Early Screening of Lung Disease using Deep Learning

Stuti Tiwari[1], Sanskriti Vidushi[1], Archie Sachdeva[1], Gangupomu Hemanthi[1], Prachi Verma[1], Ritika Kumari[1,2], Poonam Bansal[1]

[1]Department of Artificial Intelligence and Data Sciences, IGDTUW, Delhi, India.
[2]USICT, GGSIPU, Delhi, India.
stuti137btcsai22@igdtuw.ac.in, sanskriti114btcsai22@igdtuw.ac.in, archie031btcsai22@igdtuw.ac.in, gangupomu045btcsai22@igdtuw.ac.in, prachi094btcsai22@igdtuw.ac.in, ritikakumari@igdtuw.ac.in, poonambansal@igdtuw.ac.in.

Abstract

Lung Diseases refer to diseases or disorders that affect the lungs and prevent them from functioning properly. The utilisation of Artificial Intelligence in predicting lung diseases have served as a major advantage in the medical field. The detection of presence of lung diseases in a human body through the development of Deep Residual UNET (ResUNET), a Deep Learning Semantic Segmentation technique, has been classified in this paper. UNET architecture along with residual neural network was use in creation of the model. Binary Cross-Entropy classification was used to facilitate efficient training, optimization and comparison in the model. Loss function and Performance Evaluation metrics like Jaccard Index, IoU, Dice Coefficient and Accuracy were used to enhance the model output. The dataset of chest X-Rays and their masks were taken as input variables for model. The predicted masks by the ResUNET model were the evaluated output. The results showed that our model evaluation is estimated at 98.10 percent accuracy.

Keywords: Lung Disease Diagnosis, Deep Residual UNET, Deep Learning, Image Segmentation , Semantic Segmentation, Binary Classification

1 Introduction

Lung diseases are among the most widely affecting medical conditions in the world. Problems in our respiratory tract affects our lungs and subsequently causes lung damage. Lung ailments can be caused by a variety factors such as virus, bacteria, fungal infections and the environment.

Medical imaging, a process used to create segmentational images of different parts of the human body for clinical purposes such as diagnosis or analysis and research has seen major development in recent years. To detect lung diseases, Lung imaging tests are used. Various lung imaging tests include Chest X-Rays, Chest Computed Tomography (CT) scans and Chest Magnetic resonance imaging (MRIs) among which Chest X-rays and CT scans are the most common [1]. As of now Chest X-rays pose advantageous over CT scans due to low radiations, more cost effectiveness, easy availability and faster results [2]. For our model, the dataset incorporated was "Pulmonary Chest X-Ray Defect Detection". The dataset contained chest x-rays for a variety of disease-affected lungs and unaffected lungs as well as the accurate masks of every x-ray.

In the recent years, Deep Learning has revolutionized medical fields by processing huge amount of raw data. Deep Learning is equipped with estimating the location of the desired element by applying pixel-to-pixel characterisation to output the estimated shape, thus providing an accurate projection of the shape and a generalised segmentation mechanism, eliminating the need for feature extraction algorithms. Deep Learning also incorporates Neural Networks, which are layers of nodes connected to the adjacent layer. More the number of Neural Networks in the model, the deeper the network becomes [3][4].

To remove the problem of vanishing gradients as a consequence of increasing depth in neural networks, Deep Residual Networks are utilised, which simplify the training process such that the addition of more layers does not intensify the network training error [4]. To further rectify this problem, Fully Convolutional Neural Networks are used which utilize an identity mapping on the deep residual learning frameworks and help is easier training by integrating layers and refining output precision [5].

Instead of using FCNs that skip layers, UNET, a U-shaped encoder-decoder network architecture, is used. UNET although based on FCNs, operates on fewer training samples and its Encoder network applies two 3x3 convolutions, each accompanied by a Rectified Linear Unit (ReLU) and downsampling focused 2x2 max pooling . The Decoder network involves upsampling along with 2x2 convolution to halve the feature channels, concatenation with the contracting path, and two 3x3 convolutions, each accompanied by a ReLU. ResUNET combines these deep residual network frameworks with UNET, using residual units as basic blocks. The biggest advantage in using ResUNET is being able to build a deeper network without worrying about the training of the model and the problem of vanishing gradients as they are backpropagated from the outer layers to the deeper layers [6].

