International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Deep Convolution Neural Network for Respiratory Diseases Detection Using Radiology Images

Prita Patil[1], Vaibhav Narawade*[2]

Submitted: 18/09/2023 **Revised**: 17/11/2023 **Accepted**: 28/11/2023

Abstract: The respiratory issue has placed enormous and increasing pressure on the world's medical systems. The diagnosis, treatment, & even early identification of diseases may all be aided by processing performed on medical images. Deep Learning(DL) models would help physicians and radiologists provide patients with quick diagnostic support. Our research focuses on respiratory diseases like COVID-19, and pneumonia case identification model based on DL. It is trained using a dataset comprised of Chest X-ray (CXR) & CT (Computed Tomography) scan images. The major contribution of our research is to improve image dataset balancing, preprocessing radiology pictures to locate regions of interest (ROI), and further custom-built 28-layered Deep Neural Network as its architecture for deep learning. The dataset used to train the model was gathered from various publicly available time datasets collected from local hospitals. CT scan image data comprised 2 classes: COVID-19 (+) and COVID-19 (-), while X-ray image data included Multi Class(3 class labels): COVID-19, normal, and pneumonia cases. Using CXR pictures, this model trained with 99.12% accuracy for binary classes and 99.94% accuracy for Multiclass cases. The proposed model can attain a validation accuracy of 95.17% for Binary-class cases and 93.97% for multi-class cases. Improvising preprocessing to detect Region of Interest(ROI) and a high level of accuracy is a novel and potentially significant resource that enables radiologists to promptly detect and diagnose respiratory diseases using machine learning approaches even at the early stage of diagnosis.

Keywords: *Respiratory disease, COVID-19 detection, Radiology images, CT & X-ray images, DNN.*

1. Introduction

COVID-19 (Coronavirus Disease-19) is a serious respiratory viral disease that was initially started in late 2019 and has expanded internationally. Though several developed nations have made important progress in identifying & controlling COVID-19, most developing nations are still struggling to do so in huge populations [1]. There are frequently not enough COVID-19 diagnostic kits & supporting resources in nations where the number of patients is expanding rapidly. Researchers in data analysis & data science are now conducting extensive analyses of health-related data to help advance medical research. To this end, several ML (Machine Learning), DL (Deep Learning), DM (Data Mining), as well as PR (Pattern Recognition) techniques have been used to automatically classify medical images based on extracted features [2].

Using sophisticated computer-aided algorithms in medical imaging for recognition, diagnosis, and surgical planning is an active and expanding area of study [3]. People infected with COVID-19 exhibit severe inflammation of cavities & bronchioles due to the damage to their lungs. Indications derived from X-rays, Lung US (Ultrasound), and CT pictures are crucial for establishing an accurate medical diagnosis. Using an appropriate method, researchers may provide substantial assistance in examining COVID-19 accessible image datasets & produce findings that can aid in the fight against the disease. Although Image Processing (IP) has been applied in medical research & attracts the attention of researchers from a wide variety of domains [4][5], image processing is still in its initial stages in the context of COVID-19 detection & diagnosis study. Moreover, while some early research

suggests that imaging modalities may be useful in diagnosing COVID-19, no prior effort has thoroughly reviewed & synthesized this field of research to give a comprehensive knowledge of image processing to scholars & medical professionals. Regardless of the widespread use of AI, Regarding the use of ML & DL algorithms to MIP (Medical Image Processing) in context the of COVID-19 disease, there is comprehensive reviews and classifications of the literature on this subject.

This study sought to fill that gap by defining many previously published articles on the detection & diagnosis of COVID-19 respiratory disease using medical imaging modalities using systematic literature analysis. This makes it easier to assess and synthesize recent findings in a certain field, on a given problem, or into a given phenomenon that has caught the interest of researchers [6]. In addition, the methods employed for COVID-19 detection depend upon IP (Image Processing) methods investigated in this work.

[1]*Research Scholar, Department of Computer Engineering, Ramrao Adik Institute of Technology, D Y Patil Deemed to be University, India*
[2]*Department of Computer Engineering, Ramrao Adik Institute of Technology, D Y Patil Deemed to be University, India*
* *Corresponding Author Email: 1prita.patil@gmail.com, 2vaibhav.narawade@dypatil.edu*

Additionally, this study includes a conceptual map that describes research themes connected to the research study's data collecting, processing, detection, and assessment.

In this article, the authors discuss present limitations or research gaps to data preprocessing, define key processes for preparing medical imaging data of x-ray & CT scans for future development in deep learning model creation, investigate new approaches to solve the issue of data preprocessing, and afterward conduct classification using DL method. DNN (Deep Neural Networks), particularly CNN (Convolutional Neural Networks), are being used in various image classification applications, and they have shown considerably improved performance. There is evidence that CNN can perform as well as human experts in classifying medical images.

The novelty of this research work is defined by considering the following contributions:

- Extraction and interpretation of appropriate radiology image datasets, including CT scans and X-rays for respiratory disease infection identification.
- To investigate preprocessing techniques of radiology image data samples and propose improvements in the identification of ROI (Region of Interest) in detecting respiratory disease.
- Proposition and design of Custom-Built binary class and multi-class classification RDD_DNN Deep Neural Network in the respiratory disease detection and investigation of its performance analysis
- Comparative performance analysis and experimental evaluation of RDD_DNN with the existing classification models, progressive layer architectures, different respiratory datasets, etc.

This study, further classified into sections like section 2, covered the perspectives of many researchers in assessing the effect of COVID-19 as a respiratory disease on nations and persons, as well as the usage of DL in detecting COVID-19 using radiology pictures. In section 3, the dataset employed & model formulation is addressed. In section 3, several approaches to & the proposed flowchart are also provided. In addition, section 4 discusses various matrices and the assessment of training & testing outcomes using confusion matrices for models. Section 5 concludes this study by discussing its potential scope.

2. Related Work

Considering recent technological advancements, combining DL classifiers with medical pictures yields more optimistic findings corresponding to conventional RT-PCR testing, as well as more accurate identification & forecasting of COVID-19 cases. With CXR & CT-scan pictures, [7] proposes a deep transfer learning method that speeds the detection of COVID-19 patients. Since, in COVID-19, initial CXR screening may offer useful information for identifying probable COVID-19 patients. They evaluated three datasets referred to as 1) COVID-CXR, 2) SARS-COV-2 CT-scan, & 3) CXR Images (Pneumonia). As per collected findings, the proposed DL model may identify COVID-19-positive cases in two seconds quicker than RT-PCR tests presently utilized for COVID-19 case detection. They have also created a relationship between Pneumonia & COVID-19 patients that examines patterns among radiological pictures of pneumonia & COVID-19. In all of their experiments, they have employed a color visualization technique based on Grad-CAM to provide a clear interpretation of radiology image identification & subsequent action.

Recent research demonstrates that the DL algorithm that extracts the most important information from CT images may assist in COVID-19 detection. This model's pooling layer is a combination of the pooling layer & SE-Block (Squeeze Excitation Block) layer, as suggested in research [8]. Using Batch Normalization & Mish Function, the proposed model optimizes convergence time & performance of COVID-19 detection.

Using chest CT-scan pictures to identify COVID-19 automatically might save clinicians' workload & save the lives of thousands of patients. Using images obtained from chest CT scans and methods that depend upon DL, the authors of this work [9] established a reliable framework for automated COVID-19 screening. A publicly available CT-scan image dataset (including 1230 non-COVID & 1252 COVID-19 chest CT scans), two pre-trained DL models—MobileNetV2 & DarkNet19—also newly-designed lightweight DLM is used in this study to automate Covid-19 screening

DL is the most effective method of ML, and it offers helpful analysis for examining a huge number of CXR pictures, which may significantly influence the screening for COVID-19. In this study, [10] looked at CXR scans from the PA angle for individuals who had been harmed by COVID-19 and those who were well. Following cleaning the images and application of data augmentation, they employed DL-based CNN models and compared their respective levels of effectiveness. The accuracy of the Xception, ResNeXt, and InceptionV3 models has been evaluated & contrasted.

Noisy X-ray pictures may hinder the detection efficiency of deep CNNs (DCNNs). To enhance COVID-19 identification in noisy X-ray images without needing any preparation for noise reduction, [11] introduced a unique CNN technique employing adaptive convolution to increase the resilience of DCNN against impulse noise. In addition to the standard CNN architecture, this method adds an impulsive noise-map layer, adaptive convolution layer, and adaptive resizing layer. They also utilized the learning-to-augment technique with noisy X-ray images for better generalization. Their dataset of 2093 CXR images includes both non-COVID pneumonia (621 images) & COVID-19 (452 images) cases, as well as healthy controls (1020 images). GoogleNet, SqueezeNet, ResNet18, MobileNetv2, ShuffleNet, EfficientNetb0, & ResNet50 are just some of the pre-trained networks that have had their design tweaked to make them more resistant to impulsive noise. Using ResNet50 with the proposed noise-robust layers & learning-to-augment technique resulted in 2% higher classification accuracy than the advanced method during validation on noisy X-ray pictures.

A fast and accurate diagnostic approach is needed to detect COVID-19 infections quickly and effectively to halt their rapid spread. The global health crisis caused by the pandemic of COVID-19, which has now infected almost every country, was described by N. Phukkaphan et al. [12]. This paper defines some preliminary studies using E-Nose (Electronic Nose) technology to diagnose COVID-19 infection. COVID-19 positive results were confirmed using RT-PCR. Experimental results propose that the sum of sensory responses provides a useful measure of the intensity of human exhalation.

In the field of image classification, DL frameworks have made tremendous strides. DL is a kind of ML strategy in which the hierarchical structure of data processing layers is used to perform classification tasks and investigate features. Parameter sharing, equivariant representations, & sparse interactions are three important ideas supported by convolution that may help improve an ML system. With the help of neural network-based methods, it is possible to do classification & segmentation based on textures. DCNNs are well-known for their capacity to learn forms, textures, and colors, making them an ideal choice for this challenge.

