[Egyptian Informatics Journal 23 (2022) 247–257](https://doi.org/10.1016/j.eij.2022.01.002)

Contents lists available at [ScienceDirect](http://www.sciencedirect.com/science/journal/11108665)

Egyptian Informatics Journal

journal homepage: [www.sciencedirect.com](http://www.sciencedirect.com/)

[](http://crossmark.crossref.org/dialog/?doi=10.1016/j.eij.2022.01.002&domain=pdf)Efficient framework for detecting COVID-19 and pneumonia from chest X-ray using deep convolutional network

Ahmed Salem Musallam [a](#_bookmark0),[⇑](#_bookmark3), Ahmed Sobhy Sherif [b](#_bookmark1),[c](#_bookmark2), Mohamed K. Hussein [b](#_bookmark1)

a *Information Technology Dep., Faculty of information Technology and Computer Science, Sinai University, Egypt*

b *Computer Science Dep., Faculty of Computers and Informatics, Suez Canal University, Egypt*

c *Computer Science Dep., College of Computer Science and Engineering, Taibah University, Kingdom Saudi Arabia*

a r t i c l e i n f o

*Article history:*

Received 11 January 2021

Revised 8 January 2022

Accepted 18 January 2022

Available online 1 February 2022

*Keywords:*

Deep Convolutional Neural Network Chest X-ray images

Coronavirus Pneumonia

a b s t r a c t

Recently, the COVID-19 pandemic is considered the most severe infectious disease because of its rapid spreading. Radiologists still lack sufficient knowledge and experience for accurate and fast detecting COVID-19. What exacerbates things is the significant overlap between Pneumonia symptoms and COVID-19, which confuses the radiologists. It’s widely agreed that the early detection of the infected patient increases his likelihood of recovery. Chest X-ray images are considered the cheapest radiology images, and their devices are available widely. This study introduces an effective Deep Convolutional Neural Network (DCNN) called ‘‘DeepChest” for fast and accurate detection for both COVID-19 and Pneumonia in chest X-ray images. ‘‘DeepChest” runs with a small number of convolutional layers, a small number of max-pooling layers, and a small number of training iterations compared with the recent approaches and the state-of-the-art of DCNN. We conducted the experimental evaluations of the pro- posed approach on a data set with 7512 chest X-ray images. The proposed approach achieves an accuracy of 96.56% overall, 99.40% in detecting COVID-19, and 99.32% in detecting Pneumonia. In actual practice, the presented approach can be used as a computer-aided diagnosis tool to get accurate results in detect- ing Pneumonia and COVID-19 in chest X-ray images.

© 2022 THE AUTHORS. Published by Elsevier B.V. on behalf of Faculty of Computers and Information,

Cairo University. This is an open access article under the CC BY-NC-ND license ([http://creativecommons.](http://creativecommons.org/licenses/by-nc-nd/4.0/)

[org/licenses/by-nc-nd/4.0/](http://creativecommons.org/licenses/by-nc-nd/4.0/)).

1. Introduction

Respiratory system diseases such as Pneumonia and Coron- avirus (COVID-19) are common, infectious, and deadly. The results of radiography are the main factor in the accurate diagnosis of such diseases. Consolidation is a radiological expression that explains the increased lung density within the air spaces. There are different types of lung opacity in a chest radiograph, which radiologists may find due to a pathologic operation that fills in the alveoli with blood, fluid, pus, protein, or cells [[1]](#_bookmark31). Consolidation can help us in detecting many diseases, and specifically, Pneumonia and COVID-

19. Pneumonia is a kind of respiratory system infection.

\* Corresponding author.

Peer review under responsibility of Faculty of Computers and Information, Cairo University.

Furthermore, it registers the highest death rate among contagious diseases and the third reason for death in general. The delay in an accurate diagnosis increases the possibility of death. So, a fast detection is very important [[1–5]](#_bookmark31). COVID-19 is an infectious dis- ease that affects the respiratory system. Respiratory problems are considered the main COVID-19 symptom. Consequently, a chest X-ray can show an early detection of COVID-19 [[6]](#_bookmark19). The use of arti- ficial intelligence technology in automatic detection of consolida- tion in chest radiography became one of the interesting topics in medical research [[7]](#_bookmark19). Researchers applied deep learning technology to a wide range of domains in science, engineering, and medicine [[8]](#_bookmark19). Since 2012, Researches widely use a part of deep learning technology, called deep convolutional neural network (DCNN), and achieved great success in image classification [[9]](#_bookmark19). Recently, the DCNNs also achieved promising results in the

**Production and hosting by Elsevier**

<https://doi.org/10.1016/j.eij.2022.01.002>

1110-8665/© 2022 THE AUTHORS. Published by Elsevier B.V. on behalf of Faculty of Computers and Information, Cairo University. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).

medical field [[10–13]](#_bookmark19). [Fig. 1](#_bookmark4) shows the phases of the proposed ‘‘DeepChest” diagnosis system. Now, we will mention some outli- nes about the proposed model:

* The proposed approach uses a three-step pre-processing method that enhances the quality of X-ray images via removing

the noisy and confusing variables, eliminates histogram differ- ences between X-ray images, and improves their contrast.

* A judgment applied to our pre-processing approach via measur-

ing the image quality by Blind Reference less Image Spatial

Quality Evaluator (BRISQUE) [[14]](#_bookmark19).

* A balanced dataset was generated based on Pneumonia chest X- ray dataset [[15]](#_bookmark20), COVID-19 chest X-ray dataset [[16]](#_bookmark21), and

COVID-XRay-5 K dataset [[17]](#_bookmark22) in order to train the proposed problem-based deep convolutional neural network model ‘‘DeepChest”.

* The proposed diagnosis system ‘‘DeepChest” is applied to clas-

sify chest X-ray images as normal, pneumonia, or COVID-19,

and are compared with VGG16 [[18]](#_bookmark24), Mohammad et al. [[19]](#_bookmark25), DenseNet-121 [[20]](#_bookmark26), and MobileNet [[21]](#_bookmark27).

* Batch Normalization layers [[22]](#_bookmark28) are used in ‘‘DeepChest” to help

us train the network faster, get a higher learning rate, and

enable initializing weights easier.

* An analytical comparison experiment of Pneumonia and COVID- 19 detection between the proposed model ‘‘DeepChest” and the

well-known approaches like VGG16 [[18]](#_bookmark24), DenseNet-121 [[20]](#_bookmark26), MobileNet [[21]](#_bookmark27), and the recent approaches like Mohammad

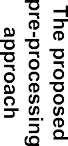


Fig. 1. The proposed ‘‘DeepChest” diagnosis system.

et al. [[19]](#_bookmark25) was applied.

We organized the rest of the paper as follows: Section 2 dis- cusses the related work, Section 3 describes the proposed three-step pre-processing approach. In Section 4, we surveyed the associated models and presented the proposed model ‘‘DeepChest”. Section 5 discusses the results. Section 6 illus- trates a performance comparison between well-known DCNNs and the proposed model. Finally, Section 7 presents the con- cluding remarks.

