International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 07 Issue: 05 | May - 2023 SJIF 2023: 8.176 ISSN: 2582-3930

Pneumonia and COVID-19 Detection on Chest X-Ray Images using Improved CNN

Arshina Momin[1], Khushbu Patil[2], Meghana Joshi[3], Parimal Dhake[4] and Prof. Pranjali Ghode[5]

Department of Computer Engineering
Genba Sopanrao Moze College of Engineering, Balewadi, Pune-45

---***---

Abstract -The extreme lung conditions pneumonia and Covid-19, tend to affect one or both lungs, frequently brought on via viruses, fungus, or bacteria. Primarily based at the x-rays we've, we can be able to identify this lung infection. Chest X-rays dataset is taken from Kaggle which incorporate numerous x-rays photographs distinguished by means of 3 classes "Pneumonia", "Covid-19" and "Normal". Our aim is developing a deep learning model which can detect the lung disorder. In this project we are using a deep learning model Improved CNN with backbone architecture Densenet-121.Within the healthcare area, ailment detection is essential because early identification and specific analysis can appreciably beautify affected person outcomes. But, conventional methods to illness detection can be exertions-in depth, pricey, and liable to mistakes. Deep learning has emerged as a viable solution to these problems. Deep learning algorithm can aid healthcare workers in detecting COVID-19 with minimal processing of chest X-ray images. In this study, 3-class datasets were created which included COVID-19, pneumonia and normal images obtained from open sources. COVID-19 and viral pneumonia CXR images contain similar features which are challenging for the radiologist to interpret. However, the CNN model can easily learn the features in just a few epochs of training and classify the images correctly. The high accuracies obtained suggest that the deep learning models could find something distinctive in the CXR images and that makes the deep networks capable of distinguishing the images correctly. These trained models can effectively reduce the workload of medical practitioners and increase the accuracy and efficiency of COVID-19 diagnosis.

Keywords**:** Deep Learning, Healthcare, CNN, Densenet121, Covid-19.

1.INTRODUCTION

The novel coronavirus of 2019, or simply known as the COVID-19, affects the respiratory tracts and the lungs leading to severe cases of pneumonia. The usual symptoms include fever, dry hack cough, body ache, and loss of taste or smell. In extreme cases, the patient may experience shortness of breath and multiple organ failure and may lead to fatality.

While the world pharmaceutical companies are trying to develop vaccination to prevent the spread of this pandemic, the current medical practice to control the spread of COVID-19 is focused on early detection and isolation of the patient. The current gold standard for COVID-19 detection is the real-time reverse transcription-polymerase chain reaction (RT-PCR), where the short sequences of DNA or RNA are reproduced or amplified and analysed.

There are two types of transfer learning in the context of deep learning, which are feature extraction and fine-tuning. In the feature extraction technique, a pretrained model on some standard dataset such as ImageNet is used, but the top layer, which is used for classification purpose, will be removed. Then on top of the pretrained model, it trains a new classifier to perform classification. The pretrained model without the top classifier is treated as an arbitrary feature extractor in order to extract useful features from the new dataset. In the second approach which is fine-tuning, the pretrained model weights are treated as the initial values for the new training, and they are updated and adjusted in the training process. In this case, the weights are fine-tuned from generic feature.

In view of the above defined objectives, the key contributions of this research work can now be summarized as follows.

- Review of the most recent work related to the COVID-19 AI-based detection techniques using patient's chest X-ray images.
- Description of the proposed multiclass classification model to classify dataset instances considering the following four image categories: (1) COVID-19 positive instances, (2) Normal instances, and (4) Viral Pneumonia instances.
- Parameter optimization of various Deep Learning models using transfer learning techniques leading to high accuracy classification performance results.

- Using Enhancement and Augmentation techniques on the published dataset describing COVID-19 X-ray patient images.
- Performance analysis of the proposed models as well as a comparative study with existing X-ray image classification models.