DCNNs can execute convolutional operations on pictures because of the way weights in the network are shared. Since the network is equivariant for translations of the input, the number of parameters that must be learned is greatly reduced since the model does not have to train various detectors for similar objects happening at different places in an image. Adding more layers to a network and more neurons to each layer is the simplest technique to improve DCNN performance. The capacity of DL to automatically extract the most important characteristics for data interpretation & inference straight from annotated data is a key factor in its explosive growth and widespread use in various fields.

Alex Net[13] is the first deep convolutional network to be utilized for image classification. It comprises five convolutional layers (Conv) and three fully connected layers (FC). It accepts as input an RGB image with a size of 227X227. There are 96 11113 kernels in the first Conv layer. After the second and fifth Conv layers, respectively, max-pooling layers are added. The last three layers are fully connected. The last FC layer serves as a classification layer using SoftMax activation. It was trained on the ImageNet dataset, which has 1.2 million photos classified into 1000 different categories. In 2014, the Google Net [14] image classification algorithm was proposed. It has 22 layers and 4 million parameters, which are 12 times fewer than Alex Net. It is also known as the Inception module. The primary principle behind inception is to convolve the image with kernels of varying sizes to learn characteristics at various resolutions. The VGG[6] is used to categorize images. The ImageNet dataset is also used to train and test it. On ImageNet, it achieved top-5 accuracy of 92.3%. Alex Net offers a larger receptive field and fewer configuration choices. It contains 19 layers and 33 kernels instead of 1111 and 55 Alex Net kernels.

Reset[15] finished first in the 2015 (ILVRC). ResNet-50 is split into five stages, each having its own convolution and identification block. Each convolution block, like the identity blocks, has three convolution layers. ResNet-50 contains around 23 million trainable parameters. It employs skip connection and batch normalization to reduce the likelihood of gradients dropping. Dense Net[16] is a skip connection-based image classification algorithm. Dense Net connects each layer in a feed-forward fashion. It explores new features because of its extensive connections, whereas Resent fosters feature reuse. Image generation is one of the most promising fields in computer vision.

Image generation can be utilized in a variety of applications, including classification, image-to-image translation, segmentation, and so on. To generate fresh images, generative adversarial networks [17] are utilized. In general, the GAN has two modules that are adversarial trained. The generative network G oversees creating fresh samples, whereas the discriminative network D oversees distinguishing between fake (produced) images and original photos. To improve classification accuracy, both networks are trained concurrently. The attention mechanism, like GANs, is one of the most significant advances in deep learning research in the recent decade. Paying more attention to a certain entity entails giving it more weight. CNN models' attention layers tailor their behavior to extract application-specific characteristics for localization and classification. Another significant use of deep learning CNN is the ability Gradient-based techniques that attribute network output to perturbations of the input pixels are frequently used to achieve deep learning explain recently created Class Activation Mapping (CAM) has no fully connected layers. A subset of image classification CNNs uses CAM to identify discriminative regions. The classification of chest X-rays (CXRs) is critical in the analysis of medical images. Chest radiography is sufficient for determining the patient's health. However, for clinicians, attentively and thoroughly reading the material is always a huge challenge. Classifying abnormalities in chest X-rays is a time-consuming procedure for clinical radiologists. As a result, several solutions for dealing with the issues connected with chest radiography categorization and analysis have been proposed in the literature [19]. The first attempt was undertaken in the 1960s to establish a computer-aided diagnosis system for chest diseases, which proved effective in assisting pathologists [20]. [21] created one of the first computer-aided diagnoses for X-ray analysis of the chest and pneumoconiosis diagnosis. For medical purposes, many industrial and commercial devices, such as Riverain, CAD4 TB, and Delft imaging systems, have been developed [22].

Deep learning (DL) algorithms have been created to read chest X-rays and classify lung diseases[23]. These methods assist radiologists by decreasing diagnostic errors, assessing typical abnormalities or uncertainty, localizing suspicious spots, and overcoming the limitations of human bias and perception. A radiograph may reveal more than one pathology, such as tuberculosis, lung cancer pulmonary nodule, edema, and so on. A deep learning-based automated system may detect several diseases from a single radiograph or a single specific condition. Thus, detection techniques for chest diseases can be divided into two categories: multi-label disease detection and single-label disease detection. In the next subsections, this examination delves into the specifics of cutting-edge single and multiple illness detection systems.

This article aims to collect several research papers on radiography medical imaging datasets of CT scans & CXR images using DL technologies. Data preprocessing is very important for getting enhanced and segmented image datasets. Then a look toward research on deep learning methods in respiratory disease detection.

3. Proposed Work

This section discusses our proposed method called RDD_DNN which consists of a new custom-built 28-layer DNN model for respiratory diseases like COVID, Pneumonia etc. detection using chest X-ray and CT images. The model performance is evaluated based on accuracy, precision, recall, F1 score and confusion matrix for 2 class classification, 3 class - Multi Class classification.

3.1 Radiology Image Dataset Details

For this study, four medical imaging datasets are taken and a brief detail about these datasets, namely, CXR images & CT images dataset and RESP dataset are given below.

1)CXR images dataset : CXR (Covid-19 & Pneumonia) collection includes CXR pictures of Covid-19, Pneumonia, & normal patients. Data distribution information for all three classes is shown below in Table 1.

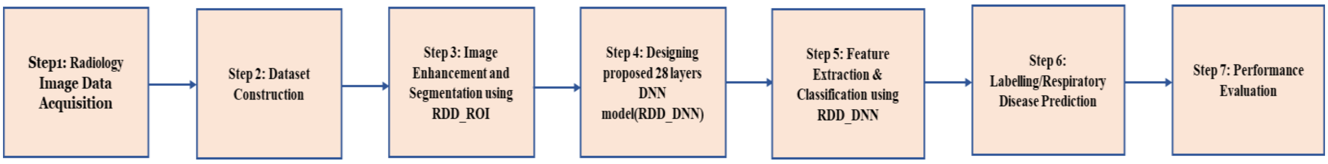


Fig. 1. A flowchart of the proposed RDD_DNN method

The dataset is arranged into two directories (train & test), and each folder has three subfoldeCOVID-19ID19, PNEUMONIA, NORMAL). The dataset comprises 6432 X-ray pictures, with test data being 20 percent of the total images. This dataset is available at https://www.kaggle.com/datasets/prashant268/chest-xray-covid19-pneumonia.[24]

2)CT images dataset: Two datasets, namely, the SARS-CoV-2 and the CT-Scan datasets, have been considered. The first data is the CT-Scan dataset, which may be employed to classify and automatically detect COVID-19 on CT scans. COVID-19 Lung CT Scans CT Scan Dataset about COVID-19 and pictures were gathered from COVID19-COVID-19-relateds published on bioRxiv, medRxiv, JAMA, Lancet, NEJM, etc. CTs with COVID-19 abnormalities are chosen by reading the captions of the figure studies. The dataset contains a total of 746 CT scan total images. Shown in Table 2 , A CT-Scan dataset is available at: https://www.kaggle.com/datasets/luisblanche/covidct?select=CT_COVID) [26] The SARS-CoV-2 The SARS-CoV-2 data set may be seen at https://www.kaggle.com/datasets/plameneduardo/sarscov2-ctscan-dataset[25]

3) RESP Dataset(X-ray images): The complete dataset consisted of labeled 316 chest X-ray images with 102 normal images, 110 bacterial pneumonia images, and 104 viral images of COVID-19 in shown in Table 3 and 60 unlabeled images 20 from each label class are used for the proposed system validation purposes. The entire dataset was acquired from a local hospital that contained chest X-ray images of various conditions. The dataset was quite imbalanced and limited.

Table 1. Data Information of Chest X-Ray

Data	*Covid19*	*Pneumonia*	*Normal*	*Total*
Training	460	3418	1266	5144
Testing	116	855	317	1288

Table 2. CT image data information

Dataset Name	*Covid images*	*Non-covid images*	*Total images*
COVID-CT dataset	349	397	746
SARS-COV-2	1252	1229	2481

Table 3. RESP Dataset image data information

Dataset Name	*Pneumonia Infected Image Samples*	*COVID Infected Image Samples*	*Normal Image Samples*	*Total images*
Training	88	83	81	252
Testing	22	21	21	64

3.2 Proposed Methodology

Our research study proposes a DL-based model named the RDD_DNN model. RDD_DNN model is a custom-built 28 layers DNN model. A research study is carried out to understand the features of each DNN layer and their parameters. After experimenting with different parameters and layer construction custom built a DNN model that gives better performance results than previously published DNN models. Fig..1 shows the workflow of our RDD_DNN method from the beginning to end. The details on each step of RDD_DNN is given below

Step1: Radiology Image Acquisition

For this study, four medical imaging datasets mentioned in dataset details are considered for a research study. Out of four image datasets, three are online research datasets from Kaggle, and another one real real-time dataset collected from local hospitals of infected Xray radiology images for testing and validating our proposed work

Step2: Dataset Construction

Dataset construction called RESP dataset majorly focuses on 2 steps: a)Image Data augmentation and b)Image Data Balancing.

One of the most significant difficulties encountered in the radiology pictures classification problem is the scarcity of images, making the datasets unbalanced. We adopted data augmentation to overcome this problem which encompasses techniques that improve the size and quality of training datasets. In this way, deep learning models can improve learning by enhancing generalization. Data Augmentation is useful to prevent models from overfitting, improve the model accuracy, and to reduce the operational cost of labeling and cleaning the raw dataset.

Our research study uses rotation, flipping, translation, and image edge detection to perform data augmentation. During the literature review, different edge detection techniques like Histogram equalization, Gaussian blurring, Bilateral filter, and Adaptive masking are studied and the experiment is carried out with different techniques to check the performance of the proposed system and it has been observed the system performs well with Gaussian blurring edge detection technique.

