misclassified as COVID positive, the same would be isolated with other patients, and spend unnecessary time in hospital and may become infected. It is very critical and challenging to maintain a balance b/w sensitivity and specificity in medical imaging

tasks, while using Deep Learning models for prediction. A single misdiagnosis can cost lives of many people especially for contagious diseases and that transmit fast like COVID-19. The given algorithm is implemented in Anaconda Jupyter platform and executed on a system with Intel Core CPU, 8 GB RAM, and 1.60 GHz.

Table 1. Comparison of Classification Accuracy of the proposed network model using different transfer learning models.

Model	Accuracy	Sensitivity	Specificity
VGG16	97	100	94
VGG19	97	100	94
ResNet50V2	97	93	100
InceptionV3	91	79	100
Xception	94	86	100
NASNetLarge	94	86	100

The plots for showing accuracy and loss function are shown in Fig. 4-7 respectively for VGG16, VGG19, ResNet50V2 and Xception Net. The plot shows both curves during the training and validation process. It can be depicted that model doesn't overfit even we didn't have a huge dataset. It is concluded that 1) Building a new model is difficult as deciding the initial random weights is challenging. Transfer Learning models are a good choice for weight initialization for the chest X-Ray images dataset. 2) Deep CNN pre-trained weights provide better performance to detect COVID from chest X-Ray images data in terms of accuracy, sensitivity and specificity. With these Deep CNN models, it is possible to detect

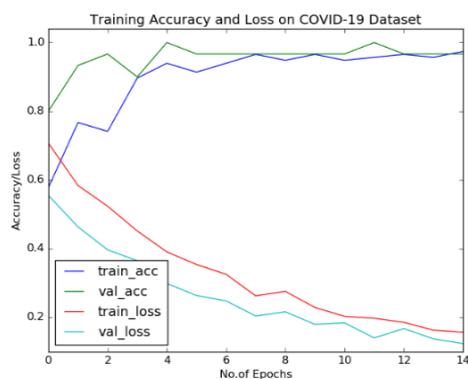


Fig. 4. Accuracy and Loss using VGG16.

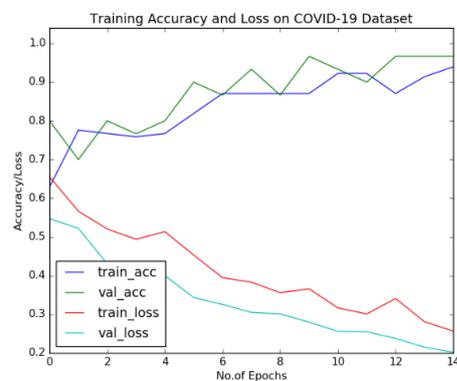


Fig. 5. Accuracy and Loss using VGG19.

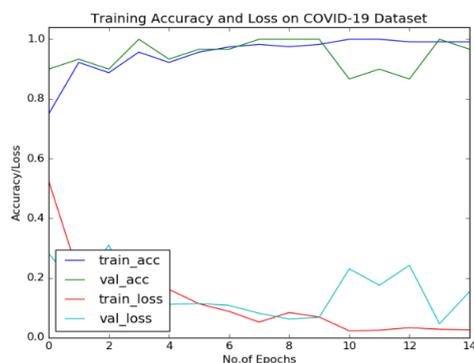


Fig. 6. Accuracy and Loss using ResNet50V2.

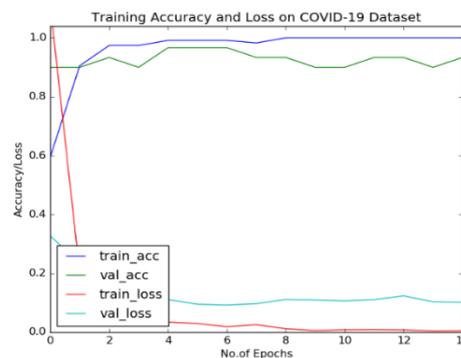


Fig. 7. Accuracy and Loss using Xception Net.

COVID-19 at an earlier stage and provide appropriate treatment to overcome the disease outbreak. This framework can be enhanced in the future by using a more powerful CNN network with high accuracy and by increasing the no. of images in the datasets.

### Conflict of Interest

On behalf of all authors, the corresponding author states that there is no conflict of interest.

### Acknowledgments

This research work was supported by Centre For Research, Anna University under the Anna Centenary Research Fellowship, Anna University, Chennai, India.

### References

1. Chen et al., Epidemiological and clinical characteristics of 99 cases of 2019 novel coronavirus pneumonia in Wuhan, China: a descriptive study, *The Lancet* 395. 10223 (2020) 507-513.
2. <https://www.who.int/emergencies/diseases/novel-coronavirus-2019>
3. <https://www.worldometers.info/coronavirus/>
4. Wu, Zunyou, and Jennifer M. McGoogan, Characteristics of and important lessons from the coronavirus disease 2019 (COVID-19) outbreak in China: summary of a report of 72 314 cases from the Chinese Center for Disease Control and Prevention, *Jama*. (2020).
5. <https://www.cdc.gov/coronavirus/2019-ncov/about/symptoms.html>
6. Jason Brownlee, *Deep Learning for Computer Vision* (eBook, 2020)
7. K. Simonyan and A. Zisserman, Very deep convolutional networks for large-scale image recognition, *arXiv Prepr. arXiv1409.1556* (2014).
8. <https://keras.io/applications/#available-models>
9. K. He, X. Zhang, S. Ren, and J. Sun., Deep residual learning for image recognition, *arXiv preprint arXiv:1512.03385* (2015).
10. Hochreiter, Sepp, The vanishing gradient problem during learning recurrent neural nets and problem solutions, *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* 6. 02 (1998) 107-116.

11. C. Szegedy et al., Going deeper with convolutions, in *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, (2015), pp. 1-9.
12. Ioffe, Sergey, and Christian Szegedy, Batch normalization: Accelerating deep network training by reducing internal covariate shift, *arXiv preprint arXiv:1502.03167* (2015).
13. Szegedy, Christian et al., Rethinking the inception architecture for computer vision, in *Proc. of the IEEE conference on computer vision and pattern recognition*, (2016), pp. 2818-2826.
14. Szegedy, Christian, et al., Inception-v4, inception-resnet and the impact of residual connections on learning, *Thirty-first AAAI conference on artificial intelligence*. 2017.
15. Chollet, François, Xception: Deep learning with depthwise separable convolutions, in *Proc. of the IEEE conference on computer vision and pattern recognition*, (2017), pp. 1251-1258.
16. Litjens et al., A survey on deep learning in medical image analysis, *Medical image analysis*. 42 (2017) 60-88.
17. <https://www.kaggle.com/darshan1504/covid19-detection-xray-dataset>
18. Nwankpa, Chigozie et al., Activation functions: Comparison of trends in practice and research for deep learning, *arXiv preprint arXiv:1811.03378* (2018).
19. Tharwat, Alaa, Classification assessment methods, *Applied Computing and Informatics*. (2018).
20. Zoph, Barret, Vijay Vasudevan, Jonathon Shlens, and Quoc V. Le, Learning transferable architectures for scalable image recognition, In *Proc. of the IEEE conference on computer vision and pattern recognition*, (2018), pp. 8697-8710.