1.1. OVERVIEW

The latest COVID-19 AI-based detection models to classify X-ray/CT scan chest images. The Convolutional Neural Networks as a Deep Learning approach. The proposed methodology of the multiclass COVID-19 classification approach is presented. We have made use of Dense Net121 along with improved CNN as our algorithm.

1.2. PROBLEM STATEMENT

Detection of Pneumonia and Covid 19 in Chest X-Ray Images using Deep Learning. The goal of this project is to detect the disease in the given chest X-Ray using the backbone architecture Densenet121 and algorithm CNN stating whether it is affected by pneumonia or covid 19.

2. BODY OF PAPER

2.1. DATASET

COVID-19 & Pneumonia Detection Dataset: - Chest X-ray

The dataset is 2.3GB in size.

Images of patients with and without pneumonia are included in the dataset, along with corresponding metadata such patient age and gender. This collection of chest X-ray images can be used to identify COVID-19 and pneumonia.

There are 5144 total trained images (1266 normal, 460 covid, and 3418 pneumonia).

There are 1288 total photos tested (317 for normal, 116 for covid, and 855 for pneumonia).

There are 6,432 images in all.

2.1.1 STATEMENT OF SCOPE

The scope of the product specifies what it will and will not be, as well as what it will and will not do and contain. As an additional input parameter, the number of layers in the neural networks should be chosen appropriately in order to create the initial weights.

The user's input percentage data determines the testing and training data, and the neural networks must be trained using that data.

We have split our dataset into train and test data in 80s% and 20% respectively.

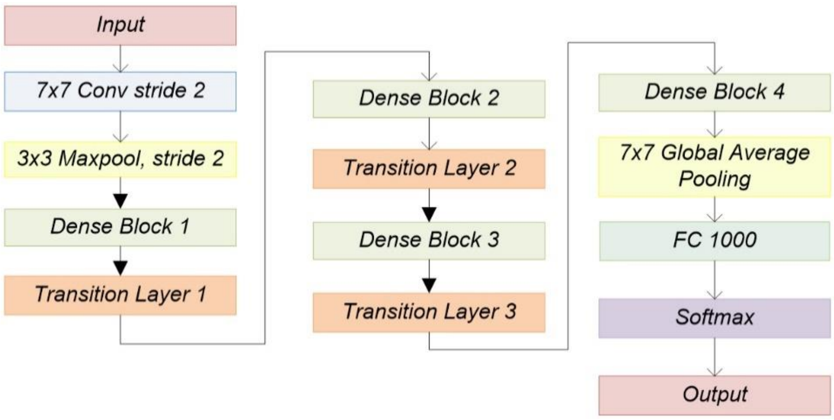
2.2 SYSTEM ARCHITECTURE

2.2.1 BACKBONE ARCHITECTURE – DENSENET-121

Dense Nets are a type of Convolutional Neural Network (CNN) that modify the standard CNN architecture to resolve the vanishing gradient problem that arises as the number of layers in the CNN increases. In a traditional feed-forward CNN, each convolutional layer except the first one receives the output of the previous convolutional layer and produces an output feature map that is then passed on to the next convolutional layer. However, as the number of layers in the CNN increases, certain information can get lost, reducing the ability of the network to train effectively. Dense Nets resolve this problem by simplifying the connectivity pattern between layers. In a Dense Net architecture, each layer is connected directly with every other layer, hence the name Densely Connected Convolutional Network. For 'L' layers, there are L(L+1)/2 direct connections.

Dense Nets have several advantages over traditional CNNs. They alleviate the vanishing-gradient problem, strengthen feature propagation, encourage feature reuse, and substantially reduce the number of parameters. By connecting each layer to every other layer in a feed-forward fashion, Dense Nets require fewer parameters than an equivalent traditional CNN, as there is no need to learn redundant feature maps. Dense Nets also improve the flow of information and gradients throughout the network, making them easy to train.