In the proposed method, oversampling is applied on train data reason if test data contains artificially generated images, the classifier results may be affected.. So, the better method is to first split the train and test data and then oversample only the training data. The function RESP_DATA_BALANCE ()creates all the artificially augmented images, till they reach the difference in values from the total high class label with maximum images. It then returns the newly created images of class labels. Data balancing obtained through the proposed method improves model accuracy and the operational cost of labeling. For more details refer: https://shorturl.at/pwBPS/

Step 3. Image Enhancement and Segmentation using the proposed RDD_ROI algorithm

For respiratory disease diagnosis using radiology images, it is important to Locate the Region of Interest (ROI). Our research study proposes RDD_ROI as an improvement in image enhancement and segmentation to preprocessed radiology images in the detection of respiratory diseases in the human body. RDD_ROI is an improved GLCM followed ban y unsupervised K-means clustering algorithm that relies on a statistical correlation between neighboring pixels. and considers the linear dependency of neighboring pixels of a given kernel window size. Features extracted for important pixels are then given to all pixels in surrounding using a weight matrix hence it reduces overall computation time during the image feature extraction. The feature vector related to every non-key pixel is a weighted sum of feature vectors for all weighting kernel windows containing that pixel. To speed up the calculation and decrease the amount of time spent on it, we compute Gray-level co-occurrence matrix (GLCM) features for essential pixels also then utilize interpolation to estimate GLCM features for other pixels. Before being sent to the classifiers, features are normalized such that they fall within the range 0–1, and the classifiers all get the identical set of features. In this work, the GLCM method extracts seven texture features like homogeneity, correlation, dissimilarity, contrast, entropy, energy, and

ASM of the radiology images.

Image Segmentation is a method of splitting an image into distinct groups. Unsupervised Machine Learning is then used to segment the extracted features. Numerous studies have been conducted in the field of IS using clustering. During the literature survey, multiple image segmentation techniques like K-Means Clustering, Adaptive Thresholding, Watershed and, Canny Edge Detection were studied. Our research uses the Resepiratory_Disease_Detection_Region of Interest(RDD_ROI) image segmentation technique which considers pixel correlations followed by K-means clustering for the image segmentation and locate Region of Interest(ROI). The pixels with similar properties are clustered together in image segmentation. RDD_ROI performs better with RDD_DNN compared to other image segmentation techniques. For more details refer: https://shorturl.at/pwBPS/

Step 4. Designing 28 layers DNN model(RDD_DNN)

The proposed Deep Neural Network model, RDD_DNN is 28 layers custom-built DNN model designed by understanding the importance of each DNN layer and setting parameters to achieve the efficient performance of the model. RDD_DNN is suitable for image classification tasks with multiple classes. The architecture includes multiple convolutional layers with pooling and dropout, followed by fully connected layers and an output layer with a sigmoid activation function. The model uses Batch Normalization to stabilize training and improve generalization.

Layer	Filters	Kernal size	Padding	Output shape	Activation function
Input	.	.	.	256 x 256 x 1	.
Conv 1	32	3 x 3	Same	256 x 256 x 32	ReLu
BatchNormalization	.	.	.	256 x 256 x 32	.
MaxPooling	.	3 x 3	.	85 x 85 x 32	
Dropout	rate = 0.25	.	.	85 x 85 x 32	.
Conv 2	64	3X3	Same	85 x 85 x 64	ReLu
BatchNormalization	.	.	.	85 x 85 x 64	
Conv 3	64	3X3	Same	85 x 85 x 64	ReLu
BatchNormalization	.	.	.	85 x 85 x 64	.
MaxPooling	.	2 x 2	.	42 x 42 x 64	.
Dropout	rate = 0.25	.	.	42 x 42 x 64	.
Conv 4	128	3 x 3	Same	42 x 42 x 128	ReLu
BatchNormalization	.	.	.	42 x 42 x 128	.
Conv 5	128	3 x 3	Same	42 x 42 x 128	ReLu
BatchNormalization	.	.	.	42 x 42 x 128	.
MaxPooling	.	2 x 2	.	21 x 21 x 128	.
Dropout	rate = 0.25	.	.	21 x 21 x 128	.
Flatten	.	.	.	56448	
Dense	.	.	.	1024	ReLu
BatchNormalization	.	.	.	1024	
Dropout	rate = 0.25	.	.	1024	
Dense	.	.	.	No of classes	Sigmoid

Fig. 2. RDD_DNN model architecture details

Fig. 2 displays the proposed RDD_DNN model by setting the parameters. Different layers are used in this architecture, and their description is discussed below.

Different Layers in RDD_DNN

Convolution Layer: This is the layer where the convolution process happens, and the CNN model learns. This layer performs most of the computations for a CNN model. It is the most important component of a convolutional network. It has some parameters and hyperparameters, namely filters, kernels, and K. Convolution layers extract features using these filters and then learn from them. For this reason, this layer is also known as the feature extraction layer. The input images are compared segment by segment to distinguish the similarities and the differences among them. The segments are called features. The convolution layer extracts one or more features from the input images, and then using the image matrix, it creates a dot product and produces one or more matrices we have an image of size 5 ×5 and the pixel values of the images are either 0 or 1, and the size of the filter matrix is 3 ×3 , then the 3 ×3 filter matrix will be multiplied by 5×5 image matrix, which is known as feature map. The filter moves from left to right with a specific stride value until it parses the full width. Then it moves down to the beginning side of the image with the same stride and repeats the process until the full image is traversed [56].The main goal of this convolution layer is to extract high-level features like edges. Moreover various operations are performed in this layer, including blur, sharpen, edge and detection, by applying various filters. If we have a volume of size A1 ×B1 ×C1 then we need four hyperparameters namely the number of filters Fi , their spatial extent SE, the stride ST, and the amount of zero padding ZP [7] . Then an output of size W2 ×H2 ×D2 is produced the y convolutional layer.

$$A2=(A1-SE+2ZP)/(ST+1) \quad (1)$$

$$B2=(B1-SE+2ZP)/(ST+1) \quad (2)$$

$$C2=C1=Fi \quad (3)$$

MaxPooling Layer:

The pooling layer controls overfitting by slowly decreasing the spatial dimension of representation, no. of parameters, memory footprint, and amount of processing in the network. Pooling is a crucial part of CNNs for object identification. This layer makes the computation fast, prevents overfitting, and reduces memory. There are various kinds of pooling layers, such as max pooling, average pooling, and sum pooling. Max pooling takes the largest value from the feature map, average pooling takes the average by calculating for each patch of the feature map, and sum pooling takes the sum of all elements in the feature map. The most popular and common type of pooling layer is max pooling. The two hyperparameters are required for the pooling layer, namely filter (Fi) and stride (ST). If the volume of an input image is A1 ×B1 ×C1 then an output of size A2 ×B2 ×C2 is produced by pooling layer [57]. The equations for A2, B2, and C2 in pooling layer are given below:

$$A2=(A1-Fi)/(ST+1) \quad (4)$$

$$B2=(B1-Fi)/(ST+1) \quad (5)$$

$$C2=C1 \quad (6)$$

Batch Normalization:

The batch normalization procedure restricts the layer's output to a defined range, imposing zero mean & one standard deviation. This acts as regularization, enhancing the neural network's stability and expediting the training process.

ReLu Layer

ReLU is an acronym for Rectified Linear Unit that uses the non-saturating activation function f(x)=max (0, x). It imparts nonlinearities to decision function & network without influencing the receptive fields of convolution layers. ReLU is frequently favored over other functions since it trains neural networks multiple times quicker while maintaining generalization accuracy.

Loss Layer

"Loss layer" or "loss function" explains how training penalizes the difference between the network's projected output and actual data labels. Based on an individual task, different loss functions may be used.

Dropout Layer

A dropout layer is a regularisation layer initially described in [32]. It may be implemented on any network layer. The primary goal of regularisation is to reduce to zero weights that do not contribute to the accuracy of the model.

Fully Connected (FC) Layer/Dense Layer

Finally, FC layers are utilized to make categorizations. The primary goal of this layer is to categorize the original picture based on all the information retrieved from the preceding layers. In this layer, the input from the past layers is flattened from a matrix into a vector. After flattening, the volume of the previous layer is given input to the fully connected layer like a neural network. By looking at the output of the previous layer, this layer decides which features mostly match a particular class. It works on high-level features that have specific weights. For this reason, a fully connected layer provides the correct probabilities for the different classes as it computes the products between the weights and the previous layer. By using the activation function, the outputs are classified.A Softmax or sigmoid function is used to output target probability after a network.

Activation Function

Activation function We have used the Sigmoid function as shown in another dense layer and ReLU function as shown in Eq. (7) . We have also used an activation function ReLU as shown in Eq. (8) which has proved to give best performance with Maxpooling 2D.

The three activation functions are shown as follows:

$$f(x)=(1)/(1 + \exp(-x)) \text{ ReLu}(0, x) \quad (7)$$

$$f(x) = 1(x < 0)(\alpha x) + 1(x >= 0)(x) \quad (8)$$

In the above equations, x is a real-valued number and αisa small constant. The value of the Sigmoid function ranges from 0 to 1. The Sigmoid function is an S-shaped curve. It is comparatively easy, but it has a vanishing gradient problem. The output is not centered at zero; as a result, the gradient updates go too far in different directions, which makes the optimization harder. On the other hand, ReLU is an efficient and straightforward activation function as it avoids the vanishing gradient problem.

Working principle

The working principle of the RDD_DNN model for layer-wise operation is outlined as follows:

- First input image of size 256 x 256 is fed to the Input Layer.
- Convolutional layer Conv1 uses 32 filters to generate the features map.
- The model starts with a 2D convolutional layer with 32 filters of size (3, 3), using the ReLU activation function. It employs zero-padding ('same') to keep the spatial dimensions unchanged after the convolution and Batch Normalization to normalize the activations.
- MaxPooling2D layer with pool size (3, 3) is applied to reduce the spatial dimensions and control overfitting through the Dropout layer with a dropout rate of 0.25.
- The model continues with two more sets of Convolutional, Activation, Batch Normalization, MaxPooling, and Dropout layers. The second set has 64 filters, and the third set has 128 filters.
- After these convolutional layers, the model adds a flattened layer to convert the 3D output into a 1D vector.
- A fully connected layer (Dense) with 1024 units and ReLU activation is added. Batch Normalization and Dropout (dropout rate of 0.5) are applied after this layer to further regularize the model.
- The output layer consists of a Dense layer with several of neurons equal to the number of classes in the classification problem. It uses the sigmoid activation function to produce probabilities for each class independently.