1. Related work

Yaniv et al. [[23]](#_bookmark29) discussed the power of deep learning tech- niques, especially convolutional neural networks, for consolidation in the detection of chest radiographs. Still, they used a small num- ber of chest radiographs in their practical experiments. Kai-lung et al. [[24]](#_bookmark30) used the context of nodule classification in their proposed convolutional neural network model to classify com- puted tomography images. But, they tend to detect lung cancer, and their achieved accuracy is not enough to be used as a computer-aided system. Mohammad et al. [[25]](#_bookmark32) proposed an ensemble convolutional neural network model to localize the anomalies in chest X-ray images. John et al. [[26]](#_bookmark33) introduced a DCNN model for classifying chest X-ray images as Normal or Pneu- monia images. They declared that the confusing variables like strings in the left and right corner could destroy the general perfor- mance of the DCNN models, so it is not possible to get favorable outputs in classification problems according to the irrelated fea- tures. Shuaijing et al. [[27]](#_bookmark34) proposed a hierarchical deep CNN model to rank chest X-ray images in normal images and anomaly images. Sergio et al. [[28]](#_bookmark35) proposed a novel approach for classifying COVID- 19 in chest X-ray images based on the texture features and neural network. Their method used gray level co-occurrence matrix, other texture operators, and the uniform pattern values of the local bin- ary patterns to extract chest X-ray image features. Still, This is a lousy descriptor compared to convolutional features. Moreover, they used a small number of COVID-19 chest X-ray images in their experiments. A.Jaiswal et al. [[29]](#_bookmark36) proposed DenseNet201 based on deep transfer learning as a pre-trained deep learning model with the ImageNet dataset to detect COVID-19 in chest CT. However, they accomplished the testing process with a small number of images, almost 374 for Normal and Covid-19. The training process has a large footprint with many computations presented in repeat- ing the training phase up to 300 times to enhance accuracy. J.Civit- Masot et al. [[30]](#_bookmark37) proposed an approach with pre-processing steps applied with the VGG16 deep learning model. However, the evalu- ation process was done with only 80 chest X-ray images for Nor- mal, Pneumonia, and COVID-19, and the small number of images used in their training process degraded the accuracy. H.Wang et al. [[31]](#_bookmark38) proposed triple attention learning for the classification of 14 thoracic diseases using DenseNet-121 as a backbone deep convolutional neural network, but the large number of classes involved in their approach negatively affected the accuracy of results. S.Minaee [[32]](#_bookmark39) proposed an approach for predicting COVID-19 from chest X-ray images using multi-use deep convolu- tional neural networks along with transfer learning technique. The COVID-19 class has a small number of images as compared to the Normal class. So their training process was unbalanced. Further- more, their evaluation process was insufficient because they used only 100 COVID-19 X-ray images. Hamed et al. [[33]](#_bookmark40) proposed a DCNN model called ‘‘chestnet” for classifying chest X-ray images as Normal or Pneumonia. They discussed different histogram dis- tributions between images included in the understudy dataset (Pediatric Chest X-ray). They reported the critical issue of selecting suitable pre-training datasets. But their DCNN model consists of 14 convolutional layers, which consume large memory space, and

their pre-processing approach achieves a humble effect on the chest X-ray images.Mohammad et al. [[19]](#_bookmark25) proposed a deep convo- lutional model for detecting Pneumonia and COVID-19 in chest X- ray images. They used a small number of COVID-19 chest X-ray images, which creates an imbalanced dataset. There are other problems in the chest X-ray images, such as the noise in these images, the poor quality for X-ray images in general, and the mod- el’s training with different domain datasets issues.

1. The proposed pre-processing approach

In radiology image classification problems, deep learning tries to identify patterns that will help in the classification process. There are many issues in the chest X-ray images dataset facing these models, downgrading the learning phase. Consequently, a wrong or misclassification may happen. In the following, we list the most common issues:

* There are confusing variables that can hugely affect the general performance of DCNN models, such as strings on the left and

right corners of chest X-ray images (first issue).

* Chest X-ray images suffer from noises that downgrade the DCNNs performance in accurately detecting the desirable pat-

terns (second issue).

* Chest X-ray images also have all its details in a small range in the histogram graph, and the contrast of chest X-ray images is

not good enough to accurately detect the right edges of the desired patterns (third issue).

To handle all of the above issues and make an accurate assess-

Table 1

The BRSIQUE score for the original image, NLM Filter, Gaussian Filter and Median Filter.

Noise Filter BRSIQUE Score

Original Image 30.422937041381687

NLM Filter 30.25358515476941

Gaussian Filter 68.95581668738586

Median Filter 44.94117491132553

*NL*[*x*](*i*) for a pixel *i*, is calculated as weighted average for all the image pixels:

*NL*[*x*](*i*)= *w*(*i*, *j*)*x*(*j*) (1)

X

*i*∈*j*

Where the weights {*w*(*i*, *j*)}*j* relies on the similarity of pixel *i* and *j*, and ensure the condition 0 6 *w*(*i*, *j*) 6 1 and *jw*(*i*, *j*)= 1. The NLM algorithm takes the road of producing a clear image with

P

less loss in detail because the NLM filter takes a mean of all image pixels regarding how similar these pixels to the output pixel.

*3.3. Histogram equalization*

In image processing, the common way to get comprehensive information and characteristics of any image is the histogram. The histogram of an image can be considered a vector that includes the frequencies of pixels at every gray level. We assume we have *X* rows and *Y* columns for an X-ray image *p* levels of intensity with

values ranging from 0 to *P* — 1. The histogram *h*(*i*) [[37]](#_bookmark40) can be

defined as:

ment, the following three-step for pre-processing were carried out efficiently on the chest X-ray dataset.

*X*—1 *Y*—1

*h*(*i*)=

XX

*x*=0 *y*=0

d(*f* (*x*, *y*)— *i*), *i* = 0, 1, ... , *P* — 1, where d(*w*)=

1 *w* = 0,

0 otherwise

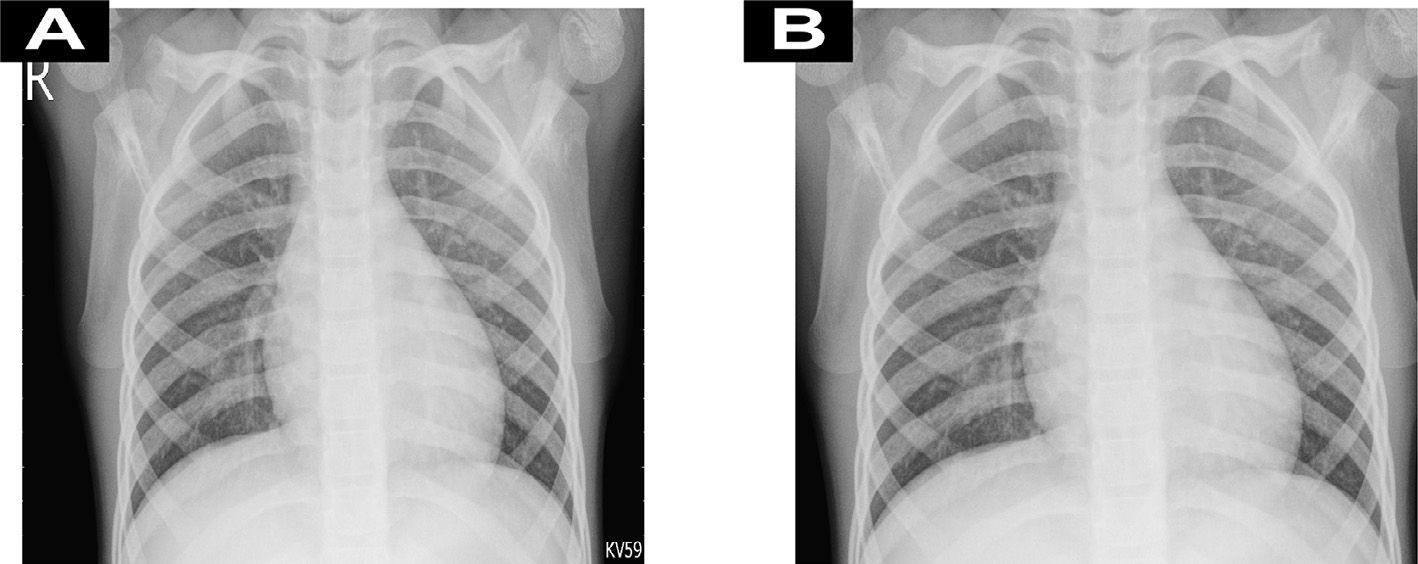
(2)

* 1. *Eliminating the confusing variables*

The confusing variables such as a string on the left and right corners were eliminated via cropping each image with a pre- defined window with a size of 100 pixels from all directions, see [Fig. 2](#_bookmark6).

* 1. *Denoising X-ray images using Non-Local Mean (NLM) Algorithm*

Chest X-ray images suffer from noise that downgrades DCNNs from accurately detecting the desirable patterns, so to cope with this issue we use the NLM algorithm [[34]](#_bookmark40), which enhances the X- ray images as compared with Gaussian [[35]](#_bookmark40) and Median [[36]](#_bookmark40) denoising algorithms according to BRISQUE evaluator see [Table 1](#_bookmark5).