Layers	Output Size	DenseNet-121	DenseNet-169	DenseNet-201	DenseNet-264
Convolution	112 × 112	7 × 7 conv, stride 2			
Pooling	56 × 56	3 × 3 max pool, stride 2			
Dense Block (1)	56 × 56	[1 × 1 conv; 3 × 3 conv] × 6	[1 × 1 conv; 3 × 3 conv] × 6	[1 × 1 conv; 3 × 3 conv] × 6	[1 × 1 conv; 3 × 3 conv] × 6
Transition Layer (1)	56 × 56	1 × 1 conv			
	28 × 28	2 × 2 average pool, stride 2			
Dense Block (2)	28 × 28	[1 × 1 conv; 3 × 3 conv] × 12	[1 × 1 conv; 3 × 3 conv] × 12	[1 × 1 conv; 3 × 3 conv] × 12	[1 × 1 conv; 3 × 3 conv] × 12
Transition Layer (2)	28 × 28	1 × 1 conv			
	14 × 14	2 × 2 average pool, stride 2			
Dense Block (3)	14 × 14	[1 × 1 conv; 3 × 3 conv] × 24	[1 × 1 conv; 3 × 3 conv] × 32	[1 × 1 conv; 3 × 3 conv] × 48	[1 × 1 conv; 3 × 3 conv] × 64
Transition Layer (3)	14 × 14	1 × 1 conv			
	7 × 7	2 × 2 average pool, stride 2			
Dense Block (4)	7 × 7	[1 × 1 conv; 3 × 3 conv] × 16	[1 × 1 conv; 3 × 3 conv] × 32	[1 × 1 conv; 3 × 3 conv] × 32	[1 × 1 conv; 3 × 3 conv] × 48
Classification Layer	1 × 1	7 × 7 global average pool			
		1000D fully-connected, softmax			



2.2.2 ALGORITHM

CONVOLUTIONAL NEURAL NETWORK (CNN)

A deep learning approach known as a convolutional neural network (CNN) is particularly effective at processing and recognising images. Convolutional layers, pooling layers, and completely connected layers are among the layers that make up this structure.

The key part of a CNN is its convolutional layers, where filters are used to extract characteristics like edges, textures, and forms from the input image. The output of the convolutional layers is then sent through pooling layers, which are employed to down-sample the feature maps and retain the most crucial data while lowering the spatial dimensions. One or more fully connected layers are then applied to the output of the pooling layers in order to predict or categorise the image.

Convolutional Neural Networks (CNNs) are a type of deep learning neural network designed for processing structured arrays of data such as images. CNNs are trained using a large dataset of labelled images, where the network learns to recognize patterns and features that are associated with specific objects or classes. Once trained, a CNN can be used to classify new images, or extract features for use in other applications such as object detection or image segmentation. CNNs have achieved state-of-the-art performance on a wide range of image recognition tasks, including object classification, object detection, and image segmentation. They are widely used in computer vision, image processing, and other related fields, and have been applied to a wide range of applications, including self-driving cars, medical imaging, and security systems. The convolutional layer is the core building block of a CNN, and it is where the majority of computation occurs. It requires a few components, which are input data, a filter, and a feature map. A basic CNN can be viewed as a series of convolutional layers, followed by an activation function, followed by a pooling (downscaling) layer, repeated many times. With the repeated combination of these operations, the first layer detects simple features such as edges in an image, and the second layer begins to detect higher-level features. By the tenth layer, a CNN is able to detect more complex shapes such as eyes.

Following are the layers in the Convolutional Neural Networks:

Convolutional Layer

The central component of a CNN is the convolutional layer, which is also where the majority of computation takes place. It needs input data, a filter, and a feature map, among other things. Assume that the input will be a colour image that is composed of a 3D pixel matrix. As a result, the input will have three dimensions—height, width, and depth—that are analogous to RGB in an image. Additionally, we have a feature detector, also referred to as a kernel or filter, which will move through the image's receptive fields and determine whether the feature is there. Convolution describes this process.