2 Literature Review

S. Bharati et al. [7]utilized a model incorporating CNN, Vanilla NN, VDSNet, VGG, and Capsule network, employing a hybrid deep learning approach to detect lung diseases from X-ray images collected from a sample dataset consisting of 5606 images. Their findings indicated that VDSNet achieved an accuracy rate of 73 percent. Adem Tekerek et. al [8] used the method to predict lung disease using chest X-ray images. Based on DenseNet and MobileNet, which found and categorized 40,000 X-ray images from a sample dataset. The model employed a convolution neural network based on chest X-rays to classify three types of infection using Deep Learning. The suggested method has a high accuracy of 96% and a ROC AUC value of 0.94. Shoji Kido et al [9] employed Convolutional Neural Network (CNN) and Regions with CNN Features (R-CNN) to detect and classify lung abnormalities on CT scans of 163 individuals with lung nodules. They studied 372 individuals with or without diffuse lung abnormalities on CT scans. Image-based CADx with CNN was used to differentiate various types of lung anomalies. CADx with R-CNN has proven effective for radiologists in diagnosing Slung anomalies. Siddhanth Tripathi et. al [10]utilized a model that combines CNN (Convolutional Neural Network), VGG, data preprocessing, and Transfer Learning (TN) to detect lung diseases using deep learning. They used sample data outputs for the respective lung X-rays as input. The CNN+VGG+data+TN model performed better than the other model (Vanilla CNN), achieving an accuracy of 69.3%. Kamran Javaid et. al [11] conducted a study on Lung nodule detection in Chest X-ray using deep learning and it was found that the combination of transfer learning model VGG16 and Custom CNN is better than other models with an accuracy of 88%. Syamala KPL et. al [12] investigated the detection and classification of lung diseases using Machine and Deep Learning techniques. Their study provides a hypothesis-based latter. They have attained an accuracy of 98.58 percent using EfficientNet B0. M. Jasmine Pemeena Priyadarsini et.al [13] investigated lung disease detection using several deep-learning algorithms. Their research was based on a Kaggle sample dataset that included a wide range of lung-related issues. They used cutting-edge CNN technology and a variety of models to properly diagnose lung illnesses. The results showed an F1 grade of 98.55 percent and a precision of 98.43%. Rakshit S. et. al [14] explored to understand the use of different pre-trained deep learning models like Resnet and Densenet to perform the classification. They offer a model (the Resnet18 network) with minimal parameters for developing and evaluating among the previously evaluated models. Rasika Naik et.al [15] identified lung diseases using Deep Learning. Convolutional neural networks (CNN) are successful in picture recognition and classification. DenseNet is a CNN-based pre-trained model, whereas Densenet-121 is the training model. The AES algorithm is thought to be one of the finest. An average AUC of 0.843 was obtained. Marios Anthimopoulos et. al [16] investigated lung pattern classification for interstitial lung diseases using a deep convolutional neural network. The researchers presented a deep CNN to categorize lung CT image patches into 7 categories, including 6 ILD patterns and healthy tissue. A neural network architecture was also constructed, and their proposed technique produced promising results with an accuracy of 85.5%.

3 Model Architecture

ResUNET is based on both UNET and Deep Residual Neural Networks, thus making its architecture quite similar to the UNET. A ResUNET consists of an expansion path, a contraption path, and a bridge connecting these networks, similar to the UNET. The contraption path collects the features from the image and copies them to higher levels for segmentation. This transferring of features from lower level to higher ones helps in well-defined backpropagation and ease of training.

The involvement of the residual block in ResUNET helps in better training of the model in deeper neural networks. The residual neural networks thus used constitute of layers of stacked residual units where each residual unit had batch normalisation functions and ReLU activation functions [17]. The combined use of UNET and residual blocks solved the problem of training deep neural networks with limited data samples and helped ease the degradation problem by skipping connections and using lesser parameters to attain better results. An overview of the architecture of the ResUNET model used is shown in Fig. 1.