Given a chest X-ray image *x* = {*x*(*i*)|*i* ∈ *I*}, the result value

In histogram equalization [[38]](#_bookmark40), the output histogram is flat, see [Fig. 3](#_bookmark7), it hugely improves the contrast in images, and it performs a separation to the most frequent intensity values, which allows rec- ognizing small details via giving areas of lower contrast to get high contrast. Histogram Equalization solves the interference of the most frequent patterns in the X-ray image and improves the contrast.

*3.4. Judging the presented pre-processing approach using Blind Reference less Image Spatial Quality Evaluator (BRISQUE)*

Chest X-ray images may have many distortions such as blur and noise. So we developed an approach to enhance the chest X-ray dataset images to improve the result of the DCCN model.

We considered the BRISQUE [[14]](#_bookmark19) to assess the quality of the output of our pre-processing approach. The higher the BRISQUE

Fig. 2. Figure A shows An example of chest X-ray images before the cropping process, and Figure B shows the same image after the cropping process.

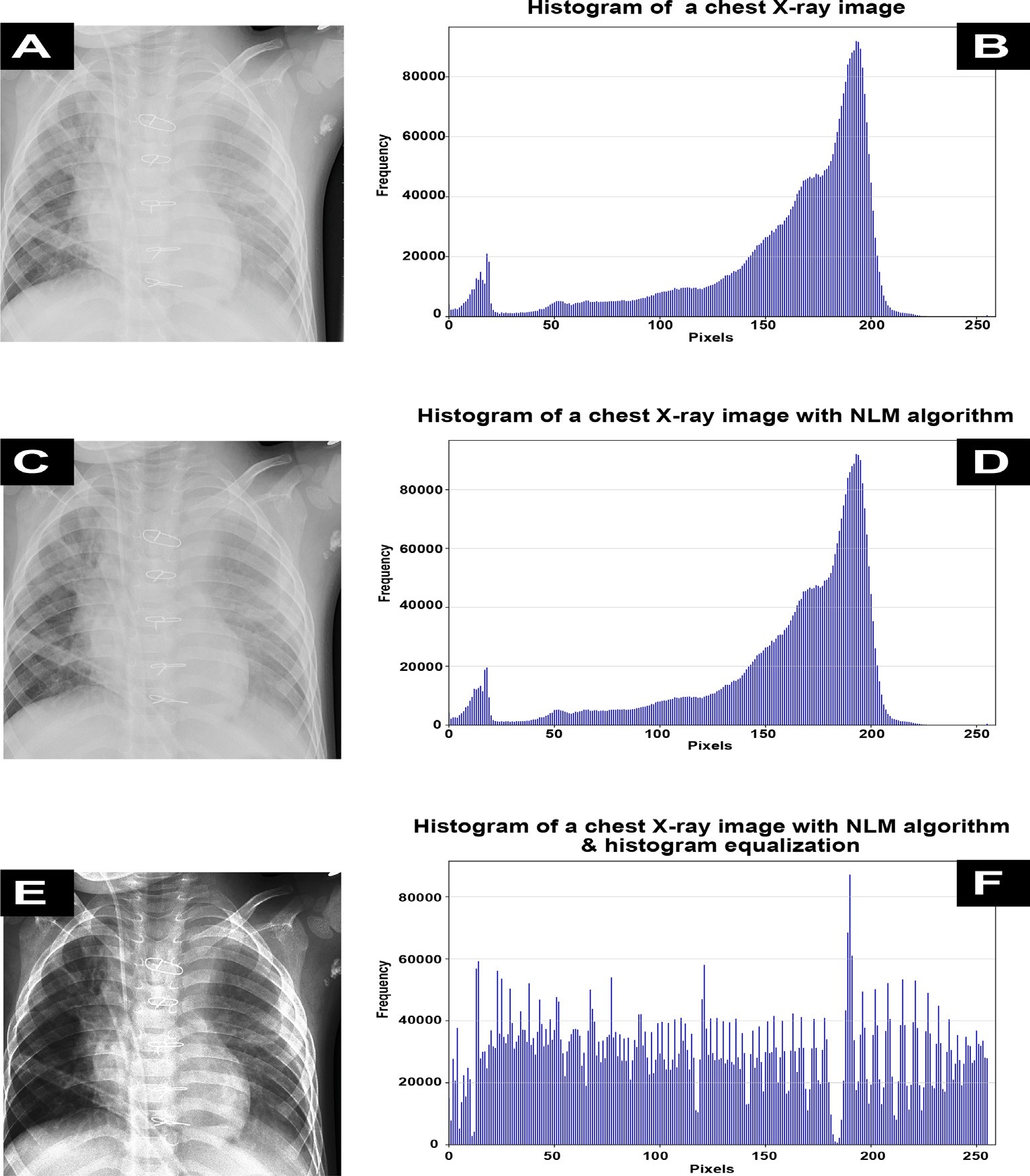


Fig. 3. Figure A shows a sample of Chest X-ray images after the cropping process. Figure B shows the histogram of Figure A. Figure C shows the same image after applying the NLM algorithm. Figure D shows the histogram of Figure C. Figure E shows the Figure D image after applying Histogram Equalization. Figure F shows the histogram of Figure E.

scores, the worst the image, as it is full of noise. BRISQUE was applied in a random chest X-ray image before and after the pre- processing steps applied, and a comparison is applied between our pre-processing approach with Hamed et al. [[33]](#_bookmark40) and we achieve better results. [Table 2](#_bookmark8) shows the image quality scores.

1. The Investigated models and the proposed ‘‘DeepChest model

In this work, we conducted an analytical comparison between different Multi-use models and our proposed problem-based model. The two following sections demonstrate this comparison and the training strategies.

Table 2

The BRSIQUE score for the original image, Hamed et al. [[33]](#_bookmark40) approach and the proposed approach.

Pre-processing Approaches BRSIQUE Score

* 1. *Multi-use models and the proposed model ‘‘DeepChest*”

This section presents an analytical comparison between differ- ent multi-use models and our proposed model ‘‘DeepChest”. We investigated the proposed model, and the multi-use DCNN models were VGG16 [[18]](#_bookmark24), DenseNet-121 [[20]](#_bookmark26), Mohammad et al. [[19]](#_bookmark25), and MobileNet [[21]](#_bookmark27).

The DCNNs contain two main parts in their architecture: convo- lutional parts and classifier parts. The convolutional parts extract the image’s features, and the classifier parts classify these features into one of several predefined classes due to the used dataset. The transfer learning technique [[39]](#_bookmark41) was applied to get pre-trained DCNN models. Therefore, the investigated models [[18–20]](#_bookmark24) were pre-trained on the standard ImageNet dataset [[48]](#_bookmark41). The classifier part of these models was removed and a dense layer with 3 classes and activation ‘‘softmax” [[41]](#_bookmark41) function was added.

[Table 3](#_bookmark9) shows the input parameters of these DCNNs models in the presented work. On the other hand, the ‘‘DeepChest” model is

|  |  |  |
| --- | --- | --- |
| Original Image  Hamed et al. Approach [[33]](#_bookmark40) | 30.422937041381687  25.523768478459573 | proposed as a DCNN problem-based model, see [Fig. 4](#_bookmark10), [Table 4](#_bookmark11)  shows the input parameter for the proposed model and [Table 5](#_bookmark12) |
| The Proposed Approach | 18.209273375260608 | demonstrates the presented model structure. The proposed |

Table 3

The input parameters of the investigated multi-use DCCN models.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Parameters | VGG16 | DenseNet-121 | Mohammad et al. model | MobileNet |
| Image Size | 244 × 244 × 3 | 244 × 244 × 3 | 244 × 244 × 3 | 244 × 244 × 3 |
| Learning Rate | 0.0001 | 0.0001 | 0.0001 | 0.0001 |
| Decay | 1*e* — 5 | 1*e* — 5 | 1*e* — 5 | 1*e* — 5 |
| Batch Size | 16 | 16 | 16 | 16 |
| Step Per Epoch | 385 | 385 | 385 | 385 |
| Validation Steps | 1 | 1 | 1 | 1 |
| Optimizer | Adam | Adam | Nadam | Adam |
| Call Backs | Model Checkpoint and CSV | Model Checkpoint and CSV | Model Checkpoint and CSV | Model Checkpoint and CSV |
| Loss | Logger  Categorical Cross Entropy | Logger  Categorical Cross Entropy | Logger  Categorical Cross Entropy | Logger  Categorical Cross Entropy |
| Class Weight | Auto | Auto | Auto | Auto |
| Pre-training Weights | ImageNet Weights | ImageNet Weights | ImageNet Weights | ImageNet Weights |
| Activation Function Of The Last | SoftMax | SoftMax | SoftMax | SoftMax |

Classifier Layer

‘‘DeepChest” model has many important differences from the com- pared models as follows:

* To speed up the X-ray images’ diagnosis, we must decrease the size of the output of the convolution layer, Max-pooling layer

was used. But it can result in losing some features from chest X- ray images. Well-known DCNN models such as VGG16 [[18]](#_bookmark24), Mohammad et al. [[19]](#_bookmark25),MobileNet [[21]](#_bookmark27), and DenseNet-121 [[20]](#_bookmark26) use 6, 9, 1, and 20 max-pooling layers [[18–20]](#_bookmark24) respectively. On the other hand, the proposed ‘‘DeepChest” model uses only 4 max-pooling layers. Fortunately, the proposed three-step pre- processing approach worked to make ‘‘Deep Chest” more speed and moreaccurateusinga suitablenumberof max-pooling layers.