A two-dimensional (2-D) array of weights serving as the feature detector represents a portion of the image. The filter size, which also controls the size of the receptive field, is normally a 3x3 matrix, however they can vary in size. Following the application of the filter to a portion of the image, the dot product between the input pixels and the filter is determined. The output array is then fed with this dot product. Once the kernel has swept through the entire image, the filter shifts by a stride and repeats the operation. A feature map, activation map, or convolved feature is the ultimate result of the series of dot products from the input and the filter.

After each convolution operation, a CNN applies a Rectified Linear Unit (ReLU) transformation to the feature map, introducing nonlinearity to the model.

Pooling Layer

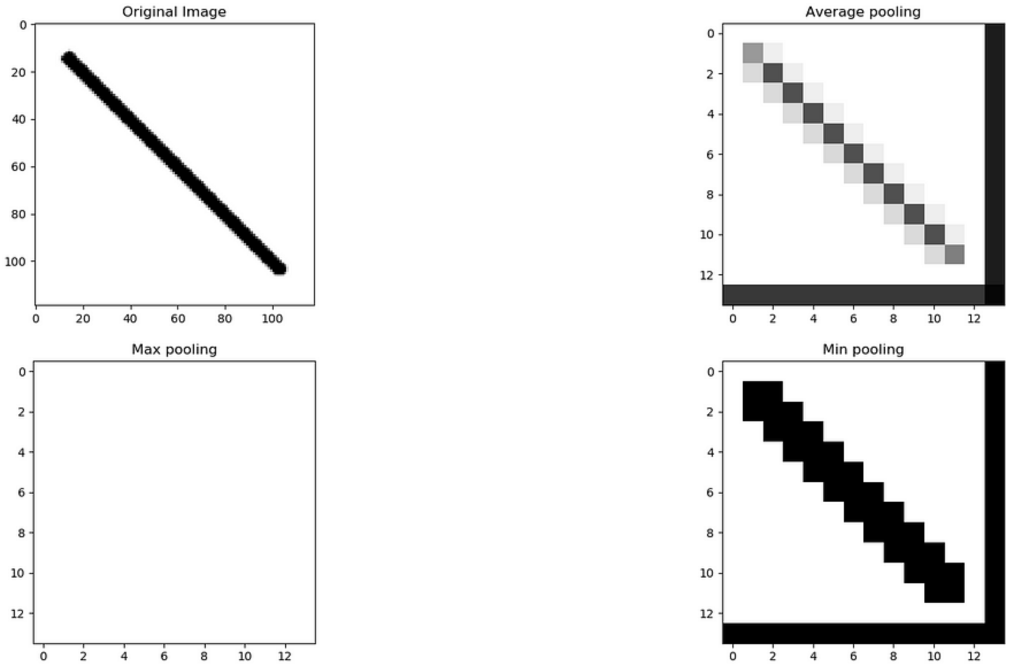
Pooling layers, also known as down sampling, carries out dimensionality reduction and lowers the number of parameters in the input. The pooling operation sweeps a filter across the entire input similarly to the convolutional layer, with the exception that this filter lacks weights. Instead, the kernel populates the output array by applying an aggregation function to the values in the receptive field. There are principally two forms of pooling:

Max pooling: The filter chooses the pixel with the highest value to send to the output array as it advances across the input. As a side note, this method is applied more frequently than average pooling. Max pooling selects the brighter pixels from the image. It is useful when the background of the image is dark and we are interested in only the lighter pixels of the image.