Deep Learning features of the model helps in the identification of the early stage of respiratory diseases and helps radiology person to label unknown sample and diagnose the disease

Summary of proposed 28-layer RDD_DNN model

Model: "sequential"

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 256, 256, 32)	320
activation (Activation)	(None, 256, 256, 32)	0
batch_normalization (BatchN normalization)	(None, 256, 256, 32)	128
max_pooling2d (MaxPooling2D)	(None, 85, 85, 32)	0
dropout (Dropout)	(None, 85, 85, 32)	0
conv2d_1 (Conv2D)	(None, 85, 85, 64)	18496
activation_1 (Activation)	(None, 85, 85, 64)	0
batch_normalization_1 (Batc normalization)	(None, 85, 85, 64)	256
conv2d_2 (Conv2D)	(None, 85, 85, 64)	36928
activation_2 (Activation)	(None, 85, 85, 64)	0
batch_normalization_2 (Batc normalization)	(None, 85, 85, 64)	256
max_pooling2d_1 (MaxPooling 2D)	(None, 42, 42, 64)	0
dropout_1 (Dropout)	(None, 42, 42, 64)	0
conv2d_3 (Conv2D)	(None, 42, 42, 128)	73856
activation_3 (Activation)	(None, 42, 42, 128)	0
batch_normalization_3 (Batc normalization)	(None, 42, 42, 128)	512
conv2d_4 (Conv2D)	(None, 42, 42, 128)	147584
activation_4 (Activation)	(None, 42, 42, 128)	0
batch_normalization_4 (Batc normalization)	(None, 42, 42, 128)	512
max_pooling2d_2 (MaxPooling 2D)	(None, 21, 21, 128)	0
dropout_2 (Dropout)	(None, 21, 21, 128)	0
flatten (Flatten)	(None, 56448)	0
dense (Dense)	(None, 1024)	57803776
activation_5 (Activation)	(None, 1024)	0
batch_normalization_5 (Batc normalization)	(None, 1024)	4096
dropout_3 (Dropout)	(None, 1024)	0
dense_1 (Dense)	(None, 3)	3075
activation_6 (Activation)	(None, 3)	0

Total params: 58,089,795

Trainable params: 58,086,915

Non-trainable params: 2,880

____

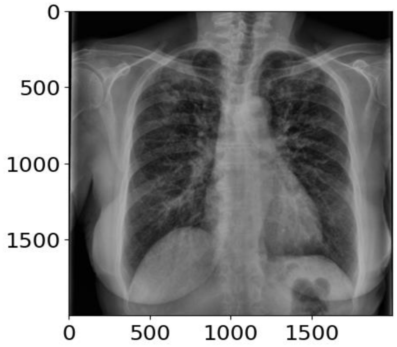
Number of layers in the model: 28

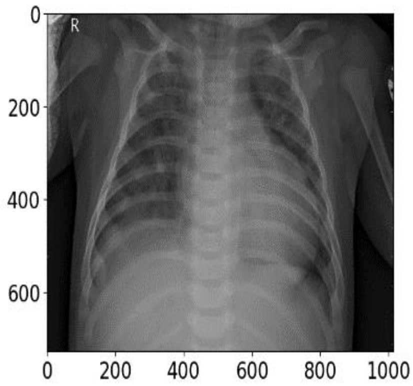
Step 5. Feature Extraction & Classification using RDD_DNN

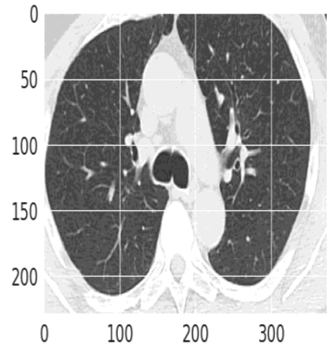
Finally, enhanced and segmented images are obtained with the preprocessing mechanism. These datasets are divided into a training set, validation set, and testing set in a 60:20:20 ratio. The RDD_DNN model is used to perform feature extraction and classification of segmented radiology images of CT-Scan & CXR images. All required hyperparameters including optimizer, batch size, epochs, and loss function are set then this model is trained with the training dataset. In this model convolution maxpooling, and batch normalization were utilized to extract features in which barrel features are constructed then a dropout layer is used that works by randomly disabling neurons and their corresponding connections, and dense layers are used to perform the classification of radiology images. Once model has been trained the test images dataset is utilized to the test model. Finally calculate loss and accuracy and other classification performance parameters for both image datasets.

Step 6. Labelling/ Respiratory Disease Prediction

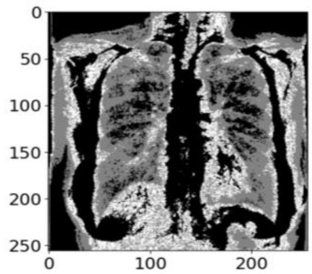
The final step of the RDD_DNN method considers the classification/prediction of unknown radiology image samples for the presence or absence of respiratory disease in the human body. Here we have considered real-time dataset -RESP dataset.





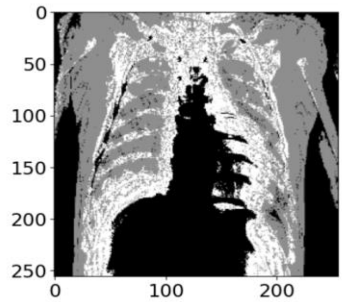


(a)Sample Input Unlabeled Radiology Images



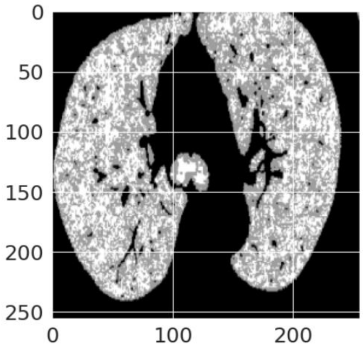

```
[ ]  img = img.reshape(-1, IMG_SIZE, IMG_SIZE, 1)
     prediction = model.predict(img)
     idx = np.argmax(prediction)
     print("Predicted class  : ", new_labels[idx])
     print("Prediction Score : ", prediction[0][idx]*100)

     1/1 [==============================] - 0s 22ms/step
     Predicted class  :  COVID-19
     Prediction Score :  99.74387884140015
```




```
img = img.reshape(-1, IMG_SIZE, IMG_SIZE, 1)
prediction = model.predict(img)
idx = np.argmax(prediction)
print("Predicted class  : ", new_labels[idx])
print("Prediction Score : ", prediction[0][idx]*100)..

1/1 [==============================] - 0s 95ms/step
Predicted class  :  BACTERIAL
Prediction Score :  99.99502301216125
```




```
img = img.reshape(-1, IMG_SIZE, IMG_SIZE, 1)
prediction = model.predict(img)
idx = np.argmax(prediction)
print("Predicted class  : ", new_labels[idx])
print("Prediction Score : ", prediction[0][idx]*100)
```

```
1/1 [==============================] - 0s 98ms/step
Predicted class  :  Non-Covid
Prediction Score :  99.99998807907104
```

(b) Output Labelled Radiology Images with RDD_DNN

Fig.3 Respiratory Disease Prediction using RDD_DNN

As shown in Fig. 3, the RDD_DNN model is able to label unknown samples and detect the presence or absence of respiratory diseases like COVID, Pneumonia etc.

Figure 4 displays the proposed flow diagram of RDD_DNN model on radiology images consisting of different phases like image acquisition, image data preprocessing, feature extraction & classification. The steps for the RDD_DNN model architecture are described below:

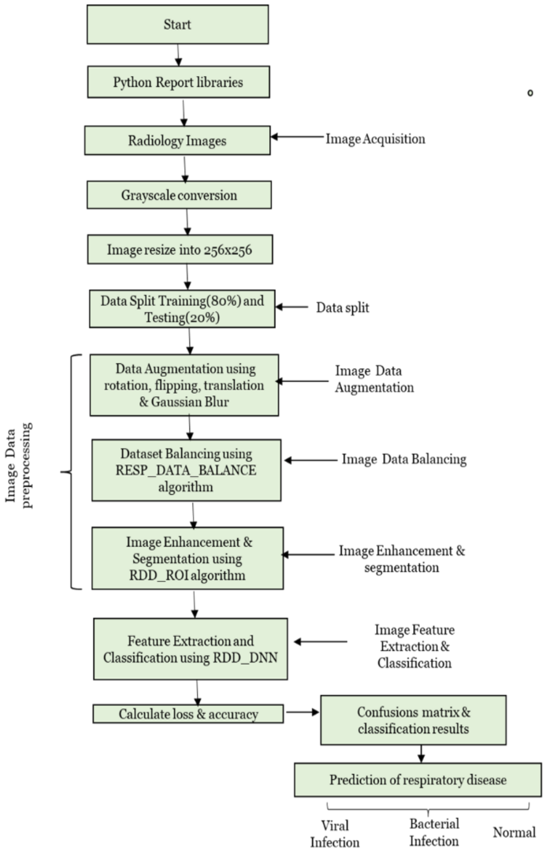


Fig.4. Proposed flow diagram of proposed RDD_DNN on radiology images

Step 1: Import Libraries

The first stage in putting the proposed concept into action is to bring in the relevant libraries such as (NumPy, Pandas, OS, OpenCV, Matplotlib, etc.). While the NumPy library offers objects for multi-dimensional arrays, the Python package Pandas for data analysis includes an in-memory 2D table object named Data Frame. OpenCV converts each & every array type to and from NumPy arrays. This improves interoperability with other NumPy-based libraries, such as SciPy and Matplotlib. Matplotlib is a Python & NumPy tool for plotting & visualizing data on several platforms. While the NumPy library offers objects for multi-dimensional arrays, the Python package Pandas for data analysis includes an in-memory 2D table object named DataFrame. OpenCV converts each & every array type to and from NumPy arrays. This improves interoperability with other NumPy-based libraries, such as SciPy and Matplotlib. Matplotlib is a Python & NumPy tool for plotting & visualizing data on several platforms.

Step 2: Image Acquisition Phase

It is an initial step is to gather radiology images. The initial step in the process is to gather a set of X-ray & CT pictures (infected and uninfected) from the existing database and real-time datasets mentioned above in Table 1, Table 2, and Table 3

Raw data is the form in which the picture was obtained. A great deal of noise can be seen in the obtained picture. It must be pre-processed to enhance contrast transparency and background noise separation.

