

# Covid 19 and Pneumonia Detection Using X-Ray

Aarzoo Sharma<sup>#1</sup>, Arnav Verma<sup>#1</sup>, Prashita Biswas<sup>#1</sup>, Divyam Kumar<sup>#1</sup>

<sup>#</sup>*Department of Computer Science, Thapar University  
Patiala, India*

**Abstract**— The exponential increase in Coronavirus 2019/2020, also called "Covid-19" patients, is overwhelming healthcare systems worldwide. With delimited testing kits, every patient with respiratory illness cannot be tested using traditional techniques (RT-PCR). The tests also have a long time and restricted sensitivity. Detecting possible Coronavirus infections based on the available data and genome information, this method may generate an optimal way of quickly identifying new covid cases and replace insufficient temperature/thermometer "screening" processes.

We propose using smart CT scan for further RT-PCR testing in this work. This may be helpful where the present systems struggle to keep patients in the ward and isolate them in quarantine areas. It would also help identify patients with a high likelihood of COVID with a false negative RT-PCR who would need repeat testing. Further, the MinION genome sequencer can produce this information in "seconds."

We propose a trained algorithm to return the prediction or classification of new unlabelled genome data and detect new cases with reasonable confidence of the latest coronavirus.

On the publicly available covid-genome dataset, our model gives 85% accuracy with 77% sensitivity (recall) for the novel coronavirus.

**Keywords**—Deep Learning, Neural Networks, CT Scan, COVID-19 Detection.

## INTRODUCTION

The abrupt spike in the number of patients with COVID-19, another respiratory infection, has put an uncommon burden on medical care frameworks worldwide. In numerous nations, the medical care frameworks have, as of now, been overpowered. There are restricted units for determination, restricted clinic beds for affirmation of such patients, restricted personal protective equipment (PPE) for medical services faculty, and restricted ventilators. It is in this manner vital to separate which patients with extreme intense respiratory ailment (SARI) could have COVID-19 disease to proficiently use the restricted assets. In this assignment, we propose using a smart CT scan to

recognize COVID-19 disease in the patients displaying side effects of SARI. Utilizing our apparatus, one can order a given CT scan in one of the four classes: ordinary, bacterial pneumonia, viral pneumonia, and Coronavirus pneumonia. The utilization of CT scan enjoys a few upper hands over standard symptomatic tests:

1. CT scanning is much more widespread than the usual diagnostic tests.
2. The move of computerized CT scans doesn't need any transportation from the point of obtaining to the reason behind the investigation, accordingly making the process incredibly speedy.
3. The primary commitment of this work is in proposing a clever, profound neural organization-based model for the exact identification of COVID-19 disease from the CT scan of the patients.

Radiographs in the current setting are by and large deciphered by non-radiologists. Further, given the curiosity of the infection, a considerable lot of the actual radiologists may not be comfortable with all the nuances of the disease and might be inadequate in the good aptitude to make a profoundly precise determination. Along these lines, this robotized device can fill in as an aide for those on the bleeding edge of this examination.

We might want to re-underline that we are not proposing the utilization of the proposed model as an option in contrast to the regular analytic tests for COVID-19 disease, yet as an emergency, apparatus to decide the appropriateness of a patient with SARI to go through the test for COVID-19.

Nonetheless, we note that both the model and this report just catch our present comprehension of this rapidly advancing issue, that too on exceptionally restricted information right now accessible. We will continue to refresh the model and this report to improve results.

## 2 RELATED WORK

### COVID-19 detection using CT Scan-

Since the recent sudden wave of COVID-19 infections worldwide, many alternative screening practices have been developed to identify suspected cases of novel coronavirus. However, there are only limited open-source applications available that use CT scans. Publicly available data on CT scans for COVID-19 is also limited. The convolutional neural network architecture can also be trained on CT-Scan image data, separate from the X-ray data upon which training occurred, initially, apart from the latest Coronavirus training sequence on Covid19 data. We treat the proposed convolutional neural network models as our baseline, comparing our results with them

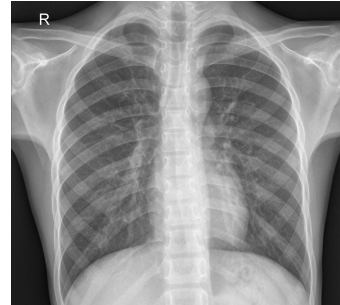
## 3 METHOD

We are using Convolutional Neural Networks(CNN) to train our model. It is a type of artificial neural network in Deep learning. It takes the image's raw pixel data and extracts features for comparison in classification.