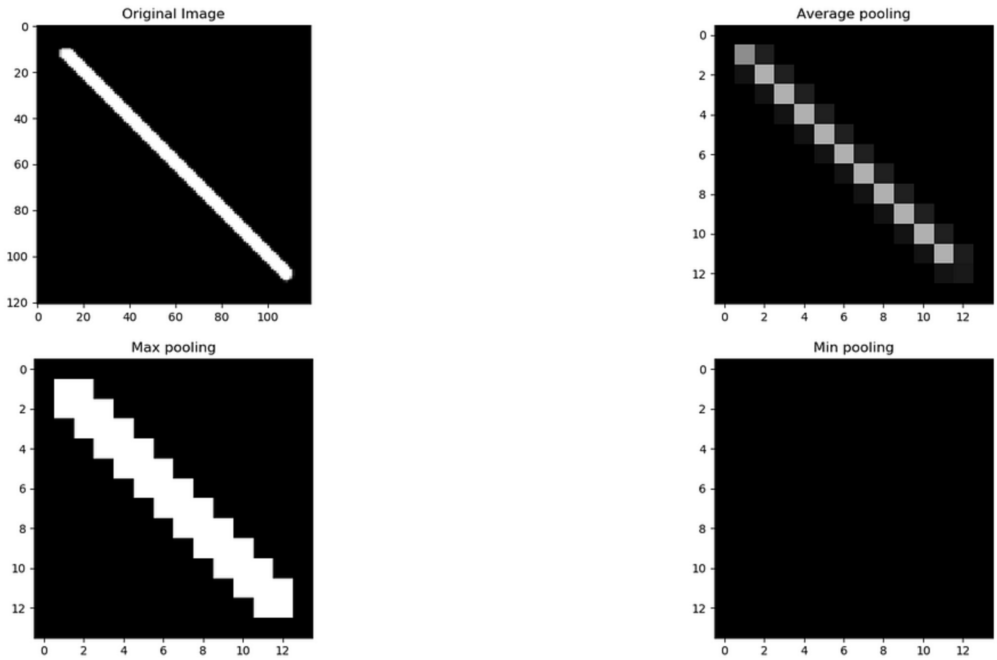
Average pooling: The filter computes the average value in the receptive field as it passes over the input to send to the output array. Average pooling method smooths out the image and hence the sharp features may not be identified when this pooling method is used.

Min pooling: The minimum pixel value of the batch is selected. Min pooling is opposite of max pooling. Min pooling selects the darker pixels from the image. It is useful when the background of the image is bright and we are interested in only the darker pixels of the image.
Pooling Layer offers the CNN several advantages. They lessen complexity, increase effectiveness, and lower the risk of overfitting.

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 07 Issue: 05 | May - 2023 SJIF 2023: 8.176 ISSN: 2582-3930



Min pooling gives better result for images with white background and black object

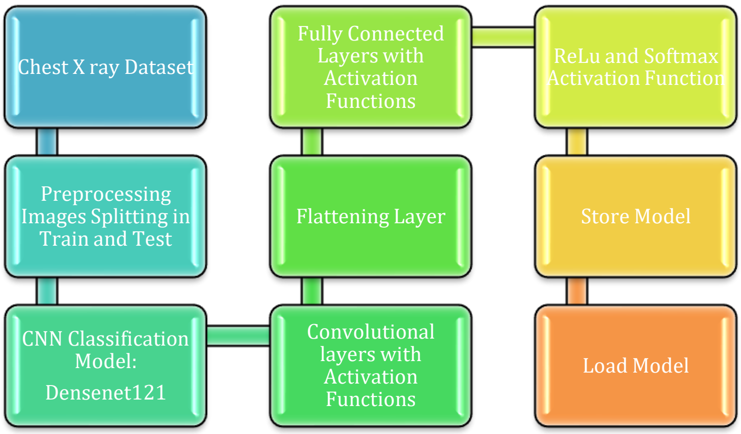


Max pooling gives better results for the images with a black background and white object

Fully-Connected Layer

The full-connected layer is exactly what its name implies. As was already noted, partially connected layers do not have a direct connection between the input image's pixel values and the output layer. In contrast, every node in the output layer of the fully-connected layer is directly connected to a node in the layer above it.

Based on the features that were retrieved from the preceding layers and their various filters, this layer conducts the classification operation. FC layers often utilize a SoftMax activation function to categorize inputs appropriately, producing a probability ranging from 0 to 1. Convolutional and pooling layers typically use ReLU functions.