Step 3: Grayscale color conversion and image resizing

However, the collected images are in Grayscale images but when processed these images are considered RGB color images, so Grayscale color conversion performs on these images to change the color. Digital scans of printed radiology pictures make up the images in the gathered dataset, and there is no uniformity in terms of image size (the smallest image in the collection is 104x153 pixels and the biggest is 484x416 pixels). Furthermore, the images in this dataset have not been standardized for contrast. So, there was a need to resize the images. Therefore, image resizing is performed to standardize the dataset, and images are resized into 256x256 pixels.

Step 4: Dataset Augmentation and dataset balancing

In this step, we are using Data Augmentation techniques like

1. Rotation- from -25 to +25 degrees at random,

2. Flipping the images horizontally,

3. Translation, with random settings both for the x and y-axis,

4. Edge detection using the Gaussian Blurring method

Following by proposed image data balancing RESP__DATA_BALANCE algorithm is applied which produces balanced dataset for training DNN. For more details refer: https://shorturl.at/pwBPS/

Step 5: Image enhancement and segmentation of radiology images

In the case of respiratory disease diagnosis using radiology images, to locate Region of Interest(ROI) is important. To preprocess the image and locate ROI, image enhancement and segmentation are achieved using RDD_ROI algorithm.

The proposed technique improvises and speeds up existing GLCM techniques on a statistical correlation between neighboring pixels. Linear dependency is the most basic type of dependency and the proposed method considers linear dependency of neighboring pixels of a given kernel window size. Features extracted for important pixels are then given to all pixels in surrounding using a weight matrix hence it reduces overall computation time during the image feature extraction. The feature vector related to every non-key pixel is a weighted sum of feature vectors for all weighting kernel windows containing that pixel. To speed up the calculation and decrease the amount of time spent on it, we compute GLCM features for essential pixels also then utilize interpolation to estimate GLCM features for other pixels.

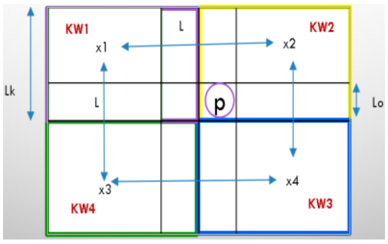


Fig 5. Four Lk×Lk weighting kernel windows

Fig.5 displayed four Lk×Lk weighting kernel windows (Kwi) with the overlap of length Lo. Here, features of pixel p are the weighted sum of that of

$$\text{xi(p)} = \sum_{i=1}^{n} ai(p)f(xi) \quad (9)$$

where,

f(p)= feature vector of non-key pixel p

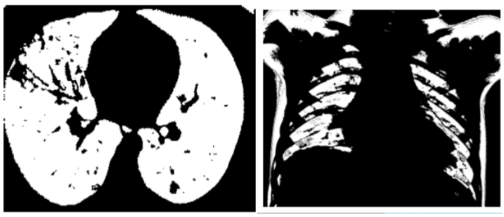
xi = is the key pixel located at the center of each of the overlapping weighting windows which include p;

ai(p) = weight associated with pixel p in the weighting window centered at xi

In this work, the GLCM method extracts seven texture features like homogeneity, correlation, dissimilarity, contrast, entropy, energy, and ASM of the X-ray and CT images. These statistics offer insights into the image's texture [25][26]. Figure 3 represents the enhanced images for both datasets.

Unsupervised segmentation and clustering of radiology images

IS (Image Segmentation) is a technique of splitting the digital image into discrete sections containing pixels (also called superpixels) with appropriate attributes. Image segmentation aims to transform an image's representation into something more meaningful & simpler to study. IS is frequently employed to detect objects & boundaries (curves, lines, etc.) in an image. IS is the labeling of every pixel in an image such that pixels with the same label have comparable attributes. IS is a low-level IP task that aims to separate an image into homogeneous areas [27]. Intensity, discontinuity, or similarity are fundamental precepts of segmentation algorithms. Fig. 6 displays The first category divides an image based on small intensity differences, like edges. The second category involves splitting the image into similar ones depending upon predefined criteria. IS is a method of splitting an image into distinct groups [28][29].



(a) CT scan image (b) X-ray image

Fig. 6. Segmented images using radiology images

Step 6: Classification of Pre-processed and Segmented Radiology Images using the RDD_DNN Model

In this step, Training of the labeled radiology images, feature extraction, and Multiclass Classification is achieved through the proposed RDD_DNN model. This deep neural network also helps in the detection or labeling of respiratory disease from unknown image samples .Therefore, the proposed image preprocessing and DNN design helps in the early detection of respiratory disease present in the human body using radiology images and benefit society.

Proposed Algorithm

A proposed algorithm used for the implementation of the proposed model is given below:

Algorithm: Proposed RDD_DNN

Input: Radiology images datasets (X-ray images & CT images)

Output: Classification of radiology images

Strategy:

Step 1. Importing requires Python libraries for data preprocessing such as (NumPy, Pandas, OS, OpenCV, matplotlib, and so on)

Step 2. Input radiology image datasets

Step 3. Loading dataset

Step 4. Pre-process data by applying data pre-processing techniques

Step 5. Convert input image into Grayscale image

Step 6. Resize the images into 256*256

Step 7. Apply the GLCM feature to characterize the texture of an image

Step 8. Generate Feature Matrices for the key pixel value of specified Kernel Windows size

Step 9. Extract features like Correlation, Contrast, Homogeneity, Entropy, Variance, and energy are features extracted from GLCM matrices

Step 10. Repeat steps 6 to 10 for the remaining neighboring key pixels at a distance

Step 11. Calculate the weight associated with non-key pixels

Step 12. Select no. of clusters to find which is k. For the proposed algorithm, K=2 is considered.

a. Cluster 1=Foreground Bright Area
b. Cluster 2= Background Dark Area

Step 13. Assign data points at random to one of the k clusters.

Step 14. Then, compute the cluster's center.

Step 15. Compute the distance between every data point & cluster's center.

Step 16. Based on the distance between every data point & cluster, reassign data points to the closest clusters.

Step 17. Determine the new cluster center once again.

Step 18. Repeat steps 9, 10, & 11 until the data points no longer alter the clusters or until the allotted number of iterations has been reached.

Step 19. The dataset should be split into training data (60 percent) & validation data (20 percent) & testing data (20 percent).

Step 20. Use deep neural network model and train the model

Step 21. Perform hyperparameter tuning and set different layers

Step 22. Test the model and evaluate the performance of the model

Output: Classification of radiology images and detection of Multiclass Respiratory diseases like COVID, Pneumonia, etc.

4. Results and Discussion

The experiments have been done using Python programming. These experiments have been done on CXR images & CT scan images and analyzed results for COVID-19 classification. The section displayed the results of segmented images and classified results of different collected image data.

Experimental Setup

The experiments have been run on the Jupyter Notebook environment. It is an initial step to gather radiology images. In the initial step of the process, a set of X-ray & CT images (infected & non-infected) are extracted from the public database at [7],[8]. The picture obtained is in its raw format. An abundance of noise is detected in the acquired picture. It must be pre-processed to enhance contrast transparency and background noise separation The dataset is partitioned into 3 parts with 60% training, 20% validation & 20% test set. For the classification model, some hyperparameters like binary crossentory as loss function, Adam optimizer, 32 batch size & 100 epochs are tuned for the CT scan image dataset.

In contrast, categorical crossentory as loss function, Adam optimizer, 32 batch size & 100 epochs are tuned for x-ray image dataset. Evaluation of the model's performance is crucial to any process, including ML. This is the process of making predictions using the trained model on unlabelled data. We then measure the accuracy with which the model made these predictions for classification tasks.

Performance Metrics

The classification performance report evaluates the quality of a proposed classification model. It works both for binary and multi-class classification of COVID-19 detection. This report is generated for an RDD_DNN model or as a comparison.

Accuracy is a metric that may be used to evaluate classification methods. Informally, accuracy is the proportion of correct prediction made by our model. It is a correlation between the accuracy of value and information.

$$.\boldsymbol{Accuracy} = \frac{(\mathrm{TP} + \mathrm{TN})}{(\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN})} \tag{10}$$

Where, TP = True Positive, TN = True Negative, FP = False Positive, & FN = False Negative.

Precision: It measures the proportion of positively categorized occurrences or samples that were accurately classified. Therefore, the precision calculation formula is as follows:

$$\boldsymbol{Precision} = \frac{\text{TP}}{(\text{TP} + \text{FP})} \tag{11}$$

Recall: It is a measure utilized to assess the prediction performance of the classification model on a certain class of interest, commonly named positive.

$$\boldsymbol{Recall} = \frac{\boldsymbol{TP}}{(\boldsymbol{TP} + \boldsymbol{FN})} \tag{12}$$

F1-Score: It represents the harmonic mean of precision & recall. It integrates recall & precision into a single number. We calculate an average of recall and precision for the F1 score.

$$\boldsymbol{F1 - Score} = \boldsymbol{2} * \frac{\boldsymbol{Precision * Recall}}{(\boldsymbol{Precision} + \boldsymbol{Recall})} \tag{13}$$

Area Under the Curve (AUC):

AUC is an acronym for "Area under the ROC Curve." Thus, AUC

evaluates the full 2D area under the complete ROC curve (consider integral calculus) from (0,0) to (1,1). (1,1). AUC is a metric that assesses the capacity of the classifier to differentiate between different types of data. The larger the AUC, the greater model's capacity to distinguish between positive and negative classifications. The receiver Operating Characteristic (ROC) curve shows the performance of the classification model across all classification levels. The graph illustrates two factors

- True Positive Rate (TPR)
- False Positive Rate (FPR)

TPR is a synonym for recall; hence it is described by this formula, which is given below:

$$\boldsymbol{TPR} = \frac{\text{TP}}{(\text{TP} + \text{FN})} \tag{14}$$

The formula for FPR is described below:

$$\boldsymbol{FPR} = \frac{\text{FP}}{(\text{FP} + \text{TN})} \tag{15}$$

A ROC curve compares TPR to FPR at various classification levels.