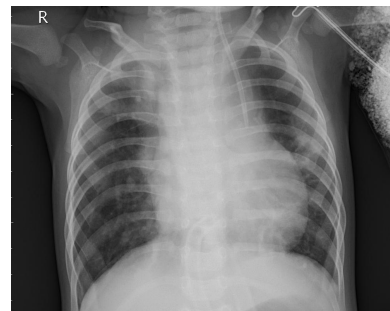
First, we train our model using VGG-16 architecture by dividing it into 3 datasets namely test, training, and validation. We apply convolution Filters first and extract useful features followed by max-pooling and ReLU activation function. Then we add 3 fully connected layers which apply flattening on the previous layers and neurons are connected to previous as well as output layers. We use Keras ImageDataGenerator() and

.flow\_from\_directory() to put the data in batches from the directory. When training the model, memory doesn't fall short due to this.

### 3.1 Problem Formulation and Loss Function



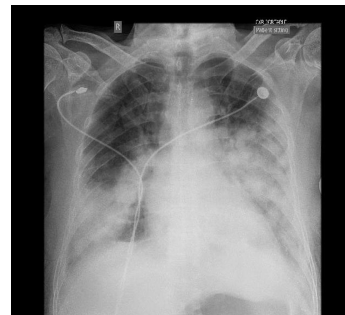
1. Normal



2. Bacterial Pneumonia



3. Viral Pneumonia



4. Covid

Softmax function to make sure the probabilities add up to 1:

$$P(y=j \mid \theta^{(i)}) = \frac{e^{\theta_j^{(i)}}}{\sum_{k=0}^K e^{\theta_k^{(i)}}}$$

where  $\theta = w_0 x_0 + w_1 x_1 + \dots + w_k x_k = \sum_{i=0}^k w_i x_i = w^T x$

softmax function

Error function:

$$E = \frac{1}{n} \sum_k \min_i d(c_i, G_k)$$

where  $d = 0$  if  $c_i = G_k$  else  $d = 1$

$$E = \frac{1}{3} (\min_i d(c_i, G_1) + \min_i d(c_i, G_2) + \min_i d(c_i, G_3))$$

So,

$$E = \frac{1}{3} (0 + 0 + 0)$$

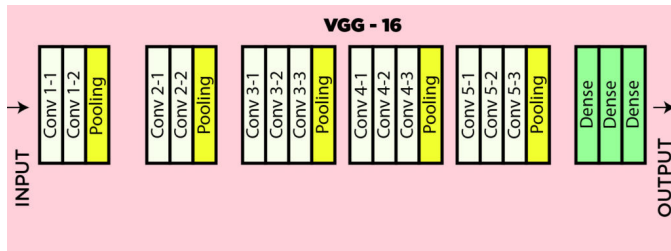
$$E = 0$$

multiplied by a 3x3 portion of the image matrix to get the value of a pixel in the convoluted image. We use multiple filters in this layer to detect different types of features in the input image. We will increase the number of filters as we go deeper into the convoluted layers to extract more complex features. We also used padding such that the resultant image has the same dimensions as the original image. We have also added batch normalization which makes the model faster and more stable. This process is followed by the application of ReLU activation function to increase the non-linearity in the intermediate data and result in faster learning.

$$W_{out} = (W-F+2P)/S+1$$

Convolution Layer Formula [5]

### 3.2 Model Architecture



Our model is based on the VGG-16 CNN Model. It was proposed in 2014, by Simonyan and Zisserman of the Visual Geometry Group Lab of Oxford University. It had won the ImageNet competition in 2014. The detailed description of 3 types of layers of the model namely Convolution Layer, Pooling Layer, and Fully Connected Layer are given as follows:

**Convolution Layer:** This layer is the result of matrix multiplication of a restricted portion of the image and a kernel. The input given to the network is an image of dimensions (150, 150, 3). In our model, a 3x3 kernel with given weights is

$$(1) \mu = \frac{1}{n} \sum_i Z^{(i)} \quad (2) \sigma^2 = \frac{1}{n} \sum_i (Z^{(i)} - \mu)^2$$

$$(3) Z_{norm}^{(i)} = \frac{Z^{(i)} - \mu}{\sqrt{\sigma^2 - \epsilon}} \quad (4) \tilde{Z} = \gamma * Z_{norm}^{(i)} + \beta$$