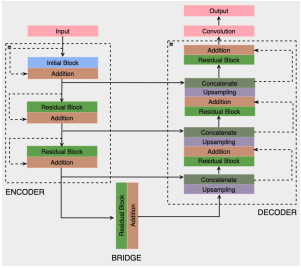

Fig. 1 ResUNET Architecture

4 Materials and Methods

Within this section, we will discuss about the classification of our model, the dataset considered for our model and the model's performance metrics.

4.1 Classification

In order to assess the model and check the associated predictions and optimizations, the Binary Cross-Entropy classification is utilised. Binary classification is used to segregate our observations based on the segmentation of our sampled features. In our model, binary classification is being used to predict the new mask for each existing original masks by binary comparisons. To predict how accurate the output is, cross-entropy loss function is being used. Loss functions tell us how well our model is performing, based on which optimizations are made. Entropy is the measure of the unpredictability associated with the model or system. Cross-entropy is the measure of difference in the entropy between two probability masses. It is often used as a loss function when optimizing a distribution model. The Binary Cross-Entropy is conveniently

used in training and solving classification problems by reducing each classification to a binary choice. The comparison between the predicted and the actual binary outcomes is done by Binary Cross-Entropy loss function [18].

4.2 Dataset Description

The dataset taken for our model was from Kaggle, titled "Pulmonary Chest X-Ray Defect Detection" [7] [8] [9] [19] [20] . The dataset contained 1408 images of 2 types, 704 images of both chest x-rays of varied lungs and each lung's original mask. The x-rays of lungs in the dataset are both of disease-affected and disease-unaffected lungs. The mask for each variety of lung was also provided. Fig 2 shown an example of the dataset images.

4.3 Performance Metrics

Performance metrics are used to estimate model training status and conditions and find out how well the model is working. The most common metric for evaluating a binary classification method in image segmentation is using Accuracy (1). In our case, Jaccard Index (2) and Dice Coefficient (3) are also employed.

Accuracy is the most used metric for evaluating the performance of a model in classification problems. It is described as the ratio of accurately classified data items in comparison with the number of observations [21].

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

The Jaccard Index is a metric that is used to determine how similar varied distributions are. The value of Jaccard Index stresses resemblance between finite sample sets. The greater the value of Jaccard Index, more is the intersection between the two samples [22].

$$JaccardIndex = \frac{TP}{(TP + FN + FP)} \tag{2}$$

The Dice Coefficient is a statistical tool which is used in measuring the similarity between two sets of data, most widely used metric for image segmentation. In Boolean description, Dice Coefficient is expressed in the form of False Negative (FN), False Positive (FP), True Positive (TP) and True Negative(TN) [22].

$$DiceCoefficient = \frac{2 * TP}{(2 * TP + FN + FP)} \tag{3}$$

5 Experimental Setup

In this segment, we will discuss the data preprocessing and visualisation before using the data samples for training in the model along with the ResUNET model implementation.

5.1 Data Preprocessing

The x-rays and the mask images in the dataset were processed using the Python Imaging Library (PIL). The x-ray image of lungs were resized and displayed. Similarly, the mask images were converted to Palette, set to a mask color, resized and displayed. Python and torch libraries like numpy, tqdm, albumentations, torch were imported for data processing. Three variables were created for the lung image directory, mask directory, and mask name directory. The data was shuffled to store 15% of the mask names to a array, randomised at 99 seed and permuted to consider different images of the total dataset at each run. The data was split into 15% for the validation set and 85% for training. The images and masks were converted to greyscale and classified into a float array, with white mask pixels converted to a binary mask. The training and validation sets were resized, normalised, converted to Tensor and saved in training transformation and validation transformation variables respectively.