* + - The well-known DCNN models, especially VGG16 [[18]](#_bookmark24), Moham- mad et al. [[19]](#_bookmark25), MobileNet [[21]](#_bookmark27), and DenseNet121 [[20]](#_bookmark26) were

the convolutional operation with 3 × 3 filter size to find small designed to classify ImageNet dataset images, so they perform patterns that help in the classification process. However, the

consolidation patterns are relatively big, so we need to use a bigger filter size in our convolutional operations. This is what

we do in the ‘‘DeepChest” model, where the filter size is 7 × 7.

This filter size increases the number of parameters in the pro-

posed model. DeepChest has approximately 104 million param- eters, while VGG16 [[18]](#_bookmark24), Mohammad et al. [[19]](#_bookmark25), DenseNet-121 [[20]](#_bookmark26), and MobileNet [[21]](#_bookmark27) have 14 million, 44 million, 7, and 3 million parameters, respectively.

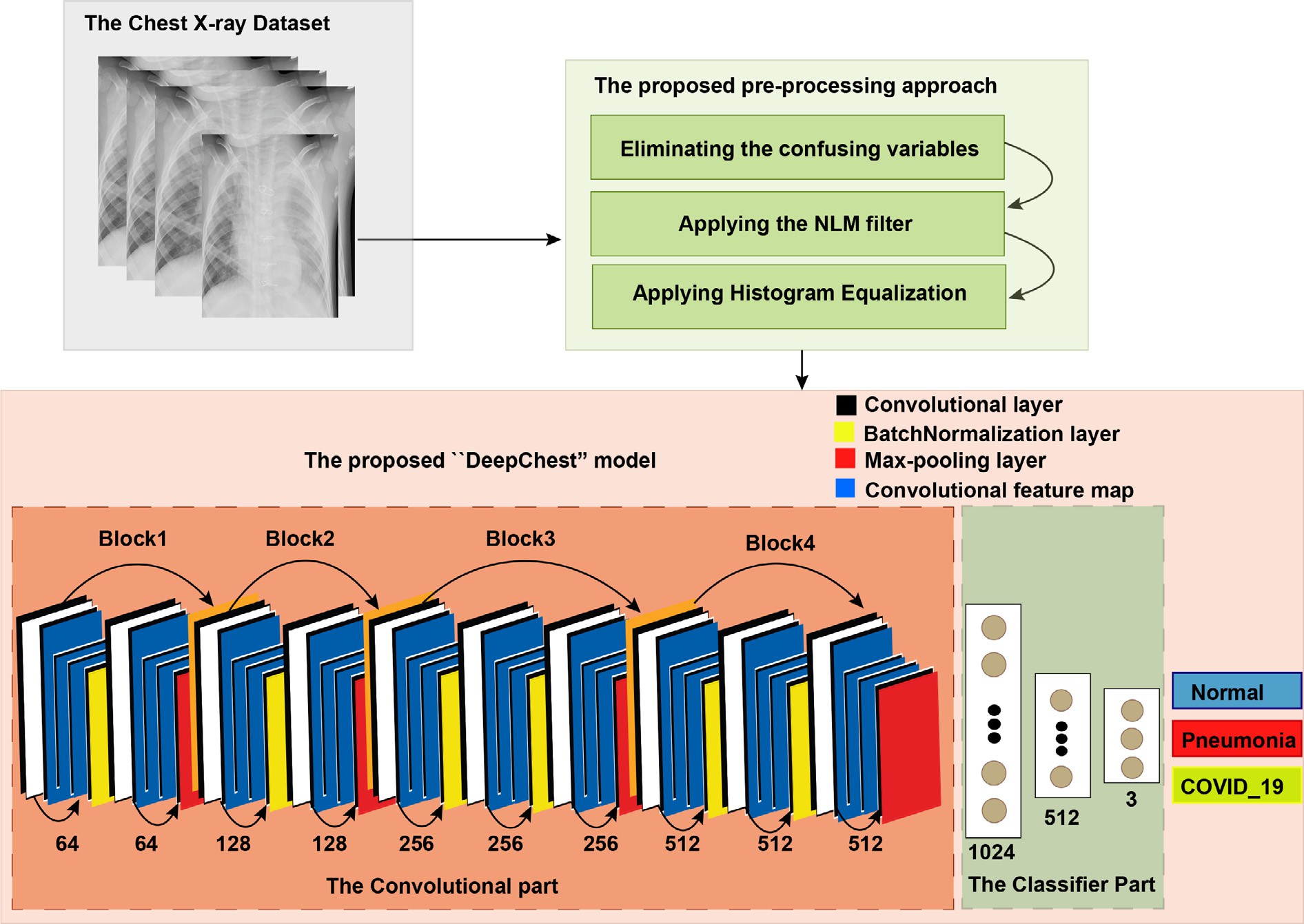


Fig. 4. The structure of ‘‘DeepChest”, this structure consists of five blocks, the first four is convolutional part and the last one is the classifier part, this structure also has ten convolutional layers, five BatchNormalization layers, and four max-pooling layers.

Table 4

The input parameters on the proposed model ‘‘DeepChest”.

Parameters ‘‘DeepChest”

Image Size 244 × 244 × 3

Learning Rate 0.0001

Decay 1*e* — 5

Batch Size 16

Step Per Epoch 385

Validation Steps 1

Optimizer Adam

CALL Backs Model Checkpoint & CSV

Logger

Loss Categorical Cross Entropy

Class Weights Auto

epoch that have the highest accuracy. The saved weights were loaded for the next training step see [Figs. 1 and 5](#_bookmark4), But the DCNNs (VGG16 [[18]](#_bookmark24), Mohammad et al. [[19]](#_bookmark25), MobileNet [[21]](#_bookmark27), and DenseNet-121 [[20]](#_bookmark26).) were pre-trained via ImageNet dataset [[48]](#_bookmark41). The proposed training strategy uses 3 times less memory space than training the investigated models directly with 30 epochs, which manages to free the random access memory (RAM) after every 10 epochs.

1. Experimental results
   1. *Dataset and statistical data analysis*

Number Of Neurons In The First Classifier Layer

1024

We generate the main used dataset in this work based on the

First Drop Out 0.7

Number Of Neurons In The Second Classifier 512

Layer

Second Drop Out 0.5

Pneumonia chest X-ray images dataset [[15]](#_bookmark20), Coronavirus chest X- ray images dataset [[16]](#_bookmark21), and COVID-XRay-5 k dataste [[17]](#_bookmark22). They are organized into three folders (Train, Test, and Val), including

Number Of Neurons In The Third Classifier Layer

Activation Function Of The Last Classifier Layer

Table 5

3

*SoftMax*

sub-folders for every image class (Normal, Pneumonia, COVID- 19). There are 7512 chest X-ray images with the extension (JPEG) divided into three classes of COVID-19 (1323), Pneumonia (4240 images), and Normal (1949 images). We applied A filtering process to the dataset that excluded 33 chest X-ray images from the Pneu- monia folder, which is included in the train folder. This exclusion is

applied because the excluded images have no consistency with the

The structure details of the proposed ‘‘DeepChest” model.