Batch normalization:

**Pooling Layer:** The pooling layer comes next in line after the application of ReLU activation function on the convolution layer. It helps in the reduction of the spatial size of the representation so as to reduce the amount of computation and parameters. We are using a 2x2 max pooling filter with stride=(2,2). This results in dimensionality reduction in the output image.

$$W_{out} = (W-F)/S+1$$

Pooling Layer Formula [5]

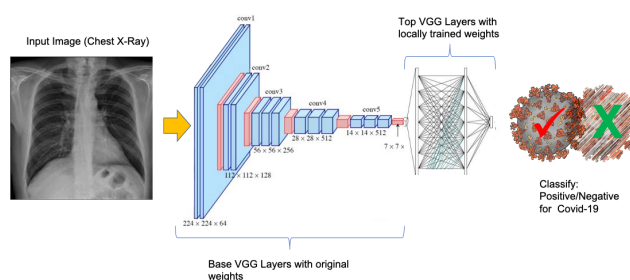
**Fully Connected Layer:** This layer takes the input image from the previous layer and flattens it to form a vector. This vector undergoes a few more FC layers where a few mathematical computations take place and the classification process begins. This is followed by a dropout layer where a few neurons are dropped during training to overcome the problem of overfitting. In our model, a dropout of 0.2 means that 20% of nodes are dropped randomly from the neural network.

Layer (type)	Output Shape	Param #
ImageInput (InputLayer)	(None, 224, 224, 3)	0
conv2d_5 (Conv2D)	(None, 224, 224, 16)	448
conv2d_6 (Conv2D)	(None, 224, 224, 16)	2320
max_pooling2d_6 (MaxPooling2	(None, 112, 112, 16)	0
separable_conv2d_9 (Separab	(None, 112, 112, 32)	688
separable_conv2d_10 (Separab	(None, 112, 112, 32)	1344
batch_normalization_5 (Batch	(None, 112, 112, 32)	128
max_pooling2d_7 (MaxPooling2	(None, 56, 56, 32)	0
separable_conv2d_11 (Separab	(None, 56, 56, 64)	2400
separable_conv2d_12 (Separab	(None, 56, 56, 64)	4736
batch_normalization_6 (Batch	(None, 56, 56, 64)	256
max_pooling2d_8 (MaxPooling2	(None, 28, 28, 64)	0
separable_conv2d_13 (Separab	(None, 28, 28, 128)	8896
separable_conv2d_14 (Separab	(None, 28, 28, 128)	17664
batch_normalization_7 (Batch	(None, 28, 28, 128)	512

max_pooling2d_9 (MaxPooling2	(None, 14, 14, 128)	0
dropout_6 (Dropout)	(None, 14, 14, 128)	0
separable_conv2d_15 (Separab	(None, 14, 14, 256)	34176
separable_conv2d_16 (Separab	(None, 14, 14, 256)	68096
batch_normalization_8 (Batch	(None, 14, 14, 256)	1024
max_pooling2d_10 (MaxPooling	(None, 7, 7, 256)	0
dropout_7 (Dropout)	(None, 7, 7, 256)	0
flatten_2 (Flatten)	(None, 12544)	0
dense_4 (Dense)	(None, 512)	6423040
dropout_8 (Dropout)	(None, 512)	0
dense_5 (Dense)	(None, 128)	65664
dropout_9 (Dropout)	(None, 128)	0
dense_6 (Dense)	(None, 64)	8256
dropout_10 (Dropout)	(None, 64)	0
Total params: 6,639,648		
Trainable params: 6,638,688		
Non-trainable params: 960		

Table: Model Summary

### 3.1 TRAINING



### 4 DATASET AND EVALUATION

Three categories of datasets have been used

1. Covid Positive X-Ray Image Dataset- <https://github.com/ieee8023/covid-chestxray-dataset>
2. Kaggle Pneumonia X-Ray Chest Images - <https://www.kaggle.com/paultimothymooney/chest-xray-pneumonia>
3. Kaggle X-Ray Normal Chest Images - <https://www.kaggle.com/paultimothymooney/chest-xray-pneumonia>

VGG16 CNN model used to predict pneumonia from chest X-ray scans  
<https://github.com/JohnChangUK/Pneumonia-Kaggle>

In the dataset for covid detection, there are 1243 normal and 830 covid positive x-ray images for training and 192 normal and 122 covid positive x-ray images for testing. Total covid positive x-ray images are 952 (830+122). Total normal x-ray images are 1435 (1243+192).