5.2 Data Visualisation

The training transformation and validation transformation mask names from the above class were split and saved into training dataset and validation dataset variables. DataLoader object was also used in which the data was loaded onto the train and validation dataloader variable. In every dataloader iteration, 1/4th of the entire data was loaded onto the 4 dataloader variables. A function was created to display the tensor form images by transposing and normalising the pixel of each image them which were in the tensor form by multiplying them with standard deviation and adding the mean into it (4) .

$$img = img * std + mean \tag{4}$$

Iterations were then performed on the training dataset and validation dataset variables at each index to fetch the images. The x-ray and mask images were displayed in a grid of 4 images, with 2 images in every row as shown in Fig. 2.

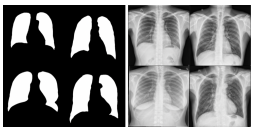

Fig. 2 Lung masks and x-ray samples after grey-scaling.

5.3 Implementation of ResUNET model

The ResUnet Model comprises two class blocks: The Initial Block for feature extraction and the Residual for boosting model efficiency. First, we will start with the InitBlock that consists of the 'Init' function that has object attributes such as 'inchannels' -¿ no of input channels (set 3 for RGB images), outchannels -¿ feature map's depth, and 'stride' -¿ step size in convolutional layers (set 1 by default). At the start block, 'self.convblock' creates a neural network using several layers. To begin, 'nn.Conv2d'

makes 2D convolutions over data having some in/out channels based on kernel size (3x3), stride length, and padding size. Next comes 'nn.BatchNorm2d', which uses mean and variance to normalize the data and perhaps even speed up computation in training neural networks. Last, 'nn.ReLU' is applied to the 'ReLU' activation function. To process feature maps, use 'nn.Conv2d'. Now, 'self.convskip' acts identically to 'self.convblock' with the same processes and parameters. Primarily creates a skip link that adds the original input to the processed output, avoiding steps in 'self.convblock'. It's a common feature in RNN that alleviated the vanishing gradient problem. The forward method is the block's forward pass that simultaneously calls input data through convolutional layers and then combines output element-wise. Important specifics from the input data can be maintained while learning minutiae from convolutional layers. Further, the second block, the Residual Block contains a similar init procedure as the first, but now 'self.convskip' deals with re-batch normalization and 2D convolution after batch normalization. Here, the forward method defines the block's forward pass, accepts tensor 'x' as input data through convolutional layers, and outputs an element-wise sum. This model enhances training and alleviates spatial dimensions. Fig.3 depicts a flow of layers in the Initial and Residual Blocks.

We define a PyTorch module called ResUnet, an extension of the U-Net architecture using residual blocks, which improves training efficiency and performance for image segmentation tasks. The init method accepts inchannels and outchannels, which define the input and output channels of the model, respectively. It also takes features, a list (e.g., [64, 128, 256]) representing feature counts for each U-Net level. The architecture begins with an InitBlock (self.input) to process the input image. The downsampling path (‘self.downs‘) consists of ResBlock modules, each halving the spatial dimensions and doubling the number of features with a stride of 2. The forward method defines ResUnet's forward pass, starting from the input layer and collecting skip connections at each downsampling step. Upsampling processes skip connections and feature maps, with the final output coming from the output layer. A for loop iterates over, adding a ResBlock to self.downs each iteration. A for loop iterates over, adding a ResBlock to self.downs each iteration. Each ResBlock accepts feature channels as input and features two channels as output, with a stride of 2. Another loop runs over features in reverse order, adding a transposed convolutional layer (‘nn.ConvTranspose2d‘) to upsample the feature map, followed by a ResBlock to process it. The transposed convolution doubles the spatial dimensions and channels (2 features). Each ResBlock in the upsampling path has input sizes 2 * feature + feature and output sizes of feature. The final layer (‘self.output‘) converts the feature maps to the desired number of output channels. In the forward method, skipconnections stores intermediate encoding feature maps. The input 'x' passes through, and its feature map sums to skipconnections. During encoding, each iteration over the 'self.downs' downsample extracts feature maps. If the iteration index i is less than 2, 'x' is added to skipconnections. After encoding, skipconnections reverse to align with the decoding layers. It enhances performance and feature preservation in image segmentation tasks.