The Proposed ‘‘DeepChest” Model Structure

Block # Layer (type) Output Shape Block1 First Layer: Conv1-1 (Conv2D) (None, 224, 224, 64)

Second Layer: bn1 (Batch Normalization) (None, 224, 224, 64)

Third Layer: Conv1-2 (Conv2D) (None, 224, 224, 64)

Fourth Layer: pool1 (MaxPooling2D) (None, 112, 112, 64)

Block2 First Layer: Conv2-1 (Conv2D) (None, 112, 112, 128)

Second Layer: bn2 (Batch Normalization) (None, 112, 112, 128)

Third Layer: Conv2-2 (Conv2D) (None, 112, 112, 128)

Fourth Layer: pool2 (MaxPooling2D) (None, 56, 56, 128)

Block3 first Layer: Conv3-1 (Conv2D) (None, 56, 56, 256)

Second Layer: bn3 (Batch Normalization) (None, 56, 56, 256)

Third Layer: Conv3-2 (Conv2D) (None, 56, 56, 256)

Fourth Layer: bn3 (Batch Normalization) (None, 56, 56, 256)

Fifth Layer: Conv3-3 (Conv2D) (None, 56, 56, 256)

Sixth Layer: pool3 (MaxPooling2D) (None, 28, 28, 256)

Block4 First Layer: Conv4-1 (Conv2D) (None, 28, 28, 512)

Second Layer: bn4 (Batch Normalization) (None, 28, 28, 512)

Third Layer: Conv4-2 (Conv2D) (None, 28, 28, 512)

Fourth Layer: bn5 (Batch Normalization) (None, 28, 28, 512)

Fifth Layer: Conv4-3 (Conv2D) (None, 28, 28, 512)

Sixth Layer: pool4 (MaxPooling2D) (None, 14, 14, 512) Block5 First Layer: flatten (Flatten) (None, 100352)

Second Layer: fc1 (Dense) (None, 1024)

Operation: dropout1 (Dropout) (None, 1024)

Third Layer: fc2 (Dense) (None, 512)

Operation:dropout2 (Dropout) (None, 512)

Fourth Layer: fc3 (Dense) (None, 3)

*4.2. Training strategy*

In the proposed training strategy, we apply the training process of ‘‘DeepChest” from scratch. However, the chest X-ray images were small in number. To overcome this problem, the Image Data Generator method [[40]](#_bookmark41) has been used to generate chest X-ray images. The generated data will be in the same domain to force DCNN models to learn only the desirable features instead of bring- ing other chest X-ray datasets for training which may confuse the

DCNN models during the training process. The proposed training

pre-processing approach. This inconsistency due to the pre-defined window that crops the images is larger than the image itself. We generate 1323 COVID images with the same extension (JPEG) based on the 70 COVID-19 X-ray images available in the Coron- avirus dataset. The generation rules were rotation range with 360 degrees, horizontal flip, vertical flip, width shift range with 0.05, height shift range with 0.05, zoom range 0.05, and fill mode ‘‘nearest” according to Keras image data generator [[40]](#_bookmark41) see [Table 6](#_bookmark13). The used generated dataset available in Github. The proposed model and the other four compared models are developed using Python and Keras library [[43]](#_bookmark41) on Tensorflow [[44]](#_bookmark41), Google Colabora- tory [[45]](#_bookmark41) Notebooks are used along with Google drive where the dataset is uploaded. All statistical and computations tasks were calculated using statistic methods and sklearn packages [[46]](#_bookmark41) of Python version 3.8.3 [[47]](#_bookmark41) which was released on 13 May 2020. We consider the test data included in the generated chest X-ray dataset to evaluate the result of the investigated DCCN models. [Fig. 6](#_bookmark14) demonstrates the train accuracy and loss functions of VGG16, DenseNet121, Mohammad et al., MobileNet, and the pro- posed ‘‘DeepChest” model.

The confusion matrices for the investigated DCNN models after applying the proposed pre-processing approach were calculated due to [Table 8](#_bookmark15) for the performance evaluation process for each model. In this presented work the Normal, Pneumonia and COVID-19 classes are recognized as negative (demonstrated by – notation in [Table 8](#_bookmark15)), positive (demonstrated by + notation in [Table 8](#_bookmark15)) and double-positive classes (demonstrated by ++ notation in [Table 8](#_bookmark15)), respectively.The confusion matrices for the investi- gated DCNN models before applying the proposed pre-processing approach also were calculated see [Table 9](#_bookmark16). The specificity, sensitiv- ity, F1-score, and accuracy factors for every investigated model before and after the pre-processing approach are demonstrated in [Tables 10 and 11](#_bookmark17) respectively. Based on the test set data which are not used in the training process, specificity, sensitivity, accu- racy and F1 score metrics are computed due to the following equations:

strategy has three steps. We applied these steps to all the investi- gated models (VGG16 [[18]](#_bookmark24), Mohammad et al. [[19]](#_bookmark25), MobileNet [[21]](#_bookmark27), and DenseNet-121 [[20]](#_bookmark26).), including the ‘‘DeepChest”.

# *Specificity* = *TN*

*TN* + *FP*

# *TP*

(3)

Each step has ten epochs with 385 stepper epoch and batch

size 16, and then after each step, we save the weights of the

*Sensivity* = *TP* + *FN* (4)



Fig. 5. VGG16 [[18]](#_bookmark24), Mohammad et al. [[19]](#_bookmark25), MobileNet [[21]](#_bookmark27), and DenseNet-121 [[20]](#_bookmark26) were pre-trained with ImageNet dataset before the training process with the used chest X- ray dataset. However, the ‘‘DeepChest” model were trained only with the used chest X-ray dataset.

Table 6

The used generated balanced dataset details.

*Accuracy* = *TP*

*TP* + *TN*

+ *TN* + *FP* + *FN*

(5)

# *TP*

|  |  |  |  |
| --- | --- | --- | --- |
| Dataset | COVID-19 | Pneumonia | Normal |
| Corona Virus dataset | 70 | 0 | 28 |
| COVID-XRay-5 k dataset | 184 | 0 | 580 |
| Pneumonia dataset | 0 | 4273 | 1583 |
| Excluded images | 0 | 33 | 0 |
| Generated images | 1323 | 0 | 0 |
| Training set | 979 | 3842 | 1949 |
| Validating set | 8 | 8 | 8 |
| Testing set | 336 | 390 | 234 |

*F*1 = *TP* 1

(6)

+ 2 (*FP* + *FN*)

In the above equations, TN, TP, FN, and FP indicate True Nega-

tive, True Positive, False Negative, and False Positive factors, respectively.



Fig. 6. Figure A shows the investigated DCNN models (VGG16, Mohammed et al., DenseNet-121, MobileNet, and the proposed model (DeepChest))’s training accuracy in the training process, and Figure B shows the investigated DCNN models (VGG16, Mohammed et al., DenseNet-121, MobileNet, and the proposed model (DeepChest))’s training loss in the training process.

The confusion matrix of the investigated models and the presented ‘‘DeepChest” after applying our proposed pre-processing approach.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| MODELS | Classes | Prediction ++ | Prediction + | Prediction — | Total |
| VGG16 [[18]](#_bookmark24) | ++ | 292 | 44 | 0 | 336 |
|  | + | 0 | 384 | 6 | 390 |
|  | — | 0 | 11 | 223 | 234 |
|  | Total | 292 | 439 | 229 | 960 |
| DenseNet-121 [[20]](#_bookmark26) | ++ | 325 | 11 | 0 | 336 |
|  | + | 0 | 381 | 9 | 390 |
|  | — | 0 | 62 | 172 | 234 |
|  | Total | 325 | 454 | 181 | 960 |
| Mohammad et al. [[19]](#_bookmark25) model | ++ | 329 | 6 | 1 | 336 |
|  | + | 0 | 366 | 24 | 390 |
|  | — | 3 | 16 | 215 | 234 |
|  | Total | 332 | 388 | 240 | 960 |
| MobileNet [[21]](#_bookmark27) | ++ | 330 | 5 | 1 | 336 |
|  | + | 9 | 369 | 12 | 390 |
|  | — | 15 | 9 | 210 | 234 |
|  | Total | 354 | 383 | 223 | 960 |
| ‘‘DeepChest” | ++ | 334 | 2 | 0 | 336 |
|  | + | 1 | 387 | 2 | 390 |
|  | — | 8 | 20 | 206 | 234 |
|  | Total | 343 | 409 | 208 | 960 |