A. Results and Comparison

In the analysis of the results, conventional CT scans & CXR pictures were compared with those of Covid-19 afflicted individuals. Also, a comparative discussion with different deep learning models and progressive DNN layer architecture i has been given in this section.

1)Results of Binary Class Classification using RDD_DNN with CT Scan images

This section discusses the preprocessing and classification results after applying the proposed model to CT scan images. EfficientNetB0 and EfficientNetB3 with the proposed RDD_DNN model are analyzed based on the accuracy and loss curve, AUC curve, and other classification results. Then, the results were compared to decide that the suggested RDD_DNN model was the best.

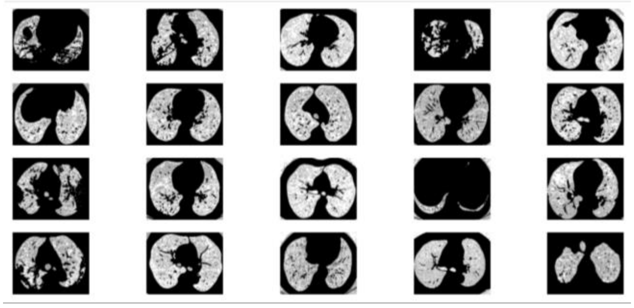
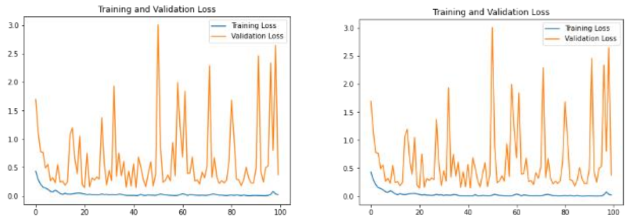


Fig. 7. Preprocessed radiology images of CT scan

Fig. 7 displayed the preprocessed image after enhancing and segmenting input CT images. It displayed the segmented region of COVID with the help of RDD_ROI.



(a) Accuracy (b) Loss

Fig. 8. Training & validation accuracy & loss of proposed RDD_DNN model on CT images

Figure 8 depicts training & validation loss & accuracy analyses of the RDD DNN model applied to CT scan images. Figure 9(a) depicts model accuracy for the RDD DNN model as it increases with successive epochs; training accuracy is greater than validation accuracy. Figure 9(b) depicts validation loss for the RDD DNN model as it decreases with successive epochs, whereas training loss was continuously reduced from 0-100 iterations.

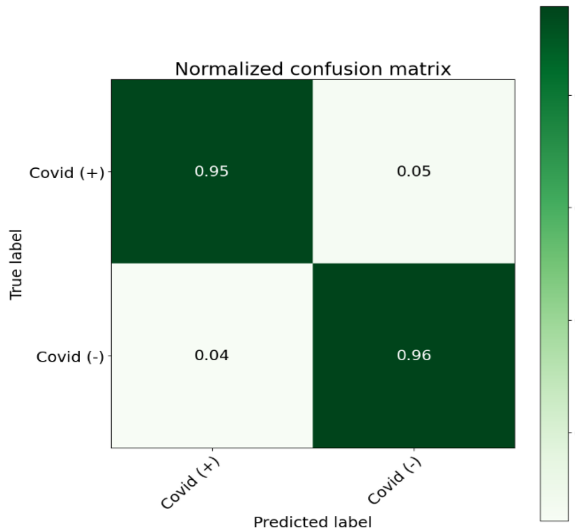


Fig . 9. Confusion matrix of proposed RDD_DNN model on 2-Class Classification using CT images

Fig. 9 shows the normalized binary confusion matrix on the CT scan test set using the RDD_DNN model to predict COVID (+) or COVID (-). This matrix plotted between a true label and a predicted label for these two classes of COVID from 0.0 to 1.0 scale. The matrix shows a true positive value of 0.95 for the COVID (+) class and a true negative value of 0.93 for the COVID (-) class.

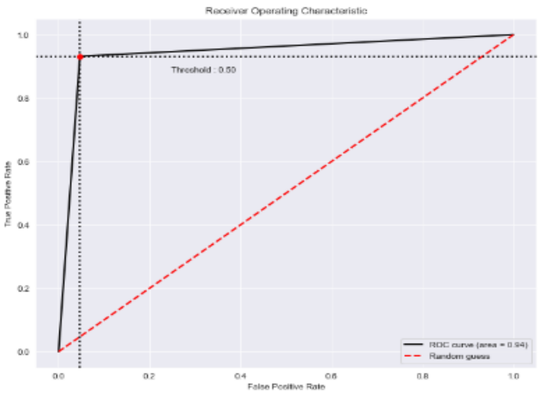


Fig. 10. ROC AUC curve of the proposed RDD_DNN model on CT images

Fig. 10 depicts the ROC curve of a model using CT scan pictures. It is positioned between TPR & FPR. The curve is found at 0.94 with a 0.50 threshold; therefore, the AUC area is 0.94.

Table 4. Comparative analysis of the proposed model with other techniques for CT-scan images

Model	***Training loss***	***Training accuracy***	***Validation loss***	***Validation accuracy***
EfficientNetB0	0.2919	0.9022	0.3985	0.8330
EfficientNetB3	0.2614	0.9103	0.2343	0.9155
Proposed RDD_DNN	0.0204	0.9912	0.3719	0.9517

In Table 4, we provide the outcomes of the proposed model using models presented in Reference [34]. On both datasets, including SARS-Cov-2 and CT-Covid, it is possible to notice that EfficientNet-B0 has worse performance when compared to EfficientNet-B3, which is discussed in the literature. We present a deeper network to avoid overfitting while training existing models. When a deeper network (see "Proposed RDD DNN" in Table 4) is used as opposed to existing models, an improvement in all reported figures is noticed. The best model utilizes the Proposed RDD_DNN input size of 256x256. However, EfficientNet-B3 provides good training and validation accuracy with loss results but has not performed better than the proposed **RDD_DNN** model.

Table 5. Classification performance results in comparison for CT-scan images

Model	***Precision***	***Recall***	***F1-Score***	***AUC***
EfficientNetB0	86.61	87.30	86.96	86.71
EfficientNetB3	93.16	92.80	92.98	92.54
Proposed RDD_DNN	0.95	0.93	0.94	0.94

The best-suggested model is compared to other models in the literature in Table 5. Despite the accuracy results presented for EfficientNetB0 and EfficientNetB3 with the proposed model in table 4 on both datasets of CT scan, both evaluated their models with additional performance parameters to classify the performance of covid detection (COVID (+) and Covid (-)) also, thus, they cannot be directly contrasted to the proposed work. Therefore, references [34] give the best results already found in this setup. Using a slightly smaller model, the work presented here surpasses it in terms of accuracy & F1-score on the SARS-Cov-2 pictures dataset. The existing model provided in Reference [34] requires 14,149,480 and 4,779,038 parameters, but the model presented here requires just 58,088,770 parameters. Training & validation loss curve results for the proposed model are 0.0204 and 0.5954for SARS-Cov-2 images; the comparison results found that the proposed model has achieved 95% precision, 93% recall, 94% F1-score and 0.94 AUC on SARS-Cov-2.

2)Results of Multi clas(3-Class) Classification using X-ray images

This section discusses preprocessing and classification results after applying the proposed model to X-ray images. Inception Net V3, Xception Net, and ResNet are studied using the suggested RDD DNN model, which is dependent upon accuracy matrices and a curve. The outcomes were then compared to identify the optimal model. Even though the model's accuracy is rather good, we suggest verifying its performance with future updates to the dataset. Due to a lack of data, only 1560 samples are used to train the model.

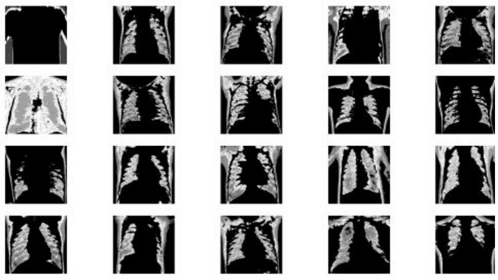
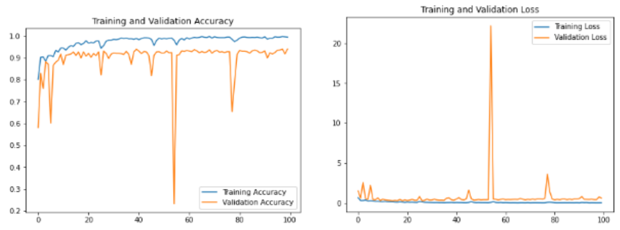


Fig.11. Preprocessed radiology images of x-ray

Fig. 11 displayed the preprocessed image after enhancement using the GLCM method and segmentation of input chest x-ray images using k-means clustering. It displayed the segmented region of COVID and pneumonia with the help of k-means clustering, as seen in figure 11 above.



(a) Accuracy (b) Loss

Fig. 12. Training & validation accuracy & loss of proposed RDD_DNN model on x-ray images

Fig. 12 depicts training & validation loss & accuracy analyses of the RDD DNN model applied to the X-ray pictures dataset. Fig. 12(a) illustrates the accuracy of the RDD DNN model as it improves over consecutive epochs; training accuracy is greater than validation accuracy & Figure 12(b) demonstrates validation loss for the RDD_DNN model as it decreases with successive epochs while training loss was constantly reduced from 0 to 100 iterations.

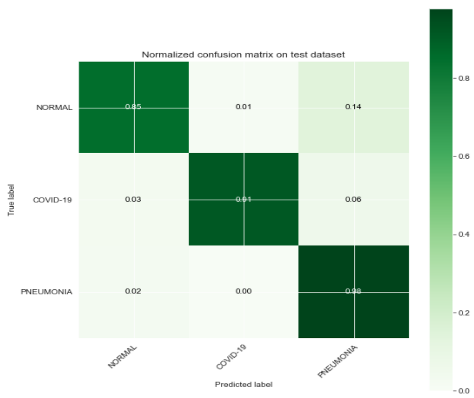


Fig. 13. Confusion matrix of proposed RDD_DNN model on 3-Class Classification with X-Ray images

Fig. 13 demonstrates the normalized confusion matrix of a testing set of the RDD_DNN model to multiclassification on the x-ray images dataset. This matrix plotted among normal, COVID-19, and Pneumonia classes from 0.0 to 1.0. The matrix shows the highest true positive value for the normal class getting 0.85, for the COVID class getting 0.91, and 0.98 for the pneumonia class.