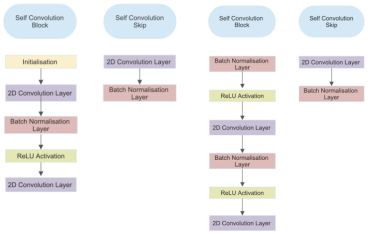

Fig. 3 Propagation in methods of Initial Block and Residual Block

6 Results

The following section describes the evaluation metrics, final results and the learning and loss curves associated with the model.

6.1 Evaluation Metrics

Our model uses the standard equation for Binary Cross Entropy loss function to calculate the epoch loss (5) [22] .

$$J_{bce} = -\frac{1}{M}\sum_{m=1}^{M}[y_m log(h_o(x_m)) + (1-y_m)(log(1-h_o(x_m))] \tag{5}$$

For our research, the evaluation metrics used are Dice Coefficient (6) and Intersection over Union or IoU (7). The Dice coefficient is used to gauge the similarities between the two distributions, which in our case are the true masks and the predicted masks. The higher the Dice score, greater is the overlap with the ground truth.

$$DiceCoefficient = \frac{2|X \cap Y|}{|X| + |Y|} \tag{6}$$

IoU is a value from range 0 to 1 through which the amount of overlap between the predicted and ground truth distributions is estimated. It is similar to the predictions of Jaccard Index [22].

$$IoU = \frac{|X \cap Y|}{|X| + |Y|} \tag{7}$$

While metrics like Dice and Jaccard are used in optimisizing loss, IoU is mainly used to increase accuracy of the model.

6.2 Training Results

The training result of our model is done through the use of epochs. The optimal number of epochs for every model depends on the dataset considered and the main factor which comes into picture is the training and validation error and the loss with every epoch. In our case, 20 epochs are taken. At the end of 20 epochs, the accuracy of the model comes out to be 0.9810 or 98.10 percent as shown in Table 1.

The training and validation curve for loss function and performance and evaluation metrics is also developed as shows in Fig. 4 and Fig. 5.

Table 1 Epoch Training Results

S.No.	*Epoch No.*	*Loss*	*Accuracy*	*Dice*	*IoU*
1	1/20	0.7736	0.9076	0.8302	0.7205
2	2/20	0.3646	0.9629	0.9261	0.8634
3	3/20	0.2550	0.9746	0.9493	0.9041
4	4/20	0.2372	0.9764	0.9530	0.9107
5	5/20	0.2449	0.9757	0.9513	0.9077
6	6/20	0.2051	0.9798	0.9597	0.9229
7	7/20	0.2004	0.9802	0.9606	0.9245
8	8/20	0.1973	0.9805	0.9611	0.9255
9	9/20	0.1981	0.9804	0.9609	0.9252
10	10/20	0.1949	0.9807	0.9616	0.9264
11	11/20	0.1939	0.9808	0.9618	0.9267
12	12/20	0.1942	0.9808	0.9616	0.9265
13	13/20	0.1940	0.9808	0.9616	0.9265
14	14/20	0.1927	0.9809	0.9621	0.9273
15	15/20	0.1927	0.9809	0.9619	0.9270
16	16/20	0.1933	0.9809	0.9620	0.9271
17	17/20	0.1929	0.9809	0.9619	0.9270
18	18/20	0.1936	0.9808	0.9619	0.9269
19	19/20	0.9809	0.1923	0.9622	0.9275
20	20/20	0.1915	0.9810	0.9622	0.9276

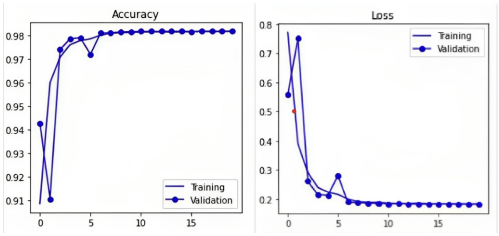

Fig. 4 Training and validation curve for Accuracy and Loss

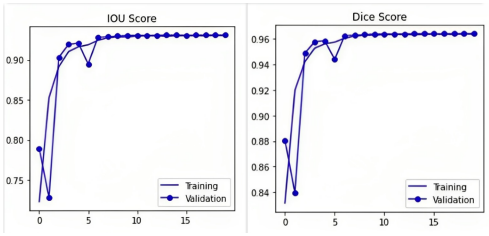

Fig. 5 Training and validation curve for IoU and Dice

6.3 Final Output

The derived output from our model based on binary classification and ResUNET is the predicted mask. The closeness of our mask with the true masks from the dataset is given in Fig. 6. As observed by the samples of predicted and true masks as well as based on the 98.1 percent accuracy, we can say that the utilisation of ResUNET and the training of the model has been done successfully.