3. RESULT AND CONCLUSION

In this study, a model based on densenet-121 backbone architecture and convolutional neural networks was presented to identify pulmonary diseases such as pneumonia and covid-19 from chest X-ray images. With this model, 95.4193% accuracy is attained. In addition to the proposed model's performance, this study adds to the body of related literature by providing a model with notably few parameters. Due to this benefit, this model can be used in hospitals and other locations without access to powerful computing equipment. Additionally, the system may be utilized by medical staff without technical computer knowledge.

```
In [11]: def test_model(model, test_data_loader):
             correct = 0
             total = 0
             with torch.no_grad():
                 for data in test_data_loader:
                     images, labels = data
                     images, labels = images.to('cuda'), labels.to('cuda')
                     outputs = model(images)
                     _, predicted = torch.max(outputs.data, 1)
                     total += labels.size(0)
                     correct += (predicted == labels).sum().item()
             print(f"Accuracy of the network on the test images: {float(correct)/float(total)*100:.4f}")

         test_model(model, test_data_loader)

         Accuracy of the network on the test images: 95.4193
```

4. FUTURE SCOPE

The quality of the chest X-ray images utilized in this work can be improved using image processing methods like Contrast Limited Adaptive Histogram Equalisation (CLAHE) in future research. Additionally, by considering the class-specific correct detection rates of various classifier models, ensemble approaches may be used. Adding fresh photos to the Pneumonia and covid-19 classes can also be taken into consideration to establish data balance and improve the models' capacity to derive superior representations.

ACKNOWLEDGMENT

With immense pleasure, we present the project report as part of the curriculum of the B.E. Computer Engineering. We wish to thank everyone who gave us unending support right from the idea's inception.

We express our sincere and profound thanks to Prof. Pranjali Ghode our Project Guide, who always stood by us as the helping and guiding support, and all those who have directly or indirectly guided and helped us in the preparation of the Project.

REFERENCES

[1] Kanakaprabha. S. and D. Radha , "Analysis of COVID-19 and Pneumonia Detection in Chest X-Ray Images using Deep Learning"(2021)

[2] Sammy V. Militante; Nanette V. Dionisio; Brandon G. Sibbaluca "Pneumonia and COVID-19 Detection using Convolutional Neural Networks**"**(2020)

[3] Muhab Hariri and Ercan Avsar **,** "COVID-19 and pneumonia diagnosis from chest X-ray images using convolutional neural networks"(2023)

[4] Saeed S. Alahmari; Baderaldeen Altazi; Jisoo Hwang; Samuel Hawkins; Tawfiq Salem , **"**A Comprehensive Review of Deep Learning-Based Methods for COVID-19 Detection Using Chest X-Ray Images"(2022)

[5] M. Jagadeeswari; S. Padmapriya; Shri Lalitha G; Suvetha S.; Sharmila D , "COVID Detection using Deep Learning"(2022)

[6] Prajakta Mhatre; Kalyani Bhole , "Prediction of COVID-19 & Pneumonia using Machine Learning & Deep Learning Model"(2022)

[7] Akshat Khare; Pranjal Patel; Suresh Sankaranarayanan; Pascal Lorenz **, "**COVID Pneumonia Prediction Based on Chest X-Ray Images Using Deep Learning"(2022)

[8] Meghna Madhu; Anushka Xavier; N. Jayapandian **, "**Covid-19 Classification and Detection Model using Deep Learning"(2022)

[9] Praneeth Gadipudi; Palachuri Sai Teja; Chandana Sowmya. Yelamancheli; Adith Thaniserikaran; Rayyan Mohiuddin; Ajin Joy ,**"**Detection of pneumonia progression in lungs of individuals affected with covid 19 severely using deep learning techniques"(2022)

[10] Tobin P. Mathew; Surya S. R , "COVID-19 and Pneumonia Detection using Hybrid VGG-16 model using Chest X-rays"(2022)