Table 9

The confusion matrix of the investigated models and the presented ‘‘DeepChest” before applying our proposed pre-processing approach.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| MODELS | Classes | Prediction ++ | Prediction + | Prediction — | Total |
| VGG16 [[18]](#_bookmark24) | ++ | 292 | 44 | 0 | 336 |
|  | + | 0 | 384 | 6 | 390 |
|  | — | 0 | 201 | 33 | 234 |
|  | Total | 292 | 629 | 39 | 960 |
| DenseNet-121 [[20]](#_bookmark26) | ++ | 325 | 11 | 0 | 360 |
|  | + | 0 | 381 | 9 | 390 |
|  | — | 0 | 132 | 102 | 234 |
|  | Total | 325 | 524 | 111 | 960 |
| Mohammad et al. [[19]](#_bookmark25) model | ++ | 329 | 6 | 1 | 336 |
|  | + | 0 | 366 | 24 | 390 |
|  | — | 3 | 96 | 135 | 234 |
|  | Total | 332 | 468 | 160 | 960 |
| MobileNet [[21]](#_bookmark27) | ++ | 310 | 21 | 5 | 336 |
|  | + | 26 | 330 | 34 | 390 |
|  | — | 13 | 31 | 190 | 234 |
|  | Total | 349 | 382 | 229 | 960 |
| ‘‘DeepChest” | ++ | 334 | 2 | 0 | 336 |
|  | + | 1 | 387 | 2 | 390 |
|  | — | 8 | 114 | 112 | 234 |
|  | Total | 343 | 503 | 114 | 960 |

Table 10

A comparison between the investigated models and the presented ‘‘DeepChest” in Accuracy, Sensitivity, Specificity, and F1-score after applying our proposed pre-processing approach.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Metrics | VGG16 [[18]](#_bookmark24) | Mohammad et al. [[19]](#_bookmark25) model | DenseNet-121 [[20]](#_bookmark26) | MobileNet [[21]](#_bookmark27) | ‘‘DeepChest” |
| Accuracy | 0.9365 | 0.9479 | 0.9145 | 0.9469 | 0.9656 |
| COVID-19 Accuracy | 0.8690 | 0.9673 | 0.9792 | 0.9821 | 0.994 |
| COVID-19 Sensitivity | 0.945 | 0.8916 | 0.8207 | 0.9402 | 0.9398 |
| COVID-19 Specificity | 0.9324 | 0.9981 | 0.9805 | 0.98302 | 0.9965 |
| COVID-19 F1 score | 0.939 | 0.9374 | 0.8935 | 0.9611 | 0.9674 |
| Pneumonia Accuracy | 0.9846 | 0.9385 | 0.9769 | 0.94615 | 0.9923 |
| Pneumonia Sensitivity | 1 | 0.9892 | 1 | 0.9584 | 0.9798 |
| Pneumonia Specificity | 0.9885 | 0.9775 | 0.922 | 0.9626 | 0.9945 |
| Pneumonia F1 score | 0.9942 | 0.9767 | 0.991 | 0.9652 | 0.9871 |
| Normal Accuracy | 0.953 | 0.9188 | 0.7350 | 0.8974 | 0.8803 |
| Normal Sensitivity | 0.8352 | 0.9729 | 0.9399 | 0.9375 | 0.9857 |
| Normal Specificity | 0.984 | 0.9722 | 0.91927 | 0.9668 | 0.9626 |
| Normal F1 score | 0.9035 | 0.9726 | 0.9294 | 0.9637 | 0.974 |

1. Performance comparison for the DCNN models

The investigated multi-use DCNN models were pre-trained via the ImageNet dataset. This allowed these models to take advantage

of feature similarity in points and edges at the low-level view. The pre-training process via ImageNet dataset [[48]](#_bookmark41) enables the DCCN to model to learn basic patterns such as points, edges, and lines. Therefore the ImageNet weights were used with VGG16 [[18]](#_bookmark24),

A comparison between the investigated models and the presented ‘‘DeepChest” in Accuracy, Sensitivity, Specificity, and F1-score before applying our proposed pre-processing approach.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Metrics | VGG16 [[18]](#_bookmark24) | Mohammad et al. [[19]](#_bookmark25) model | DenseNet-121 [[20]](#_bookmark26) | MobileNet [[21]](#_bookmark27) | ‘‘DeepChest” |
| Accuracy | 0.7385 | 0.8646 | 0.8417 | 0.8646 | 0.8677 |
| COVID-19 Accuracy | 0.8690 | 0.9792 | 0.9673 | 0.9226 | 0.9940 |
| COVID-19 Sensitivity | 0.5852 | 0.7327 | 0.6974 | 0.8267 | 0.7422 |
| COVID-19 Specificity | 0.9046 | 0.9862 | 0.7773 | 0.9524 | 0.996 |
| COVID-19 F1 score | 0.7106 | 0.9408 | 0.73521 | 0.8851 | 0.8506 |
| Pneumonia Accuracy | 0.9846 | 0.9385 | 0.9769 | 0.8462 | 0.9923 |
| Pneumonia Sensitivity | 1 | 0.9882 | 1 | 0.9483 | 0.9798 |
| Pneumonia Specificity | 0.9819 | 0.9508 | 0.989 | 0.8929 | 0.9933 |
| Pneumonia F1 score | 0.9909 | 0.9692 | 0.9943 | 0.91964 | 0.9865 |
| Normal Accuracy | 0.1410 | 0.5769 | 0.4359 | 0.8119 | 0.4786 |
| Normal Sensitivity | 0.4286 | 0.9575 | 1 | 0.8017 | 0.97313 |
| Normal Specificity | 0.7708 | 0.8753 | 0.8425 | 0.9357 | 0.8553 |
| Normal F1 score | 0.3314 | 0.91454 | 0.91436 | 0.8635 | 0.9108 |

Mohammad et al. [[19]](#_bookmark25), MobileNet [[21]](#_bookmark27), and DenseNet-121 [[20]](#_bookmark26). These models are trained with a large amount of data from scratch. In this presented work, DCNN models were fine-tuned and retrained with chest X-ray images. Lastly, the generated chest X- ray images dataset was used to train these models. The weights were initialized randomly in the classifier layers to learn how to classify X-ray images’ features as Normal, pneumonia, or COVID-

19. So, the final result showed that the well-known DCNN models were customized by training and fine-tuning the trainable layers of the model.

The DCCN models [[18–21]](#_bookmark24) have millions of parameters(fat and deep), so they have many purposes. The one, these models config- ured several convolutional layers to extract features from images and max-pooling layers to reduce the dimension of X-ray images through layers. In the second one, the categories of ImageNet data- set [[48]](#_bookmark41) have features that managed the DCCN models from effi- cient classification. On the other hand, the generated chest X-ray dataset has only three classes, Normal, Pneumonia, and COVID-

19. A small fraction of the chest X-ray image distinguishes the image with COVID-19 or Pneumonia from the Normal image. Therefore when we use many max-pooling layers will remove the feature separator of the three classes. Moreover, DCNN models need large chest data to learn the chest pattern, see [Tables 8–11](#_bookmark15). To overcome these issues, we proposed the ‘‘DeepChest” model that is fat and deep enough for chest X-ray images and has a suitable number of max-pooling layers as we compared with the investi- gated multi-use DCNN models. We input chest X-ray images into the training process after the pre-processing step. Batch Normal- ization [[22]](#_bookmark28) layers allow us to train the proposed model faster, get-

ting higher learning rates as shown in [Fig. 7](#_bookmark18) and the process of the initializing weight becomes easier. For an overall demonstration of what our contribution was, see [Fig. 1](#_bookmark4).The accuracy, specificity, sen- sitivity, and F1score were used to make a demonstrative compar- ison between all of the investigated models in this paper, see [Table 10 and 11](#_bookmark17). The experimental results showed the following:

* [Figs. 7 and 6](#_bookmark18) show that ‘‘DeepChest” achieved the highest and stablest training accuracy.
* [Table 10](#_bookmark17) shows that DeepChest model has the highest accuracy, sensitivity, specificity, and F1score among the investigated

models, so ImageNet weights should not be used as pre- trained weights as the first choice.