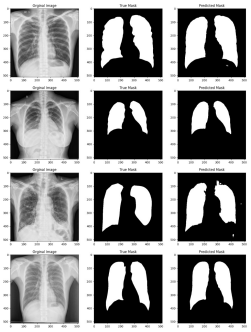

Fig. 6 Final output: Predicted and True masks after model training

7 Conclusion

In this research paper we introduce the ResUnet model, for detecting lung diseases using high-resolution chest X ray images. Our approach combines the benefits of Deep learning and the U-Net Architecture and Residual Neural Networks. By utilizing skip connections within the residual blocks and between the downsampling and upsampling paths, our network effectively extracts features and propagates information in both forward and backward computations. This unique architecture incorporated the best state of the art deep learning approaches for lung disease detection using binary classification. Our proposed model not only simplifies training but also enables us to design networks that are both straightforward and powerful.

References

[1] Vliegenthart R, Fouras A, Jacobs C, Papanikolaou N. "Innovations in thoracic imaging: CT, radiomics, AI and x-ray velocimetry. Respirology", 2022 Oct;27(10):818-833. doi: 10.1111/resp.14344. Epub 2022 Aug 14. PMID: 35965430; PMCID: PMC9546393.

[2] Wang G., Yu H., De Man B. "An outlook on X-ray CT research and development". Med. Phys. 2008;35:1051–1064. doi: 10.1118/1.2836950.

[3] Haruna Chiroma, Shafi'i M. Abdulhamid, Ibrahim A. T. Hashem, Kayode S. Adewole, Absalom E. Ezugwu, Saidu Abubakar, Liyana Shuib. "Deep Learning-Based Big Data Analytics for Internet of Vehicles: Taxonomy, Challenges, and Research Directions". Volume 2021.

[4] Mainak Biswas, Venkatanareshbabu Kuppili, Luca Saba, Damodar Reddy Edla, Harman S. Suri, Elisa Cuadrado-Godia, John R. Laird, Rui Tato Marinhoe, João M. Sanches, Andrew Nicolaides, Jasjit S. Suri. "State-of-the-art review on deep learning in medical imaging". 2019, 24(3), 380–406. https://doi.org/10.2741/4725

[5] Jonathan Long, Evan Shelhamer, Trevor Darrell. "Fully Convolutional Networks for Semantic Segmentation". 1411.4038.

[6] Zhengxin Zhang, Qingjie Liu, Yunhong Wang. "Road Extraction by Deep Residual U-Net". IEEE Geoscience and Remote Sensing Letters. May 2018. 10.1109/lgrs.2018.2802944.

[7] Bharati, S., Podder, P., and Mondal, M. R. H. (2019). "Hybrid deep learning for detecting lung diseases from X-ray images". Informatics in Medicine Unlocked, 20, 100391. https://doi.org/10.1016/j.imu.2020.100391

[8] Tekerek A, Al-Rawe IAM. "A Novel Approach for Prediction of Lung Disease Using Chest X-ray Images Based on DenseNet and MobileNet" . Wirel Pers Commun. 2023 May 12:1-15. doi: 10.1007/s11277-023-10489-y. Epub ahead of print. PMID: 37360137; PMCID: PMC10177707.

[9] S. Kido, Y. Hirano and N. Hashimoto, "Detection and classification of lung abnormalities by use of convolutional neural network (CNN) and regions with CNN features (R-CNN)," 2018 International Workshop on Advanced Image Technology (IWAIT), Chiang Mai, Thailand, 2018, pp. 1-4, doi: 10.1109/IWAIT.2018.8369798.