Table 6. Comparative analysis of the proposed model with other techniques for X-ray images

Model	***Training Acc***	***Training Loss***	***Val. Acc***	***Val. Loss***
InceptionV3	0.9051	0.2288	0.9092	0.2397
ResNet	0.8989	0.2338	0.8820	0.3102
Xception	0.9349	0.1752	0.9293	0.1979
Proposed RDD_DNN	0.9994	0.0081	0.9397	0.6045

Table 6 represents the comparison results of the proposed RDD_DNN with existing state-of-art deep learning models to evaluate training & validation results of accuracy & loss. Four total models are compared, including the proposed model on the X-ray images dataset. This comparison shows that the InceptionV3 model acquired more than 90% training and testing accuracy, which is higher than ResNet. Still, it is not better than the Xception model, which has 93.49% training accuracy and 92.93% validation/testing accuracy. However, the results of InceptionV3 and Xception models were performed well by achieving more than 90% accuracy, but it was insufficient; therefore, the proposed RDD_DNN model was proposed and obtained 99.94% training accuracy and 93.97% testing accuracy, which was the top most among all models. `

Table 7. Classification performance results in comparison of X-ray images

Model	***Precision***	***Recall***	***F1-Score***
Inception V3	0.91	0.90	0.90
ResNet	0.92	0.92	0.91
Xception	0.93	0.92	0.92
Proposed RDD_DNN	0.94	0.91	0.93

Table 7 represents the comparative classification results of the proposed RDD_DNN with three existing deep CNN pre-trained models [14] to compare the training and validation results in terms of precision, recall, and f1-score. These findings were compared to the X-ray images dataset. This comparison shows that the InceptionV3 model achieved 91% precision, 90% recall, and f1-score, but the ResNet model achieved 91% f1-score, 90% recall, and precision. Still, it is not well that the Xception model has 93% precision, 92% recall & f1-score. However, the results of the ResNet and Xception model were performed well by achieving more than 90% classification results, but the proposed RDD_DNN model obtained 94% precision, 91% recall, and 93% f1-score. This comparison found that the proposed model outperforms existing models in terms of precision & f1-score but the recall was not higher compared to these existing models.

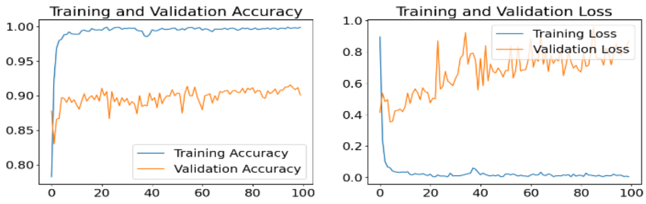
From this comparative analysis, it has been summoned that the proposed model beats many existing DL models of CNN in terms of accuracy and other classification parameters for both CT scan images & X-ray images. The results of the proposed RDD_DNN model are approx. 94% Covid detection rate for X-ray images, but this detection rate was high for CT scan images of Covid, i.e., 95.10%. Thus, overall, we can say the proposed RDD_DNN model achieved 93% to 95% accuracy results for COVID-19 patients' detection.

Table 8. Justification on our 28-layers RDD_DNN model

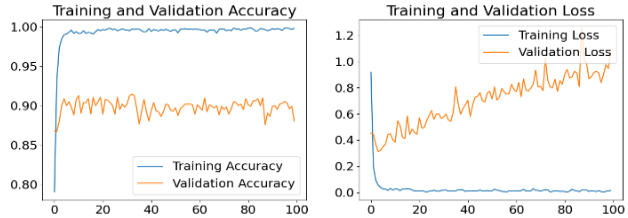
Proposed DNN models	***Accuracy***	***Precision***	***Recall***	***F1-Score***
RDD_DNN with 12 Layers Architecture	0.9155	0.9012	0.8975	0.8954
RDD_DNN with 15 layers architecture	0.9243	0.8912	0.8884	0.8865
RDD_DNN with 20 Layers Architecture	0.9355	0.9214	0..9167	0.9158
RDD_DNN with 23 Layers Architecture	0.9412	0.9363	0.9321	0.9313
RDD_DNN with 28 layers Architecture	0.9513	0.9521	0.9457	0.9476
RDD_DNN with 31 layers Architecture	0.9514	0.9522	0.9458	0.9478
RDD_DNN with 36 layers Architecture	0.9515	0.9523	0.9458	0.9479

Table 8 gives progressive experimental evaluation in the detection of respiratory diseases like COVID-19, Pnuemonia etc. with RDD_DNN model. We have initiated our basis design with 12 layers of DNN including Convolutional layer Conv1 uses 32 filters to generate the features map. The model starts with a 2D convolutional layer with 32 filters of size (3, 3), using the ReLU activation function. It employs zero-padding ('same') to keep the spatial dimensions unchanged after the convolution and Batch Normalization to normalize the activations.MaxPooling2D layer with pool size (3, 3) is applied to reduce the spatial dimensions and control overfitting through the Dropout layer with a dropout rate of 0.25. Slowly progressive approach by The model continues with two more sets of Convolutional, Activation, Batch Normalization, MaxPooling, and Dropout layers. The second set has 64 filters, and the third set has 128 filters and so on.

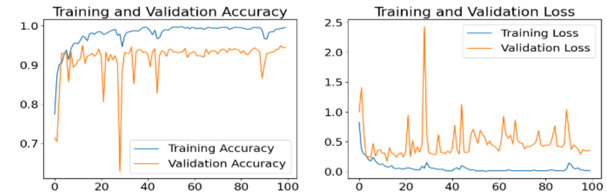
Fig. 14 experimental evaluation shows that , 28 layers custom built model gives higher accuracy and performance in the detection of respiratory disease COVI-19, Pneumonia etc.



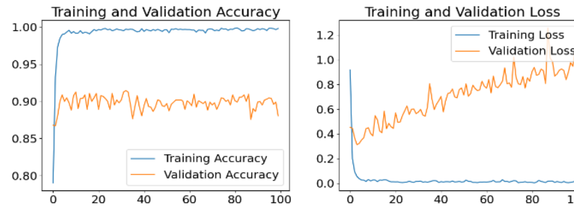
a)Accuracy & loss obtained with RDD_DNN with 12 Layers Architecture



b) Accuracy & loss obtained with RDD_DNN with 15 Layers Architecture



c) Accuracy & loss obtained with 23 Layers Architecture



d)Accuracy & loss obtained with 28 Layers Architecture

Fig. 14. Experimental evaluation Accuracy & loss graph of progressive layers RDD_DNN architecture

Therefore, with the experimental evaluation and performance measurement of RDD_DNN proves that RDD_DNN architecture works well with small number of radiology image dataset as well as large radiology image dataset too. Also helps in the early detection of respiratory disease in human body.

5. Conclusion and Future Work

The coronavirus pandemic 2019 (also known as Covid-19 disease) is a highly infectious respiratory illness. Since then, researchers have investigated several tactics and methods for combating this pandemic. The covid-19 epidemic is constantly multiplying. Due to the rising no. of cases, rapid testing of bulk cases may be necessary. This research suggests a novel DL-based architecture by following image data preparation & DCNN model architectures. The data preprocessing architecture has been designed and processed to balance dataset and enhance image quality. This architecture consists of image acquisitions, image dataset balancing , image enhancement and improved texture feature extraction using GLCM and the segmentation method to extract the region using K-mean clustering.

Further, the preprocessed image was trained by an effective technical RDD_DNN model for early detection of the spread of respiratory disease infection using deep learning techniques to benefit society. Consequently, the DNN model was employed for feature extraction & classification. On the four radiology image datasets of COVID-19 scan, comprising x-ray pictures & CT scan images, experiments have been examined. Proposed work and experimental results focused on improving the preprocessing of radiology images to detect the spread of respiratory disease. Experimental results found that the proposed RDD_DNN model has achieved 95.17% and 93.97% testing accuracy for CT-scan & X-ray images. Also, a comparison has been performed with state-of-arts methods of DL. The comparative analysis found that the proposed RDD_DNN model outperformed existing state-of-art deep learning techniques by achieving more than 93% classification and detection rate. In addition, several performance parameters were used and achieved more than 94% for precision, 91% for recall, and 93% for f1-score for both radiology image datasets. These findings demonstrated that COVID-19 detection techniques in CT images must be greatly improved before they can be considered clinical alternatives and that bigger and more varied datasets are required to test the approaches in a practical scenario. In addition, we determined that RDD_DNN model has the greatest performance and is appropriate for use. We have efficiently classified covid-19 images, Pneumonia images in binary class and multiclass classification. Also, this demonstrates the possible scope of employing such algorithms to automate diagnostic tasks shortly.

The big dataset of CXR may be examined to verify our proposed model. In addition, medical professionals should be consulted for any practical applications of this initiative. We do not want to create a perfect detection system but investigate economically viable methods of combating this disease. Such approaches may be investigated further to demonstrate their use in real-world settings.

6.References and Footnotes

Conflicts of interest

The authors declare no conflicts of interest.

References

[1] K. U. Ahamed et al., “A deep learning approach using effective preprocessing techniques to detect COVID-19 from chest CT-scan and X-ray images,” Comput. Biol. Med., vol. 139, p. 105014, 2021, doi: https://doi.org/10.1016/j.compbiomed.2021.105014.

[2] L. O. Hall, R. Paul, D. B. Goldgof, and G. M. Goldgof, “Finding Covid-19 from Chest X-rays using Deep Learning on a Small Dataset.” pp. 1–8, 2020, doi: 10.36227/techrxiv.12083964.v1.