* DeepChest also achieved better training and testing time costs

than the other investigated models, see [Table 7](#_bookmark23).

* We made up the problem of a small number of images available in the chest X-ray dataset via using a data generator [[40]](#_bookmark41) pro-

vided by Keras, which allowed us to generate more data from the same domain.

* ‘‘DeepChest” used Drop-out layers two times with a rate of 0.7

and 0.5, respectively, to prevent data overfitting [[42]](#_bookmark41).

* [Tables 8–11](#_bookmark15) shows the huge effect of the proposed pre- processing approach on the achieved results.
  1. *The effect of the proposed pre-processing approach*

As demonstrated in Section 2, the pre-processing approach was critical to improving the DCNN model classification process. In order to evaluate the positive effect of the presented pre-



Fig. 7. The training accuracy metric among investigated models through the three stages training strategy.

Table 7

The average training time for one epoch and the average testing time for the test set in seconds.

Models Average Training Time Average Testing Time

(s) (s)

|  |  |  |
| --- | --- | --- |
| VGG16 [[18]](#_bookmark24) | 252 | 66 |
| Mohammad et al. [[19]](#_bookmark25) | 241.5 | 73 |
| model DenseNet-121 [[20]](#_bookmark26) | 223.3 | 62 |
| MobileNet [[21]](#_bookmark27) | 158.8 | 59 |
| ‘‘DeepChest” | 148.3 | 53 |

processing approach, the generated chest X-ray dataset was input to the investigated DCNN models after and before applying the pre-processing step. [Tables 8–11](#_bookmark15) shows the huge improvement of the achieved results.

1. Conclusion

This paper proposes a comprehensive method for detecting Pneumonia and COVID-19 in chest X-ray images. This method includes: First, generating a balanced chest X-ray dataset with three classes Normal, Pneumonia, and COVID-19. Second, a pre- processing approach stands for eliminating the confusing variables, removing noise from the X-ray images, and improving the contrast of these images. Third, a training strategy with three phases. Each phase has ten epochs. DeepChest saves the weights of the highest epoch accuracy. Then it loads the saved weights to the model before the second training phase begins. The same thing happens in the second and the third phase. Fourth, our problem-based model ‘‘DeepChest” was proposed to learn the desirable features of the chest X-ray images. We conducted The experimental evalu- ations of the proposed approach on a dataset with 7512 chest X- ray images. The proposed approach achieved an accuracy of 96.56% overall, 99.40% in detecting COVID-19, and 99.32% in detecting Pneumonia. In actual practice, the presented approach can be used as a computer-aided diagnosis tool to get accurate results in detecting Pneumonia and COVID-19 in chest X-ray images.

Code availability

We shared all the DCNN models investigated in this paper and the used dataset through this GitHub repository. We hope that the presented approach will be helpful in future research.

Declaration of Competing Interest

The authors declare that they have no known competing finan- cial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