[10] Siddhanth Tripathi, Sinchana Shetty, Somil Jain, Vanshika Sharma. (2021). "Lung Disease Detection Using Deep Learning". 10.35940/ijitee.H9259.0610821.

[11] Javaid, K., Rehman, F., and Haq, N. (2020). "Lung nodule detection in chest x-ray using deep learning" . Pakistan Science Bulletin, 70(1), 1-14.

[12] Syamala KPL, Niharika CS, Jenny AM, Pavani P (2022). "Detection and Classification of Lung Diseases using Machine and Deep Learning Techniques". J Comput Sci Software Dev 2: 1-10.

[13] Jasmine Pemeena Priyadarsini M, Kotecha K, Rajini GK, Hariharan K, Utkarsh Raj K, Bhargav Ram K, Indragandhi V, Subramaniyaswamy V, Pandya S. "Lung Diseases Detection Using Various Deep Learning Algorithms". J Healthc Eng. 2023 Feb 3;2023:3563696. doi: 10.1155/2023/3563696. PMID: 36776955; PMCID: PMC9918362.

[14] Rakshit, S., Saha, I., Wlasnowolski, M., Maulik, U., Plewczynski, D. (2019). "Deep Learning for Detection and Localization of Thoracic Diseases Using Chest X-Ray Imagery". In: Rutkowski, L., Scherer, R., Korytkowski, M., Pedrycz, W., Tadeusiewicz, R., Zurada, J. (eds) Artificial Intelligence and Soft Computing. ICAISC 2019. Lecture Notes in Computer Science(), vol 11509. Springer, Cham. https://doi.org/10.1007/978-3-030-20915-5-25

[15] Naik, Rasika and Wani, Tejas and Bajaj, Sakshi and Ahir, Shiva and Joshi, Atharva. "Detection of Lung Diseases using Deep Learning" (April 8, 2020). Proceedings of the 3rd International Conference on Advances in Science and Technology (ICAST) 2020.

[16] M. Anthimopoulos, S. Christodoulidis, L. Ebner, A. Christe and S. Mougiakakou, "Lung Pattern Classification for Interstitial Lung Diseases Using a Deep Convolutional Neural Network," in IEEE Transactions on Medical Imaging, vol. 35, no. 5, pp. 1207-1216, May 2016, doi: 10.1109/TMI.2016.2535865.

[17] Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun. "Identity Mappings in Deep Residual Networks". 2016. 1603.05027.

[18] Ruby, Usha, Yendapalli , Vamsidhar. "Binary cross entropy with deep learning technique for Image classification". International Journal of Advanced Trends in Computer Science and Engineering. 9. 10.30534/ijatcse/2020/175942020 (2020).

[19] Pandey, Nikhil. "Chest Xray Masks and Labels." Kaggle, https://www.kaggle.com/datasets/nikhilpandey360/chest-xray-masks-and-labels. Accessed 7 Sept. 2023.

[20] Candemir S, Jaeger S, Palaniappan K, Musco JP, Singh RK, Xue Z, Karargyris A, Antani S, Thoma G, McDonald CJ. "Lung segmentation in chest radiographs using anatomical atlases with nonrigid registration". IEEE Trans Med Imaging. February 2014.

[21] Nillmani, Sharma N, Saba L, Khanna NN, Kalra MK, Fouda MM, Suri JS. "Segmentation-Based Classification Deep Learning Model Embedded with Explainable AI for COVID-19 Detection in Chest X-ray Scans." Diagnostics (Basel). 2022 Sep 2;12(9):2132. doi: 10.3390/diagnostics12092132. PMID: 36140533; PMCID: PMC9497601.

[22] Ishu Anand, Himani Negi, Deepika Kumar, Mamta Mittal, Tai-hoon Kim,Sudipta Roy. "Residual U-Network for Breast Tumor Segmentation from Magnetic Resonance Images". Computers, Materials and Continua. DOI:10.32604/cmc.2021.014229. 30 November 2020.