[3] S. R. Sabbavarapu, S. R. Gottapu, and P. R. Bhima, “A discrete wavelet transform and recurrent neural network based medical image compression for MRI and CT images,” J. Ambient Intell. Humaniz. Comput., vol. 12, no. 6, pp. 6333–6345, 2021, doi: 10.1007/s12652-020-02212-7

[4] M. Razzak, S. Naz, and A. Zaib, “Deep Learning for Medical Image Processing: Overview, Challenges and the Future,” in Lecture Notes in Computational Vision and Biomechanics, 2018, pp. 323–350.

[5] J. Tavares, Advances in Computational Vision and Medical Image Processing: Methods and Applications, vol. 13. 2008.

[6] B. Kitchenham, “Procedures for Performing Systematic Reviews,” Empir. Softw. Eng., vol. 33, no. 2004, pp. 1–26, 2004, [Online]. Available: https://www.researchgate.net/profile/Barbara-Kitchenham/publication/228756057_Procedures_for_Performing_Systematic_Reviews/links/618cfae961f09877207f8471/Procedures-for-Performing-Systematic-Reviews.pdf.

[7] H. Panwar, P. K. Gupta, M. K. Siddiqui, R. Morales-Menendez, P. Bhardwaj, and V. Singh, “A deep learning and grad-CAM based color visualization approach for fast detection of COVID-19 cases using chest X-ray and CT-Scan images,” Chaos, Solitons & Fractals, vol. 140, p. 110190, 2020,

doi: https://doi.org/10.1016/j.chaos.2020.110190.

[8] S. JavadiMoghaddam and H. Gholamalinejad, “A novel deep learning based method for COVID-19 detection from CT image,” Biomed. Signal Process. Control, vol. 70, p. 102987, 2021, doi: https://doi.org/10.1016/j.bspc.2021.102987

[9] K. Gupta and V. Bajaj, “Deep Learning Models-Based CT-Scan Image Classification for Automated

Screening of COVID-19," Biomed. Signal Process. Control, p. 104268, 2022, doi: 10.2139/ssrn.4031534.

[10] R. Jain, M. Gupta, S. Taneja, and D. J. Hemanth, "Deep learning based detection and analysis of COVID-19 on chest X-ray images," Appl. Intell., vol. 51, no. 3, pp. 1690–1700, 2021, doi: 10.1007/s10489-020-01902-1.

[11] A. Akbarimajd et al., "Learning-to-augment incorporated noise-robust deep CNN for detection of COVID-19 in noisy X-ray images," J. Comput. Sci., vol. 63, p. 101763, 2022, doi: https://doi.org/10.1016/j.jocs.2022.101763

[12] N. Phukkaphan et al., "Detection of COVID-19 infection based on electronic nose technique: preliminary study," in 2022 International Electrical Engineering Congress (iEECON), 2022, pp. 1–4, doi: 10.1109/iEECON53204.2022.9741576.

[13] Krizhevsky A, Sutskever I, Hinton GE (2017) Imagenet classification with deep convolutional neural networks.Commun ACM 60(6):84–90 Kroft LJM, van der Velden L, Girón IH, Roelofs

[14] Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V, Rabinovich A (2015) Going deeper with convolutions. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1–9

[15] He K, Zhang X, Ren S, Sun J (2016) Identity mappings in deep residual networks. European conference on computer vision. Springer, Berlin, pp 630–645

[16] Huang G, Liu Z, Van Der Maaten L, Weinberger KQ (2017) Densely connected convolutional networks. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp 4700–4708

[17] Goodfellow I, Pouget-Abadie J, Mirza M, Bing X, Warde-Farley D, Ozair S, Courville A, Bengio Y (2020) Generative adversarial networks. Commun ACM 63(11):139–144

[18] Zhou B, Khosla A, Lapedriza A, Oliva A, Torralba A (2016) Learning deep features for discriminative localization. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp :2921–2929

[19] Ashizawa K, Ishida T, MacMahon H, Vyborny CJ, Katsuragawa S, Doi K (1999) Artificial neural networks in chest radiography: application to the differential diagnosis of interstitial lung disease. Acad Radiol 6(1):2–9

[20] Lodwick GS, Keats TE, Dorst JP (1963) The coding of roentgen images for computer analysis as applied to lung cancer. Radiology 81(2):185–200

[21] Kruger RP, Thompson WB, Turner AF (1974) Computer diagnosis of pneumoconiosis. IEEE Trans SystMan Cybern 1:40–49

[22] Zakirov AN, Kuleev RF, Timoshenko AS, Vladimirov AV (2015) Advanced approaches to computer-aided detection of thoracic diseases on chest x-rays. Appl Math Sci 9(88):4361–4369

[23] Xiaosong Wang, Yifan Peng, Le Lu, Zhiyong Lu, Mohammadhadi Bagheri, Ronald M. 'ChestX-ray8: Hospital-scale Chest X-ray Database and Benchmarks on Weakly-Supervised Classification and Localization of Common Thorax Diseaseses'. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 2097–2106, https://doi.org/10.48550/arXiv.1705.02315

[24] Prashant Patel "Chest X-ray (Covid-19 & Pneumonia)," 2020 : https://www.kaggle.com/datasets/prashant268/chest-xray-covid19-pneumonia

[25] E. Soares, P. Angelov, S. Biaso, M. H. Froes, and D. K. Abe, "SARS-CoV-2 CT-scan dataset: A large dataset of real patients CT scans for SARS-CoV-2 identification," medRxiv, p. 2020.04.24.20078584, 2020, doi: 10.1101/2020.04.24.20078584.: https://www.kaggle.com/datasets/plameneduardo/sarscov2-ctscan-dataset

[26] Z. Xing and H. Jia, "Multilevel Color Image Segmentation Based on GLCM and Improved Salp Swarm Algorithm," IEEE Access, vol. 7, pp. 37672–37690, 2019, doi: 10.1109/ACCESS.2019.2904511.

[27] M. Liang and P. Malm, "3D co-occurrence matrix based texture analysis applied to cervical cancer screening," in Department of Information Technology, 2012, vol. Master in, p. 35.

[28] P. Shan, "Image segmentation method based on K-mean algorithm," EURASIP J. Image Video Process., vol. 2018, no. 1, p. 81, 2018, doi: 10.1186/s13640-018-0322-6.

[29] M. O'Byrne, B. Ghosh, V. Pakrashi, and F. Schoefs, "Texture Analysis Based Detection And Classification Of Structure Features On Ageing Infrastructure Elements," Hal, pp. 1–6, 2018.

[30] A. D.O, O. J. A, A. A.O, and D. A.O, "Comparative Analysis of Textural Features Derived from GLCM for Ultrasound Liver Image Classification," Int. J. Comput. Trends Technol., vol. 11, no. 6, pp. 239–244, 2014, doi: 10.14445/22312803/ijctt-v11p151.

[31] P. Panwar, G. Gopal, and R. Kumar, "Image Segmentation using K-means clustering and Thresholding Image Segmentation using K-means

clustering and Thresholding," Int. Res. J. Eng. Technol., vol. 3, no. 5, pp. 1787–1793, 2016.

[32] M. K. Jalagam, R. Nanda, R. Rath, and G. Rao, "Image Segmentation using K-means Clustering," J. Adv. Sci., pp. 3700–3704, 2020.

[33] N. Dhanachandra, K. Manglem, and Y. J. Chanu, "Image Segmentation Using K-means Clustering Algorithm and Subtractive Clustering Algorithm," Procedia Comput. Sci., vol. 54, pp. 764–771, 2015, doi: 10.1016/j.procs.2015.06.090.

[34] N. Shi, X. Liu, and Y. Guan, "Research on k-means clustering algorithm: An improved k-means clustering algorithm," in 3rd International Symposium on Intelligent Information Technology and Security Informatics, IITSI 2010, 2010, pp. 63–67, doi: 10.1109/IITSI.2010.74.

[35] P. Singh and S. S. Bose, "A quantum-clustering optimization method for COVID-19 CT scan image segmentation," Expert Syst. Appl., vol. 185, no. April, p. 115637, 2021, doi: 10.1016/j.eswa.2021.115637.

[36] Y. Tian, S. Ma, M. Wen, Y. Liu, S.-C. Cheung, and X. Zhang, "To what extent do DNN-based image classification models make unreliable inferences?," Empir. Softw. Eng., vol. 26, no. 5, p. 84, 2021, doi: 10.1007/s10664-021-09985-1.

[37] S. S. Yadav and S. M. Jadhav, "Deep convolutional neural network based medical image classification for disease diagnosis," J. Big Data, vol. 6, no. 1, p. 113, 2019, doi: 10.1186/s40537-019-0276-2.

[38] P. Silva et al., "COVID-19 detection in CT images with deep learning: A voting-based scheme and cross-datasets analysis," Informatics Med. Unlocked, vol. 20, p. 100427, 2020, doi: 10.1016/j.imu.2020.100427.

[39] Pablo Vieira,Orrana Sousa,Deborah Magalhães,Ricardo Rabêlo and Romuere Silva " Detecting pulmonary diseases using deep features in X-ray images", Elsevier - PMC COVID-19 Collection, Published online 2021 Jun 11. doi: 10.1016/j.patcog.2021.108081

[40] Prita Patil; Vaibhav Narawade,' Emphasize of Deep CNN for ChestRadiology Images in the detection of COVID", 2022 IEEE 7th International conference for Convergence in Technology (I2CT), DOI: 10.1109/I2CT54291.2022.982537

[41] Maha Mesfer Meshref AlghamdiORCID Icon,Mohammed Yehia Hassan Dahab &Naael Homoud Abdulrahim Alazwary,'Enhancing deep learning techniques for the diagnosis of the novel coronavirus (COVID-19) using X-ray images', Cogent Engineering,2023, https://doi.org/10.1080/23311916.2023.2181917

[42] Mona Hmoud Al-Sheikh, Omran Al Dandan, Ahmad Sami Al-Shamayleh, Hamid A. Jalab & Rabha W. Ibrahim,' Multi-class deep learning architecture for classifying lung diseases from chest X-Ray and CT images', Scientific Reports, https://doi.org/10.1038/s41598-023-46147-3

[43] Adnane Ait Nasser & Moulay A. Akhloufi,' Deep Learning Methods for Chest Disease Detection Using Radiography Images',SN Computer Science, https://doi.org/10.1007/s42979-023-01818-w