In the dataset for pneumonia detection, there are 1342 normal and 3876 pneumonia positive x-ray images for training, 9 normal and 9 pneumonia positive x-ray images for validation, and 234 normal and 390 pneumonia positive x-ray images for testing. Total pneumonia positive x-ray images are 4275 (3876+9+390). The total normal x-ray images are 1585(1342+9+234).

Approximately 10-15% of the data is used for testing.

	Normal	Pneumonia		Total
		Bacterial	Viral	
Training	1342	2530	1337	5209
Validation	9	4	5	18
Testing	234	242	148	624

Table1: Data Split for Pneumonia detection

	Normal	Covid	Total
Training	1243	830	2073
Testing	192	122	314

Table2: Data Split for Covid detection

## 5 RESULTS

The problem has been classified as binary.



Fig: Confusion Matrix for Pneumonia

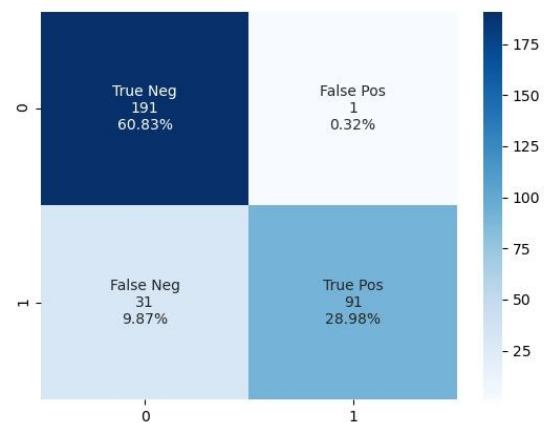


Fig: Confusion Matrix for Covid

An accuracy of 89% is obtained in our binary classification. Detailed results for Sensitivity, Positive Predictive Value (PPV or Precision), and F1-score are given in the tables below.

	Pneumonia vs Normal	Covid vs Normal
Accuracy	89.102564%	89.808917%
Precision	92.819148%	98.913043%
Recall	89.487179%	74.590163%
Specificity	88.461538%	99.479166%
F1-score	91.122715	85.046728

Evaluation Matriz

## 6 CONCLUSION

Using a deep-learning model, we have presented some initial results on detecting COVID-19 positive cases from them. It can be a valuable tool in evaluating the individual disease burden. We plan to validate our approach further using larger COVID-19 X-ray image datasets and clinical trials.

Since the beginning of the pandemic, the role of chest imaging has grown extensively. The role of CT imaging will be required to be reassessed as different strains of COVID-19 continue to emerge. Finally, it is anticipated that artificial intelligence will likely play an ever-larger role over a range of COVID-19-related issues in CT imaging.

## REFERENCES

1. Bennett, J.M.: Smart ct scan based covid19 virus detector. [https://github.com/JordanMicahBennett/SMART-CT-SCAN-BASED-COVID19 VIRUS DETECTOR](https://github.com/JordanMicahBennett/SMART-CT-SCAN-BASED-COVID19-VIRUS-DETECTOR) (2020)
2. Huang, G., Liu, Z., van der Maaten, L., Weinberger, K.Q.: Densely connected convolutional networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2017)
3. <https://towardsdatascience.com/applying-artificial-intelligence-techniques-in-the-development-of-a-web-app-for-the-detection-of-9225f0225b4>
4. Petsiuk, V., Das, A., Saenko, K.: Rise: Randomized input sampling for explanation of black-box models. In: Proceedings of the British Machine Vision Conference (BMVC) (2018)
5. <https://towardsdatascience.com/convolutional-neural-networks-explained-9c5188c4939>
6. Ranjan, E., Paul, S., Kapoor, S., Kar, A., Sethuraman, R., Sheet, D.: Jointly learning convolutional representations to compress radiological images and classify thoracic diseases in the compressed domain (12 2018). <https://doi.org/10.1145/3293353.3293408>
7. Ruiz, P.: Understanding and visualizing densenets. <https://towardsdatascience.com/understanding-and-visualizing-densenets-7f688092391a> (2018)
8. Wang, L., Wong, A.: Covid-net: A tailored deep convolutional neural network design for detection of covid-19 cases from chest radiography images (2020)
9. [Medical Image Analysis - Journal - Elsevier](#)
10. [A Gentle Introduction to Pooling Layers for Convolutional Neural Networks \(machinelearningmastery.com\)](#)