1. [Al-Ani Z, Suut S, Khan AN. Multifocal Lung Consolidation: Differential](http://refhub.elsevier.com/S1110-8665(22)00002-0/h0005) [diagnosis The Role of Imaging. United Kingdom Radiological Congress](http://refhub.elsevier.com/S1110-8665(22)00002-0/h0005) [(UKRC); 2015](http://refhub.elsevier.com/S1110-8665(22)00002-0/h0005).
2. [Jackson A, Bridgman A, Black A, Youzguin B. Can You Tell What It Is Yet? An](http://refhub.elsevier.com/S1110-8665(22)00002-0/h0010) [Investigation Into Identification Of Consolidation Of Chest X-Rays In Patients](http://refhub.elsevier.com/S1110-8665(22)00002-0/h0010) [Diagnosed With Community Acquired Pneumonia. American Thoracic Society](http://refhub.elsevier.com/S1110-8665(22)00002-0/h0010) [International Conference Abstracts 2017. A3929–A3929](http://refhub.elsevier.com/S1110-8665(22)00002-0/h0010).
3. Schuijt Tim J, Lankelma Acqueline M, Scicluna Brendon P, et al. The gut microbiota plays a protective role in the host defence against pneumococcal pneumonia. Gut Microbiota 2016;65:575–83. doi: [https://doi.org/10.1136/](https://doi.org/10.1136/gutjnl-2015-309728) [gutjnl-2015-309728](https://doi.org/10.1136/gutjnl-2015-309728).
4. Anthony J, Scott G, Wonodi Chizoba, Deloria Maria, et al. The definition of pneumonia, the assessment of severity, and clinical standardization in the pneumonia etiology research for child health study. Clin Infect Dis 2012:54. doi: <https://doi.org/10.1093/cid/cir1065>.
5. Tong Sabine, Amand Caroline, Kieffer Alexia, et al. Trends in healthcare utilization and costs associated with pneumonia in the United States during 2008–2014. BMC Health Services Res 2018;18:1–8. doi: [https://doi.org/](https://doi.org/10.1186/s12913-018-3529-4) [10.1186/s12913-018-3529-4](https://doi.org/10.1186/s12913-018-3529-4).
6. World Health Organization (WHO). Accessed on 2020-08-19 [https://www.](https://www.who.int/) [who.int](https://www.who.int/)..
7. Qin Chunli, Yao Demin, Shi Yonghong, et al. Computer-aided detection in chest radiography based on artificial intelligence: A survey. BioMed Eng Online 2018;17:1–23. doi: <https://doi.org/10.1186/s12938-018-0544-y>.
8. Russakovsky Olga, Deng Jia, Hao Su, et al. ImageNet Large Scale Visual Recognition Challenge. Int J Comput Vis 2015;115:211–52. doi: [https://doi.](https://doi.org/10.1007/s11263-015-0816-y) [org/10.1007/s11263-015-0816-y](https://doi.org/10.1007/s11263-015-0816-y).
9. Liu Weibo, Wang Zidong, Liu Xiaohui, et al. A survey of deep neural network architectures and their applications. Neurocomputing 2017;234:11–26. doi: <https://doi.org/10.1016/j.neucom.2016.12.038>.
10. Litjens Geert, Koo Thijs, Bejnordi Babak Ehteshami, et al. A survey on deep learning in medical image analysis. Med Image Anal 2017;42:60–88. doi: <https://doi.org/10.1016/j.media.2017.07.005>.
11. Rajpurkar Pranav, Irvin Jeremy, Zhu Kaylie, et al. CheXNet: Radiologist-Level Pneumonia Detection on Chest X-Rays with Deep Learning, arXiv preprint1711.05225. 2017: 3–9..
12. Shen Dinggang, Guorong Wu, Suk Heung-Il, et al. Deep Learning in Medical Image Analysis. Adv Exp Med Biol 2017;19:221–48. doi: [https://doi.org/](https://doi.org/10.1146/annurev-bioeng-071516-044442) [10.1146/annurev-bioeng-071516-044442](https://doi.org/10.1146/annurev-bioeng-071516-044442).
13. Chen Jonathan H, Asch Steven M. Machine learning and prediction in medicine-beyond the peak of inflated expectations. New Engl J Med 2017;376:2507–9. doi: <https://doi.org/10.1056/NEJMp1702071>.
14. Mittal Anish, Moorthy Anush Krishna, Bovik Alan Conrad. No-reference image quality assessment in the spatial domain. IEEE Trans Image Process 2012;21:4695–708. doi: <https://doi.org/10.1109/TIP.2012.2214050>.
15. Mooney P. Chest X-Ray Images (Pneumonia) Dataset, Uploaded since (2018). <https://www.kaggle.com/paultimothymooney/chest-xray-pneumonia>..
16. Sajid Nabeel. Chest X-Ray Images (Corona Virus Dataset) Dataset, Uploaded since (2020). [https://www.kaggle.com/nabeelsajid917/covid-19-x-ray-](https://www.kaggle.com/nabeelsajid917/covid-19-x-ray-10000-images) [10000-images](https://www.kaggle.com/nabeelsajid917/covid-19-x-ray-10000-images)..
17. Minaee Shervin, Kafieh Rahele, et al. COVID-XRay-5K Dataset, Accessed since (2020). <https://github.com/ieee8023/covid-chestxray-dataset>..
18. Simonyan Karen, Zisserman Andrew. Very deep convolutional networks for large-scale image recognition, arXiv preprint1409.1556 (2015)..
19. Rahimzadeh Mohammad, Attar Abolfazl. A modified deep convolutional neural network for detecting COVID-19 and pneumonia from chest X-ray images based on the concatenation of Xception and ResNet50V2. Inf Med Unlocked 2020;19: . doi: <https://doi.org/10.1016/j.imu.2020.100360>100360.
20. Huang Gao, Liu Zhuang, van der Maaten Laurens, et al. Densely Connected Convolutional Networks. In Proceedings of the IEEE conference; 2017. [https://](https://ieeexplore.ieee.org/document/8099726) [ieeexplore.ieee.org/document/8099726](https://ieeexplore.ieee.org/document/8099726). .
21. Howard AG, Zhu M, Chen B, Kalenichenko D, Wang W, Weyand T, Andreetto M, Adam H. Mobilenets: Efficient convolutional neural networks for mobile vision applications. arXiv preprint1704.04861 (2017)..
22. Ioffe Sergey, Szegedy Christian. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. Mach Learn; 2015. https://arxiv.org/abs/1502.03167. .
23. Yaniv Bar, Idit Diamant, Lior Wolf, et al. Chest pathology detection using deep learning with non-medical training. In: Proceedings – International Symposium on Biomedical Imaging. p. 294–7. doi: [https://doi.org/10.1109/](https://doi.org/10.1109/ISBI.2015.7163871) [ISBI.2015.7163871](https://doi.org/10.1109/ISBI.2015.7163871).
24. Hua Kai-lung, Hsu Che-hao, chusnul Hidayati shintami, et al. Computer-aided classification of lung nodules on computed tomography images via deep learning technique. OncoTargets Ther 2015;8:2015–22. doi: [https://doi.org/](https://doi.org/10.2147/OTT.S80733) [10.2147/OTT.S80733](https://doi.org/10.2147/OTT.S80733).
25. Islam MT, Aowal MA, Minhaz AT, Ashraf K. Abnormality Detection and Localization in Chest X-Rays using Deep Convolutional Neural Networks, ar CoRR1705.09850; 2017..
26. Zech John R, Badgeley Marcus A, Liu Manway, et al. Confounding variables can degrade generalization performance of radiological deep learning models, Unsupervised domain adaptation for medical imaging segmentation with self- ensembling 2018;9:1–15.1807.00431..
27. Xu Shuaijing, Wu Hao, Bie Rongfang. CXNet-m1: Anomaly Detection on Chest X-Rays with Image-Based Deep Learning. IEEE Access 2019;7:4466–77. doi: <https://doi.org/10.1109/ACCESS.2018.2885997>.
28. Varela-Santos Sergio, Melin Patricia. A new approach for classifying coronavirus COVID-19 based on its manifestation on chest X-rays using texture features and neural networks. Inf Sci 2021;545:403–14. doi: [https://](https://doi.org/10.1016/j.ins.2020.09.041) [doi.org/10.1016/j.ins.2020.09.041](https://doi.org/10.1016/j.ins.2020.09.041).
29. Jaiswala A, Gianchandania N, et al. Classification of the COVID-19 infected patients using DenseNet201 based deep transfer learning. J Biomol Struct Dyn 2020:1–8. doi: <https://doi.org/10.1080/07391102.2020.1788642>.
30. Civit-Masot J, Luna-Perej’on F, et al. Deep Learning System for COVID-19 Diagnosis Aid Using X-ray Pulmonary Images. Appl Sci 2020;10:4640. doi: <https://doi.org/10.3390/app10134640>.
31. Wang Hongyu, Wang Shanshan, et al. Triple attention learning for classification of 14 thoracic diseases using chest radiography. Med Image Anal 2021;67. doi: <https://doi.org/10.1016/j.media.2020.101846>.
32. Minaee Shervin, Kafieh Rahele, et al. Deep-COVID: Predicting COVID-19 from chest X-ray images using deep transfer learning. Med Image Anal 2020:65. doi: <https://doi.org/10.1016/j.media.2020.101794>.
33. Behzadi-khormouji Hamed, Rostami Habib, Salehi Sana, et al. Deep learning, reusable and problem-based architectures for detection of consolidation on chest X-ray images. Comput Methods Programs Biomed 2020;185: . doi: <https://doi.org/10.1016/j.cmpb.2019.105162>105162.
34. Buades A, Coll B, Morel J-M. A Non-Local Algorithm for Image Denoising. In: Proceedings, In CVPR. IEEE; 2005. p. 60–5. doi: [https://doi.org/10.1109/](https://doi.org/10.1109/CVPR.2005.38) [CVPR.2005.38](https://doi.org/10.1109/CVPR.2005.38).
35. Deng Guang, Cahill LW. An adaptive Gaussian filter for noise reduction and edge detection. In: 1993 IEEE conference record nuclear science symposium and medical imaging conference. IEEE; 1993. doi: [https://doi.org/10.1016/](https://doi.org/10.1016/0165-1684(95)00020-E) [0165-1684(95)00020-E](https://doi.org/10.1016/0165-1684(95)00020-E).
36. Pitas, Venetsanopoulos I. Median Filters, Nonlinear Digital Filters: Principles and Applications. Springer US; 1990. doi: [https://doi.org/10.1007/978-1-4757-](https://doi.org/10.1007/978-1-4757-6017-0_4) [6017-0\_4](https://doi.org/10.1007/978-1-4757-6017-0_4). 63-116.
37. Paranjape Raman B. Fundamental Enhancement Techniques, 3–18, Biomedical Engineering. University of Regina 2000. doi: [https://doi.org/10.1016/B978-](https://doi.org/10.1016/B978-012077790-7/50004-7) [012077790-7/50004-7](https://doi.org/10.1016/B978-012077790-7/50004-7).
38. Singh Navdeep, Kaur Lakhwinder, Singh Kuldeep. Histogram equalization techniques for enhancement of low radiance retinal images for early detection of diabetic retinopathy. Eng Sci Technol Int J 2019;22:736–45. doi: [https://doi.](https://doi.org/10.1016/j.jestch.2019.01.014) [org/10.1016/j.jestch.2019.01.014](https://doi.org/10.1016/j.jestch.2019.01.014).
39. Pan Sinno Jialin, Yang Qiang. A survey on transfer learning. IEEE Trans Knowl Data Eng 2010;22:1345–59. doi: <https://doi.org/10.1109/TKDE.2009.191>.
40. Chollet F. Keras ImageDataGenerator, Uploaded since; 2015.[https://keras.io/](https://keras.io/api/preprocessing/image/) [api/preprocessing/image/](https://keras.io/api/preprocessing/image/)..
41. Arribas JI, Cid-Sueiro J, Adali T, Figueiras-Vidal AR. Neural architectures for parametric estimation of a posteriori probabilities by constrained conditional density functions, Neural Networks for Signal Processing IX: Proceedings of the 1999 IEEE Signal Processing Society Workshop (Cat. No.98TH8468), 1999.

pp. 263–272. doi:10.1109/NNSP.1999.788145..

1. Srivastava Nitish, Hinton Geoffrey, Krizhevsky Alex, et al. Dropout: A Simple Way to Prevent Neural Networks from Overfitting 2014;15:1929-1958..
2. Chollet F. Keras Library, Uploaded since; 2015.<https://keras.io/>..
3. Google Brain Team. TensorFlow library, Uploaded since; 2015. [https://www.](https://www.tensorflow.org/) [tensorflow.org/](https://www.tensorflow.org/)..
4. Google. Google Colab Notebook, Established since; 2017. [https://colab.](https://colab.research.google.com/) [research.google.com/](https://colab.research.google.com/)..
5. Cournapeau D. scikit-learn library, Accessed in (2020–08-19). [https://scikit-](https://scikit-learn.org/) [learn.org/](https://scikit-learn.org/)..
6. Rossum Gv. Python library, uploaded since; 2007. <https://www.python.org/>..
7. Deng, Dong, Socher, et al. ImageNet: A Large-Scale Hierarchical Image Database; 2009. CVPR09. [https://www.image-net.org/papers/imagenet\_](https://www.image-net.org/papers/imagenet_cvpr09.bib) [cvpr09.bib](https://www.image-net.org/papers/imagenet_cvpr09.bib). .